Jean-Marc Pierson, Helmut Hlavacs (Ed.)

Proceedings of the

COST Action IC0804
on
Energy Efficiency in Large Scale Distributed Systems

1st Year

Editors

Jean-Marc Pierson
IRIT
University Paul Sabatier
31062 Toulouse cedex 9, France
Email: Jean-Marc.Pierson@irit.fr

Helmut Hlavacs
Distributed and Multimedia Systems
University of Vienna
1080 Wien, Austria
Email: helmut.hlavacs@univie.ac.at

ISBN: 978-2-917490-10-5 - EAN : 9782917490105

COST- the acronym for European Cooperation in Science and Technology- is the oldest and widest European intergovernmental network for cooperation in research. Established by the Ministerial Conference in November 1971, COST is presently used by the scientific communities of 35 European countries to cooperate in common research projects supported by national funds.

The funds provided by COST - less than 1% of the total value of the projects - support the COST cooperation networks (COST Actions) through which, with EUR 30 million per year, more than 30 000 European scientists are involved in research having a total value which exceeds EUR 2 billion per year. This is the financial worth of the European added value which COST achieves.

A ”bottom up approach” (the initiative of launching a COST Action comes from the European scientists themselves), ” la carte participation” (only countries interested in the Action participate), ”equality of access” (participation is open also to the scientific communities of countries not belonging to the European Union) and ”flexible structure” (easy implementation and light management of the research initiatives) are the main characteristics of COST. As precursor of advanced multidisciplinary research COST has a very important role for the realisation of the European Research Area (ERA) anticipating and complementing the activities of the Framework Programmes, constituting a ”bridge” towards the scientific communities of emerging countries, increasing the mobility of researchers across Europe and fostering the establishment of ”Networks of Excellence” in many key scientific domains such as: Biomedicine and Molecular Biosciences; Food and Agriculture; Forests, their Products and Services; Materials, Physical and Nanosciences; Chemistry and Molecular Sciences and Technologies; Earth System Science and Environmental Management; Information and Communication Technologies; Transport and Urban Development; Individuals, Societies, Cultures and Health. It covers basic and more applied research and also addresses issues of pre-normative nature or of societal importance.

Web: http://www.cost.eu

EUROPEAN SCIENCE FOUNDATION ESF provides the COST Office through an EC contract

COST is supported by the EU Framework programme

Introduction to the 1st Year Proceedings of the COST Action IC0804

Jean-Marc Pierson (Chair)
IRIT, Université de Toulouse, France

Helmut Hlavacs (Co-Chair)
, University of Vienna, Austria

The COST Action IC0804 proposes realistic energy efficient alternate solutions to share distributed resources. As large scale distributed systems gather and share more and more computing nodes and storage resources, their energy consumption is exponentially increasing. While much effort is nowadays put into hardware specific solutions to lower energy consumptions, the need for a complementary approach is necessary at the distributed system level, i.e. middleware, network and applications.

The Action characterizes the energy consumption and energy efficiencies of distributed applications. Then based on the current hardware adaptation possibilities and innovative algorithms it proposes adaptive and alternative approaches taking into account the energy saving dimension of the problem. The Action characterizes the trade-off between energy savings and functional and non-functional parameters, including the economic dimension.

This book gathers the presentations held at the working group meetings and the 1st year Workshop of the Action, in Toulouse (November 2009) and in Passau (April 2010). It is organised in three sections, for Working Group 1, 2 and 3.

This book was made possible by the commitments of the Working Group chairs of the Action that we would like to thank hereby:

- WG1 : Davide Careglio, Universitat Politcnica de Catalunya, Spain, and Georges Da Costa, Université Paul Sabatier, France
- WG2 : Alberto E. Garcia, University of Cantabria, Spain, and Karin A. Hummel, University of Vienna, Austria
- WG3 : Hermann de Meer, University of Passau, Germany, and Laurent Lefévre, INRIA, France

WG1 : State of the art and continuous learning of hardware adaptation possibilities

Georges Da Costa Davide Careglio

This first year lead us to create the foundations of knowledge about leverages on hardware. In order to smartly manage systems to reduce energy, the first step is to obtain knowledge about current possibilities of hardware.

WG1 focused on providing summary and references on existing and future leverages to adapt the underlying hardware infrastructures of large scale distributed computing systems in order to decrease their energy consumption.

This work is intended to be used by other Working Groups of this Action to drive their research fostering activities towards energy aware middleware for large scale distributed computing systems. It can can also be useful for other audiences: ex. academics, researchers, companies, general public, data-center administrators, politicians,...

Current work address three main areas:

- Fine analysis of possible hardware leverages
- Middleware for accessing hardware leverages
- Coarse grained possible hardware leverages and best practices (such as datacenter cooling, monitoring,...).

Apart from publishing a brochure consisting of several chapters each addressing one hardware resource, such as the processor, main memory, storage (disk and flash), motherboard, fan, network interface and including a chapter discussing/presenting existing energy aware practices in large scale systems, WG1 participants presented their current work to the Cost Action participants.

Those works were contribution to the three main areas.

I. Fine analysis of possible hardware leverages

A broad evaluation[4] of each element which can be tuned in order to reduce energy consumption: Network, CPU, Memory, Storage, PSU, Accelerator. Two other studies focuses on two different elements, a study of the possibilities of CPU[2], and one on Hard Drives[3]. This last study focuses on the impact of quiet mode and access pattern on energy consumption of Hard Drives.

II. Middleware for accessing hardware leverages

The SMOA Devices platform power consumption monitoring and control architecture[1] uses a RESTFul xmpp protocol to enable communication and remote control of physical elements.

III. Coarse grained possible hardware leverages and best practices (such as datacenter cooling)

An example of possible improvement was shown based on the construction of MareNostrum in 2005 at the Barcelona Supercomputing Center as it achieved a 10% reduction of energy consumption thanks to taking into account the energy consumption during the design of the computing center.

IV. Workshop proceedings

The following articles will show those current researches that were presented in Toulouse and Passau during the first year of the Action. They will mainly focus on each subsystem (CPU and Hard drives).

Apart from the best practices that were describe in the WG1 Brochure, it is to be noted that installation of MareNostrum in the Computing Center of Barcelona provided us with an interesting use case. It reminded the participants that working in an integrated way provide the more benefits and thus, several leverage can be used at the computing center level during its design:

- Improving the Power Chain
- Improving the Air Management
- Improving Humidity Control
- Provide a precise monitoring to users

The final conclusion is that providing simple and efficient ways to monitor and change hardware state allows to improve energy consumption drastically. There is still lacks of systems, middlewares, and abstraction in this field but in the following some contribution are exposed to provide them.

References

[1] Krzysztof Kurowski, Ariel Oleksiak and Michał Witkowski, *SMOA Devices - A Distributed Management And Control System For Energy Consumption In Computing Environments*

[2] Avi Mendelson, *Managing Modern Computer systems – Principles and challenges*

[3] Doron Chen, George Goldberg, Roger Kahn, Ronen I. Kat, Kalman Meth, and Dmitry Sotnikov, *Disk Storage Power Management*

[4] Wolfgang Rehm, René Oertel *GreenCUT - Research and Future Plans*

GreenCUT - Research and Future Plans

René Oertel
Computer Architecture Group
Department of Computer Science
Chemnitz University of Technology
09107 Chemnitz, Germany
Email: rene.oertel@cs.tu-chemnitz.de

Wolfgang Rehm
Computer Architecture Group
Department of Computer Science
Chemnitz University of Technology
09107 Chemnitz, Germany
Email: wolfgang.rehm@cs.tu-chemnitz.de

***Abstract*—Energy-aware software and hardware design and usage is a popular theme in many areas, e.g. high-performance computing and large scale data centers. Every component of these systems are potential targets for power and energy optimizations. During the last years the main focus of optimizations were costs and size constraints. Package density of the systems increased quickly and with this the cooling and power requirements per square-meter. This short paper gives an overview where our research activities are focused and which further investigations our group proposes. It is a summary of a presentation held in November 2009 in Toulouse, France. Our knowledge and technical expertise is based partly on our high-performance cluster *CHiC* at Chemnitz University of Technology (CUT) and new results and developments will influence the successor of the *CHiC*.**

I. Introduction

Energy-aware technologies are a trend today and will get in focus of many more researchers all over the world in the future. This trend is a result of more and more complex systems which should satisfy the growing demands on IT infrastructure to solve problems of the humankind. Unfortunately, solving these problems e.g. of energy production creates new problems in the same field of research.

This short paper is organized as follows. Section 2 describes our research environment at CUT as an overview. Section 3 discusses several power-related topics and research areas of our group in different subsections. Finally, the Chapter Conclusion provides a short advice for future research in the field of energy optimizations.

II. Equipment at CUT

In late 2006, the Chemnitz High-Performance Linux Cluster (*CHiC*) [1] designed by our computer architecture group, went into a production environment. It is used by the whole research community at the Chemnitz University of Technology (CUT) for e.g. calculations in maths, mechanical engineering and chemistry. It is a 2048-core+ AMD Opteron system with 10 GBits SDR InfiniBand interconnect technology. The 60 TiBytes parallel file system Lustre is distributed over 10 storage servers and 10 JBODs with 16 SATA hard disk drives each. The majority of the compute nodes is equipped with 2 dual-core AMD Opteron processors and 4 GiBytes of DDR2 RAM. Only a few nodes have a memory extension to 8 and 16 GiBytes.

The cluster and storage systems are housed in 18 water-cooled racks manufactured by Knürr AG. The cool water is provided by the local energy supplier over a long-distance cooling system. The water temperature is initially approx. 9 degrees Celsius, but we use a input water temperature of 17 degrees Celsius. The water is heated to approx. 21 degrees Celsius. The thermal energy is then re-used for community heating in near shopping centers.

III. Energy-aware HPC Today

Our research interests focus on the following topics. We have a broad knowledge about network technologies like InfiniBand and extended expertise about the relationship of communication and computation. Aspects of CPUs and memory power consumption are essential for optimizations. The usage of adapted storage and application specific accelerators will drive our future research.

A. Network

Different types of network equipment infrastructure could show different power consumptions. Hoefler et al. [2] showed that different protocols and technologies may have a large influence. Application benchmarks with InfiniBand over copper, Myrinet over copper and Myrinet over fiber cables resulted in different application run times. Additionally, the power draw on the wall plug was different for the different kinds of cables. In this single case, Myrinet with fiber cables provided the best energy efficiency. In a typical cluster interconnect, not only the NICs are relevant, but the switches too. Port count, number of line cards, cooling fans and redundant PSUs can make a huge different between network technologies or manufactures.

As a result, benchmarks of a particular use case or application could provide a different view to power consumption. Synthetic benchmarks may not show the real applications behavior in case of performance and power consumption.

B. CPU

CPUs are very important for optimizations. They are one of the components with the largest power requirements. Fortunately, CPU design and efficiency was improved during the last years both for calculation capability and adaptability. Today, CPUs are built for HPC and embedded systems with different capabilities. The embedded sector is driven by constraints of

power consumption and heat and the HPC sector mostly only by the performance requirements. Modern CPUs have different internal power modes (C-/P-states) and allow several steps of scaling, e.g. frequency or cache size in relation to the constraints given to the system. Unfortunately, these technology enhancements need additional support of the software layer. The operating system must also be power-aware to allow an enhancement in energy efficiency, because a bad distribution of idle or full workloads forces more units to work than necessary. The Linux OS has scheduler support for power saving loadbalancer and the recent version of the Microsoft Windows operating system improves scheduler efficiency of multi-core CPUs. What is true for a single system is also important for a cluster system. Batch systems (job schedulers and resource managers) should be able to consolidate running jobs to fewer cluster nodes if they are not full utilized. This is the case of under-utilized systems. In other cases additional technology enhancement of the bare-metal are necessary to extend efficiency. Hardware vendors like AMD or Intel are required to improve CPU efficiency for all types of workloads and to integrate e.g. mobile power enhancements into high-performance CPU models, too.

C. Memory

Analysis of different memory modules of the same or different type showed that there are several parameters influencing the power consumption of a system. The general rule, more DIMMs use more power is true in most cases. Additionally, memory chips with different configurations can consume twice the power. Technology enhancements, i.e. DDR2 to DDR3 RAM, reduce power consumption at the same clock speed. Unfortunately, newer systems also raise the clock speed to improve performance, but this is not a problem, if the computing efficiency increases in the same ratio.

D. Storage

The storage systems of large data centers or high-performance clusters ofter consist of several hundreds to thousands of spinning hard disk drives. But not only the disk as such require a lot of energy, but the storage server managing the disks and the Storage Area Network (SAN) infrastructure, too. A current trend is the usage of solid-state disks (SSDs) for special applications. Surprisingly, many SSDs have a power consumption comparable to previous spinning hard disk drives. Fortunately, a large advantage of SSDs is the capability to serve many more I/O requests in the same time than their traditional counterparts. Problems at the moment are bad GiB/EURO ratios, capacity constraints and wearing of SSDs (refer to [3]). Our proposal is to use different kinds of mass storage devices, i.e. large and lower-power spinning hard disk drives for large amounts of data and small, but fast SSDs for high responsive scratch space in a parallel cluster file system.

E. PSU

The efficiency of power supply units (PSUs) of all IT infrastructure components play an important role of the whole system power consumption. PSUs are often designend for the highest power state the component requires. Unfortunately, these components often work at a lower power level. At this level, PSUs are not as efficient as intended. The 80 PLUS[1] program is an interesting initiative to provide a guideline to development and integrate efficient power supplies, which work optimal in different power levels.

F. Accelerator

One of our research fields are application-specific accelerators. The main drawback of accelerators like GPUs or FPGAs are their specialization. They are not useful for general computations and must be selected carefully for distinct usage scenarios. In the other cases they would only consume additional valuable energy and would not provide any benefit. A guideline which accelerator best fit to a given problem dimension and algorithm would be very beneficial for application porting decisions and optimizations of power consumption.

IV. Conclusion

Analyzing, modeling and optimizing of today's and future IT infrastructures are challenging tasks. Different approaches for this tasks are imaginable. The top-down approach should be used to analyze current systems to understand there power consumption in detail. In contrast, future developments of all kinds of IT components may comprise energy- and power-aware technologies from the beginning on. The bottom-up approach for designing systems would benefit if power properties and parameters are well-known. Not only the bare metal must be involved into power optimization processes, but the software, i.e. the user of these real entities, too. This results in a tight communication between hardware and software designers, technology providers and system architects. A common base for discussions must be created and understand by all people. It should hide specific details of the several areas but provide enough information to enable optimizations between the several levels of abstractions.

Acknowledgment

The authors would like to thank the COST ICT Action IC0804 chair, Jean-Marc Pierson and the Working Group 1 chair Georges Da Costa and Davide Careglio for the exciting possibility to participate at the first workshop meeting in November 2009 in Toulouse, France.

References

[1] F. Mietke, T. Mehlan, T. Hoefler, and W. Rehm, "Design and Evaluation of a 2048 Core Cluster System," in *Proceedings of 3rd KiCC Workshop 2007*. RWTH Aachen, Dec. 2007.

[2] T. Hoefler, T. Schneider, and A. Lumsdaine, "A Power-Aware, Application-Based, Performance Study Of Modern Commodity Cluster Interconnection Networks," in *Proceedings of the 23rd IEEE International Parallel & Distributed Processing Symposium, CAC'09 Workshop*, May 2009.

[3] D. Narayanan, E. Thereska, A. Donnelly, S. Elnikety, and A. Rowstron, "Migrating server storage to SSDs: analysis of tradeoffs," in *EuroSys '09: Proceedings of the 4th ACM European conference on Computer systems*. New York, NY, USA: ACM, 2009, pp. 145–158.

[1] http://www.80plus.org

Managing Modern Computer systems – Principles and challenges

Avi Mendelson
Microsoft
avim@microsoft.com

1 Introduction

Power related issues consider as one of the most important aspects of designing modern processors since it affects many design aspects of the entire system. When discussing power issues, we need to consider different aspects of the problem:

- Energy consumption – how much energy (power over time) the system consumes when execute a piece of work (workload). This parameter mainly affects battery life of mobile systems and the cost of operation of other systems such as servers, data centers and cloud computers.
- Power consumption – how much power the processor (or the system) consumes at a certain point of time. This parameter affects the power delivery subsystem and in many cases, the cooling system, since the die must not exceed max temperature due to chemical and reliability limitation.
- Power density – the distribution of power consumption to different subsystems. This parameter has a significant impact of the internal design and the max temperature of certain parts of the system. This aspect is out of the scope of this presentation.
- Dynamic power vs. leakage power; dynamic power is defined as power the system consumes while working and leakage power is defined as a power the system, or sub-system consumes while not doing any active work. Leakage power starts to be very significant in modern architectures. Many techniques proposed to reduce leakage power, all of them require significant latency when moving from sleep mode to active mode.

All systems are sensitive to power related issues, but not all systems are sensitive to the same set of constrains, For example, some battery operated systems are so sensitive to energy consumption that they work in minimum frequency, thus heat is a less of a problem for them. Gaming machines are designed to run as fast as possible, they are mostly connected to the utility and so, energy is not consider to be a problem for them (users will to pay the electricity bill), thus heat is what prevent them from running faster. In general, laptops and servers need to balance between the different power affects as described before. Some systems aims to get maximum performance for a given power envelop, while other systems aims at achieving best power consumption for a given task, while not exceeding the power and heat limitations.

Power management of the processor can be critical for man systems, since on one hand, the processor consumes a significant part of the entire system's energy consumption, and on the other hand, due to his small die size, it is relatively difficult to cool it and so the cost of cooling can be very significant, or the processor power consumption need to be restricted. Since the amount of heat the processor produced, is proportional to the power it consumes and the

power consumption (P) depends on its frequency (f) and voltage (V) and its effective capacitance (C); i.e.,

Equation : P = C * f * V^2

Thus in order to prevent the system from overheating, we may need to reduce its frequency and the voltage of the processor and so to lose performance.

Observation 1: In most modern systems, power and performance depends on each other.

It means that suppose you can come with a technique that reduce power (please note, power not Energy), I can always use the thermal headroom created, in order to increase frequency and get more performance (it assumes that modern processors are designed to run at higher frequency if thermal budget allows).

The understanding that the power consumption of the processor impacts the direct and indirect cost of the system and its performance cause power to be first class citizen in any modern design. But controlling power through Hardware only mechanisms was found not to be optimal, so many HW/SW techniques were developed such as: AMD's PowerNow! and Intel's SpeedStep, that help to control the Voltage and the Frequency of the processor as a function of the workload being executed. But the most common techniques to control the different aspects of the power related issues in processors and the entire system, was developed as a consortium between Intel, Microsoft, HP Phoenix and Toshiba and is called Advanced Configuration and Power Interface (ACPI).

2 ACPI

The ACPI specification [1] is quite complicated and contains 700 hundreds of pages that cover many SW and HW related issues. In this report we will focus only on the main features that impact the operation modes of the CPU (processor) and include three mechanisms, termed Thermal control Zone, Power state (P-state) and CPU's state (C-state).

The ACPI defines the power states of the entire system. But in this document we will focus on the power states of the processor only.

2.1 T- States: Thermal control Zone:

the ACPI defines a set of events to prevent the system form getting over-heated. These events include "Trip_points" which are dynamically defined events that indicate to the OS to change the speed of the fan or to change the speed of the CPU, and a "Critical Shutdown" event that that indicates that the system MUST shut down immediately to prevent damages [2].

2.2 C-States: Sleeping states. Controls static power

Controls how deep the CPU "sleeps" when is not active. The deeper the CPU goes, less leakage power it consumes but it also significantly increases the latency of the system to wake-up. Thus the ACPI defines 4 states (as we will see later HW companies extend it)

- C0 is the operating state, it consumes dynamic power
- C1 (Halt) is the state where the processor is not executing anything but can come back at C0 in a few cycles (but the saving is minimal)
- C2 (Stop-Clock) is a deeper sleep state that consumes less leakage power than C1 at the cost of a slower wake-up (this state is optional and usually not implemted)
- C3 (Sleep) offers improved power savings over the C1 and C2 states. The worst-case wakeup latency for this state is provided via the ACPI system firmware and the operating software can use this information to determine when the C3 can be used and when higher states must be used to guarantee critical response time

All modern processors extend the notion of C3 to further refinements. For example, I7 (Intel) implements the notion of C6 state. But from the

ACPI point of view (SW/HW interfaces) only 3 of them exist and the rest are handled y HW only.

2.3 P-States: dynamic power control state

indicates how fast the processor should run when in C0 state. System can define a table of frequency/voltage operational points and OS/SW can define at what operational work the system will work

- P0 is the max frequency/voltage state
- P1 is a state where frequency and voltage are reduced
- Pn is a state where frequency and voltage are reduced compared to Pn-1

The way OS handled P states is by applying dynamic learning algorithms, It sample the system every period of time (usually 100MS) and determine the utilization of the system during that period of time. If found that the system was “busy” most of the time, it reduces its P state (faster) and if found that the CPU was at sleep state most of the time, it increases its P-state (slower). By doing that the systems tries to optimize between the power the system consumes to the performance it can get. Please note that T states are independent of P-State and may impact the “absolute frequency” the system can run at P0.

3 Vendors

3.1 Intel

Recently Intel published significant amount of new data regarding the power management of the new I7 processor’s family. This section extends the discussion on some of these features to provide a better picture on how power is managed in modern cores.

The implementation of the Throttling mechanism in I7 is not well document, so we will based the discussion on the implementation of CoreDue-2 as appeared in [2]. Here two mechanisms were discussed, the two levels mechanism and the dynamic throttling mechanism.

The two levels mechanism (implemented in P4 family or processors) defines two operations points “normal” and “halt”. While in “normal” mode the system works corresponding to the Pn state. When the system reaches Max-temp trip point (this is usually at 90C or 100C) it indicates the OS to change the state to halt. While in “Halt” state, the system waits until cool down below a specified point and change to “normal” again. If for any reason the system reaches the critical-shutdown point, the HW shut the system down immediately to prevent any damages.

The more sophisticated mechanism descried there is a dynamic throttling, here, at any operational point, the system defines two trip points, upper and lower. When the temperature cross the upper trip point, an HW mechanism force the system to slow down (redefine the values of the P-state table) and when the temperature cross the low-trip point, it allow the system to work faster (limited by max frequency.

I7 Core extends the implementation of the C states and defines new state termed C6. (since it is not exposed to the ACPI it is purely handled by HW.).

The states I7 Core implements are:

- C0: when the microprocessor is in the active state (some P-State)
- C1: no instructions are being executed; controller clock-gate all gates pertaining to the core pipeline. Clock gating is accomplished by logically ANDing the clock signal of a particular clock domain with a conditional control signal
- C3: the core phase-locked-loops (PLLs) are turned off, and all the core caches are
- flushed. A core in C3 is considered an inactive core. The time it takes to the core to wake up is significantly longer that C1 since the PLLs are linear-feedback based control systems, which need to be turned back on, time must be allocated for the PLLs to lock (stabilize) to the correct frequency return to “full speed”. More than that, the time it take the system to return to full utilization is even longer, due to the cold start of the cache.

- C6: the most power efficient state, the core PLLs are turned off, the core caches are flushed and the core state is saved to the Last Level Cache (LLC). The power gate transistors activated to reduce leakage power consumption to a particular core to near to zero Watts. A core in idle state C6 is considered an inactive core. The wakeup time a core in idle state C6 is the longest since the core state must be restored from the LLC, the core PLLs must be re-locked, the power gates must be deactivated, and the caches starts from clean state. Please note that waking up from C6 may consume significant power, so the system need to make sure that the power it saves is greater than the power it waste for entering and exiting from the state. To prevent this, i7 Core, includes an auto-demote capability that uses intelligent heuristics to optimize the use of this aggressive state.

I7 Core also presents a revolutionary approach on how to manage P states and the overall thermal budget of the core. Intel discovered [5] that when only a single core is active it never use the entire power and thermal envelop allowed for the 4 CPU die. Thus they introduce the notion of "Intel Turbo Boost Technology" that allows a core to increase its frequency above the maximum allowed frequency (the official frequency of the core) if thermal and power head rooms allows it.

4 Challenges

Although significant amount of design and research efforts are spent in understanding and improving power management of computer systems in general and processor control in particular, the fact the power related issues depends on many different aspects and parameters, make this problem to be contra intuitive and so, this area needs a very careful treatment in order to avoid (common) mistakes for example:

4.1 Power vs. Performance curve

Looking at equation 1, doesn't provide to the reader the full picture, because due to design an other considerations, the voltage of the system can very only between V_{min} to V_{max}

- The system cannot lower the Voltage below V_{min}, but can still reduce the frequency using the same V_{min}.
- The system cannot go beyond V_{max}, and so cannot increase the frequency beyond that point.
- In the region between V_{min} to V_{max} the C and F are almost linearly dependent.

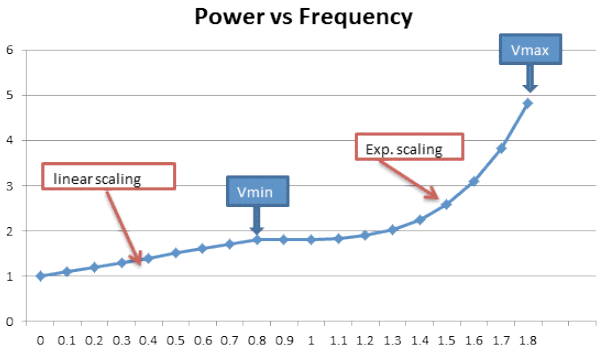


Figure : Frequency vs. Power dependencies in the system

This graph indicate that the amount of power we are paying as a result of "delta" change in the frequency depends in the operational point of the system. If the operational point tis beyond Vmin, the power penalty is in order of cube of the change, but if the change is done below the Cmin, the impact is only linear.

4.2 Operational point of multiprocessors systems

Most of the multiprocessor system are design to work at Vmin, or at an

operational point which is very close to Vmin.
The reason for that is that multiprocessors share the same heatpipe and so are thermally limited. Thus in such systems, when die become too hot, we can reduce the frequency but it will provide only a linear reduction of the heat, and under the same token, when only part of the processors are active, we can use the extra thermal headroom to speed up some of the active processors (similar to what Intel are doing), but will cost us in the order of cube power

This fact is usually ignored by many researchers

4.3 Thermal tornado

Few researchers offer that when only subset of the cores are active, to change the active subset of the processors all the time, so that the core can run faster than their thermal steady state point, and when reach the thermal critical point, the execution will be swapped to a new (hopefully) cold core.

This techniques sounds very promising, but also has few challenges that need to be address; e.g., need to reduce the overhead of swapping cores so that the overall performance will have some gain, the calculation of the heat time, vs. useful execution time and it may also impact reliability issues of the die due to frequent change in the thermal of the die

5 Summary and future work

This white paper provides short description of some of the state of the art aspects of power management of processors as part of computer based systems.
It also presents few challenges in this field.

The next step of the research is to focus on the specific techniques that can help laptops and server systems to achieve better performance at the same power envelop
provide a schematic dependency between F and P in the system

References

[1] ACPI specification -

http://www.acpi.info/spec.htm

[2] Alon Naveh, Efraim Roterm, Avi Mendelson, Simcha Gochman, Rajshree Chabukswar, Karthik Krishnan, Arun Kumar , "Power and Thermal Management in the Intel® Core™ Duo Processor Architecture" in Intel Technology Journal, Volume 10, issue 02, pp 109-122, 2006.

[3] Efi Rotem, Alon Naveh, Micha Moffie and Avi Mendelson, "Analysis of Thermal Monitor features of the Intel® Pentium® M Processor “ in TACS workshop, at ISCA-31, June 2004.

[4] Power management of Intel I7 cores -

http://cs466.andersonje.com/public/pm.pdf

[5] AMD Reveals More Llano Details at ISSCC: 32nm, Power Gating, 4-cores, Turbo?:

[6] http://www.anandtech.com/show/2933

[7] **Gelsinger talk at IDF-08 on Nehalem power management: http://news.zdnet.com/2422-19178_22-216954.html**

Disk Storage Power Management

Doron Chen, George Goldberg,,
Roger Kahn, Ronen I. Kat, Kalman Meth, and Dmitry Sotnikov
{cdoron,georgeg,rogerk,ronenkat,meth,dimitrys}@il.ibm.com
IBM Research - Haifa, Israel

***Abstract*—Data center power management has become increasingly important in recent years. In particular, the need to understand and manage storage power consumption has arisen. We developed a framework for estimating the power consumed by the storage components of a data center under varying workloads. Such a framework is useful for capacity planning tools, for enabling estimation of future performance and power consumption, and for online storage systems providing power estimation per disk, per array, and per volume. In addition, we present a technique for controlling the power consumed by disk drives that support *acoustic modes*. This technique reduces instantaneous power consumption but sacrifices performance.**

1 Introduction

Data center power considerations play an increasingly important role in the operation of data centers. This trend only increases with the growing demand for storage [22]. Since storage accounts for 13-20% of the cost of powering and cooling a data center [11], [23], [25] understanding and controlling storage power consumption is becoming increasingly important.

The power consumption of disk drives consists of two parts. The fixed portion, or *static power*, is the power consumed when the disk is in the idle state. The static power is the result of the disk spindle motor that spins the platters, and the onboard disk electronics. The variable portion, or *dynamic power*, is the power that is affected by the I/O workload. The factors which contribute to the dynamic power are the data transfer to/from the disk and the power required to move the disk head during a seek. The total power consumed by a disk is the sum of the dynamic and static power. The dynamic power of the disk can be as much as a third of the total disk power consumption. The power consumption of the disk can be further divided into mechanical power (using a $12V$ power source) for the disk spindle and seek head, and electronics power (using a $5V$ power source) for the disk electronics and data transfer operations.

Detailed understanding of storage power consumption is critical to data center management. Proper management of power consumption, in accordance with realistic workloads, can prevent over-provisioning for power and cooling.

Our contribution. We present two innovations: one for power modeling and estimation of storage, and the other for controlling and budgeting the power consumption of disks. Our modeling and estimation framework provides workload-aware power estimations for disks, disk arrays and storage controllers. It translates system-level or RAID-level operations to disk-level activities, such as disk seek and data transfers. Once the disk-level activities are determined, the workload-dependent dynamic power can be estimated.

We use *Acoustic Management*, the ability to reduce the acoustic noise of a disk drive when performing a seek operation, for controlling and budgeting disk power and energy consumption. While acoustic modes were designed to reduce the noise of the disk during a seek operation, they also reduce the instantaneous power consumption and often the energy consumption of the disk during I/O operations. Disks which support acoustic modes are in accordance with the ATA/ATAPI-6 specification [3] which defines automatic acoustic management (AAM). In this paper we emphasize the difference between power and energy. *Power* is an instantaneous measurement while *energy* is the overall power consumed over a given interval.

2 Related Work

There are several recent works on power reduction in storage systems utilizing ideas such as spinning down disks during idle time and taking advantage of caching for the purpose of increasing idle time [9], [10], [13], [16], [17], [18], [26].

A large body of research deals with multiple speed disks, also known as dynamic RPM (DRPM). While acoustic modes affect the disk-head speed, DRPM deals with the disk's rotational speed. In notable contrast to acoustic modes, there are currently no available disks that support DRPM. The works of [7], [12], [16], [18], [25] show that adapting the disk rotational speed to the required performance level can reduce power consumption.

A power simulator called *Dempsey* [24] reads I/O traces and interprets them for power and performance using DiskSim [6]. Dempsey was tested on mobile disk drives and does not take into account the effect of disk arrays. Since it requires exact traces, it cannot be used as a predictive tool. Another simulator was presented in [19], whose goal was investigating disk design optimizations for power, performance, and capacity. Stoess et al. [20] model power consumption based on disk utilization. Their model takes into account disk transfer rates and response times, but ignores the effect of seek operations on power consumption. Recently, Hylick et al. [14], [15] studied disk drive power dissipation, but they did not address storage arrays.

3 Storage Power Modeling

Our power modeling framework computes the power consumption of each storage I/O path component as it handles an

I/O operation from the time the I/o request is received and until the time the request processing is completed. The framework takes into account workloads, power states, and configurations. In addition to modeling power consumption of the I/O path, the model also takes into account power consumed by the storage components while idle. The method can be applied to single disk drives or to a storage array (e.g., a RAID array). We use the term storage controller when referring to a storage array.

3.1 The Model

It is common practice for storage controllers to report statistical performance counters (information) for each type of I/O workload operation. These operations are: sequential read, sequential write, random read, and random write. Performance counters typically include the rate of each type of operation, the transfer sizes, response time, and other statistical information. Our framework uses performance counters and differentiates between the I/O workload at the frontend of the storage controller and at the backend of the controller. The *frontend workload* refers to the I/O operations arriving from the host. The *backend workload* refers to the actual I/O operations performed by the disks. It is the backend workload which determines the power consumption of the disks.

The backend workload is affected by the read and write caching activities, by virtualization layers, and by resiliency (e.g., RAID) mechanisms. Caching activities include caching read data, performing read ahead during sequential data access (pre-fetching), and delayed (cached) writes. Caching leads to less disk activity and therefore, lower power consumption. The virtualization and resiliency layers influence the backend workload, as data can be organized into stripes across the disks in the array and write operations are translated into write transactions. For example, in a RAID 10 array, two copies of the data must be updated. Therefore, computing the backend workload from the frontend workload requires taking into account the type of operation, transfer size, data organization across stripes, etc. In order to estimate the power cost of disk I/O operations, our model calculates how many backend disk operations are needed for each type of workload. We then estimate the dynamic power cost of those backend I/O operations based on the estimated number of seeks and on the amount of transferred data.

Our framework uses a small dataset of power consumption tables, one for each storage component, to compute the dynamic power consumption. The dataset consists of power consumption values for various amounts of backend operations. For example, the dataset includes power consumption data for various amounts of data transferred and various seek rates. Building the dataset is a one-time process for each type of storage array.

The framework consists of: (i) translating frontend workloads to backend workloads; and (ii) using interpolation to estimate the power consumption of each activity, based on the pre-computed dataset. Additional details on the process can be found in [5].

3.2 Validation

We performed extensive validation runs over several types of disks and RAID configurations, using a variety of I/O access patterns and disk utilization levels. We ran various micro-benchmarks, using Iometer [1] and an industry standard SPC-1-like workload [2] to examine the accuracy of the power modeling estimations.

Disk drive results. When comparing our modeling power estimation with the actual power measured for a single disk we have observed an average modeling error of less than 3% and maximal error of 6.5% for a $15K$ $300GB$ disk. For a $10K$ $300GB$ we have observed an average modeling error of less than 5.2% and maximal error of 10%.

Disk array results. We validated our results on a RAID 5 array in a mid-range enterprise controller populated with 16 146GB 10K enterprise disks. For random read (write) workloads with transfer sizes ranging from $4K$ to $512K$ (up to $64K$, respectively) we observe a modeling estimation error of less than 5%. For larger random write transfer sizes, $128K$ to $512K$, we observe a modeling estimation error of up to 10%. We have also run SPC-1-like workloads showing a maximal power estimation error of 2.5%.

4 Power Management Using Acoustic Modes

For our investigation of acoustic modes we measured the performance and power consumption of disk drives. We use a custom-made LabVIEW [4] application for measuring the power consumption of the disk drives. Vdbench [21], a Java-based open-source tool, was used for running I/O workloads. We ran random access micro-benchmarks using Vdbench to observe the effect of the differences in seek operations in normal and in quiet acoustics.

In addition, in order to understand the effect on real-world workloads, we ran an industry standard SPC-1 workload [2]. The SPC-1 workload is a synthetic, yet sophisticated and fairly realistic, online transaction processing (OLTP) workload.

We studied a high capacity Hitachi HUA721010KLA330 1TB 3.5" disk drive, which supports acoustic modes. A full and detailed report on how acoustic modes affect the power and energy profile can be found in [8].

4.1 Power Capping

We analyzed the behavior of a seek operation in normal and quiet modes by reviewing the power profile of a single seek operation. We sampled the power dissipation of a single *long-range* seek, from one end of the platter to another, at a rate of 50K samples per second. We observed the power dissipation for the different phases of a seek operation:

Acceleration: During this phase the seek head accelerates to its maximum speed. In quiet mode the acceleration is slower, so the power dissipated at any given time throughout this phase is less than in normal mode. Only the 12V power dissipation is impacted here.

Coast: In this phase the disk head remains at its maximum speed (the maximum speed of quiet mode is slower than that for normal mode). The power dissipated at any time throughout this phase is about the same in both normal and

quiet modes. This phase lasts longer in quiet mode, causing more overall energy to be consumed per seek.

Deceleration: During this phase the disk head is slowed down by reversing the current direction of the voice coil motor (VCM). The power dissipation is generally similar to that of the acceleration phase. In quiet mode less power is dissipated at any given time.

Data transfer: During this phase, the disk head is at a complete standstill, and the data is being transferred to and from the disk. The 5V power dissipation increases, but the behavior is the same in both normal and quiet mode.

Although the power in quiet mode can be capped at 73% of the power dissipated in normal mode, the overall energy consumed by a single long-range seek operation is greater in quiet mode. For example, our analysis of 12V and 5V power consumption shows that overall energy (both 12V and 5V components) consumed by a single seek operation is 17% greater for quiet mode than for normal mode. This is due to: *i*) the fact that the duration of the long-range seek is longer, since the head moves the same distance but at a lower velocity; and *ii*) the 12V power decreases only during acceleration and deceleration (and not during coast), while the 5V power is not affected by the acoustic mode. This leads to a 5V energy consumption increase of 49% and 12V energy consumption increase of about 3%.

4.2 Energy Reduction

We now investigate the energy consumption of various workloads when using quiet mode. One effect of running in quiet mode is that moving the disk head takes longer - that is, the seek time increases. Since the disk power consumption has a static component, a seek operation that takes longer may consume more energy, depending on the balance between the saved energy of the slower acceleration and deceleration and the added energy for longer seek time.

We simulated a real-world online transaction processing workload by running SPC-1 workloads. SPC-1 is a concurrent workload composed of random reads, random writes, and sequential access across various parts of the disk drive. We generated an SPC-1 I/O trace and replayed the I/O trace in normal and quiet modes. We ran the benchmark at three I/O rates of 10, 25 and 50 I/O's per second. We measured the power consumption and computed the energy in Joules of each of the three runs. In all cases both the power consumption and the total energy consumed was lower for the quiet mode. The energy saving was between 2.2% for 10 I/O's per second and 12.54% for 50 I/O's per second. We executed I/O's at the same rate for both normal and quiet modes. At low I/O rates, 10 and 25 I/O's per second, the response time increased slightly. In these cases, when running in normal mode, the disk is in fact idle in between some I/O operations. In quiet mode, the seeks take longer, and the disk has less or no idle time. In this case, we exchange wasted disk idle time, when power is also consumed, with a longer and slower seek. Running at 50 I/O's per second results in little or no idle time, even in normal mode. The I/O requests are generated at the same rate both in normal and quiet modes. However, in quiet mode, the disk serves these requests at a slower rate, which may cause a longer queue of I/O's to form based on the fact that the response time doubles.

There are examples of workloads for which the use of quiet mode leads to an *increase* in the overall (total) energy consumption. We generated and executed a trace of $30,000$ random-read I/O's using $1, 2$, and 4 concurrent I/O threads in both normal and quiet modes. Each thread executed the I/O's synchronously without delay. We measured the power consumption and computed the energy consumption of each execution. When using less than 4 concurrent threads, the energy consumption is notably higher in quiet mode due to longer seek times. Longer seek times, in turn, lead to a longer run time. When the number of I/O threads increased to 4 we achieved a reduction in total energy. Using 4 concurrent threads we achieved an energy savings of over 2%, but when using 1 or 2 concurrent threads the energy consumption increased by nearly 6%.

4.3 Application Performance

We analyzed the impact of acoustic modes on application performance. Runing an online-generated SPC-1-like workload in both normal and quiet modes shows that in quiet mode we are able to achieve only up to 55 I/O's per second, while in normal mode we can reach more than 70 I/O's per second. At I/O rates up to 20 I/O's per second the response time in quiet mode is only slightly higher than in normal mode. However, beyond 20 I/O's per second the response time in quiet mode increases significantly and reaches, at 55 I/O's per second, almost double the response time as in normal mode.

5 Concluding Remarks

Power modeling and estimation. Our power modeling and estimation methods are based purely on performance information. Therefore, any inaccuracy in the performance data leads to an estimation inaccuracy as well. We have encountered cases where the controller failed to correctly identify the workload pattern. For example, incorrectly reporting a sequential stream as a random stream introduces errors to the estimations. Another possible source of inaccuracy is lack of information regarding background tasks (e.g., bit scrubbing, battery maintenance); better reporting of background activity will lead to improved accuracy.

Our power modeling can be used in a power-aware capacity planning tool predicting the power consumption based on the given configuration and workloads. Our modeling can also provide online power estimations, per disk array and disk volume, for storage systems.

Power and energy management using acoustic modes. We have explored the effects of acoustic management on performance and power consumption. While acoustic management can in some cases be applicable for energy savings, it is always effective for power capping (or budgeting).

Quiet acoustic modes change the way disks perform seek operations, so there is no power reduction when no seeks are performed, for example, during idle time or during sequential access. Since only seeks are affected, the power for

the electronics remains the same. This limits the ability of acoustic modes to save power. For random-read workloads the power reduction is at most 23%, depending on the actual I/O workload.

Quiet mode causes an increase in response time. This prevents the use of quiet mode for mission-critical applications that are sensitive to I/O response time. Single-threaded applications that require high throughput will suffer a 25% reduction in I/O throughput. Moreover, they will consume more energy in quiet mode than in normal mode, but will benefit from a lower peak power consumption. Multi-threaded applications with a mixed workload of read and write operations, both random and sequential, will be able to sustain the same I/O throughput, but with longer response time. Such applications may need to use a larger number of threads while using quiet mode, in order to sustain the same I/O throughput as in normal mode.

We have found that in some cases, seek operations consume more overall energy in quiet mode than in normal mode, though they consume less instantaneous power. We have also encountered workloads for which quiet mode leads to energy savings. The SPC-1 workload tests clearly demonstrate that OLTP applications are good candidates for energy savings, when they can tolerate a degradation in response time.

REFERENCES

[1] "Iometer, performance analysis tool." http://www.iometer.org/.

[2] "Storage performance council," http://www.storageperformance.org/.

[3] "INCITS 361-2002 (1410D): AT attachment - 6 with packet interface (ATA/ATAPI - 6)," 2002.

[4] "LabVIEW release notes," 2009, http://www.ni.com/pdf/manuals/371778e.pdf.

[5] M. Allalouf, Y. Arbitman, M. Factor, R. Kat, K. Meth, and D. Naor, "Storage modeling for power estimation," in *Proceedings of SYSTOR 2009: The Israeli Experimental Systems Conference*, 2009.

[6] J. S. Bucy, J. Schindler, S. W. Schlosser, G. R. Ganger, and Contributors, "The DiskSim simulation environment - version 4.0 reference manual," May 2008.

[7] E. V. Carrera, E. Pinheiro, and R. Bianchini, "Conserving disk energy in network servers," in *Proceedings of the 17th Annual International Conference on Supercomputing*, June 2003, pp. 86–97.

[8] D. Chen, G. Goldberg, R. Kahn, R. I. Kat, K. Meth, and D. Sotnikov, "Leveraging disk drive acoustic modes for power management," in *MSST'10 Research Track: Proceedings of the 26th IEEE Conference on Mass Storage Systems and Technologies (MSST2010): Research Track*, 2010.

[9] D. Colarelli and D. Grunwald, "Massive arrays of idle disks for storage archives," in *Proceedings of the 2002 ACM/IEEE conference on High Performance Networking and Computing*, November 2002, pp. 1–11.

[10] F. Douglis, P. Krishnan, and B. Bershad, "Adaptive disk spin-down policies for mobile computers," in *Proceedings of the 2nd USENIX Symposium on Mobile and Location-Independent Computing*, April 1995, pp. 121–137.

[11] EPA, "Epa report to congress on server and data center energy efficiency," *Public Law 109-431*, 2007.

[12] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir, and H. Franke, "DRPM: Dynamic speed control for power management in server class disks," in *Proceedings of the 30th Annual International Symposium on Computer Architecture*, June 2003, pp. 169–181.

[13] S. Gurumurthi, J. Zhang, A. Sivasubramaniam, M. Kandemir, H. Franke, N. Vijaykrishnan, and M. J. Irwin, "Interplay of energy and performance for disk arrays running transaction processing workloads," in *Proceedings of the International Symposium on Performance Analysis of Systems and Software*, March 2003, pp. 123–132.

[14] A. Hylick, A. Rice, B. Jones, and R. Sohan, "Hard drive power consumption uncovered," *SIGMETRICS Performance Evaluation Review*, vol. 35, no. 3, pp. 54–55, 2007.

[15] A. Hylick, R. Sohan, A. Rice, and B. Jones, "An analysis of hard drive energy consumption," in *MASCOTS*, 2008, pp. 103–112.

[16] X. Li, Z. Li, F. David, P. Zhou, Y. Zhou, S. Adve, and S. Kumar, "Performance directed energy management for main memory and disks," in *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems.* ACM Press, 2004, pp. 271–283.

[17] D. Peek and J. Flinn, "Drive-thru: fast, accurate evaluation of storage power management," in *ATEC '05: Proceedings of the annual conference on USENIX Annual Technical Conference.* Berkeley, CA, USA: USENIX Association, 2005, pp. 30–30.

[18] E. Pinheiro and R. Bianchini, "Energy conservation techniques for disk array-based servers," in *Proceedings of the 18th Annual International Conference on Supercomputing*, June 2004, pp. 68–78.

[19] S. Sankar, Y. Zhang, S. Gurumurthi, and M. R. Stan, "Sensitivity-based optimization of disk architecture," *IEEE Trans. Comput.*, vol. 58, no. 1, pp. 69–81, 2009.

[20] J. Stoess, C. Lang, and F. Bellosa, "Energy management for hypervisor-based virtual machines," in *ATC'07: 2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference.* Berkeley, CA, USA: USENIX Association, 2007, pp. 1–14.

[21] H. Vandenbergh, "Vdbench 5.00 users guide," 2008, http://garr.dl.sourceforge.net/project/vdbench/vdbench/Vdbench%205.00/vdbench.pdf.

[22] R. Villars, "Three keys for storage success: Content, architecture and getting personal. IDC," July 2007.

[23] S. W. Worth, "SNIA green storage tutorial," 2007, http://www.snia.org/forums/green/programs/SWWorth_Green_Storage.pdf.

[24] J. Zedlewski, S. Sobti, N. Garg, F. Zheng, A. Krishnamurthy, and R. Wang, "Modeling hard-disk power consumption," in *FAST '03: Proceedings of the 2nd USENIX Conference on File and Storage Technologies.* Berkeley, CA, USA: USENIX Association, 2003, pp. 217–230.

[25] Q. Zhu, Z. Chen, L. Tan, Y. Zhou, K. Keeton, and J. Wilkes, "Hibernator: helping disk arrays sleep through the winter," in *Proceedings of the Symposium on Operating Systems Principles (SOSP)*, October 2005, pp. 177–190.

[26] Q. Zhu, F. M. David, C. F. Devaraj, Z. Li, Y. Zhou, and P. Cao, "Reducing energy consumption of disk storage using power-aware cache management," in *HPCA '04: Proceedings of the 10th International Symposium on High Performance Computer Architecture.* Washington, DC, USA: IEEE Computer Society, 2004, p. 118.

SMOA Devices - A Distributed Management And Control System For Energy Consumption In Computing Environments

Krzysztof Kurowski
Poznań Supercomputing and Networking Center
Applications Department
ul. Noskowskiego 10
61-704 Poznań, Poland
krzysztof.kurowski@man.poznan.pl

Ariel Oleksiak
Poznań Supercomputing and Networking Center
Applications Department
ul. Noskowskiego 10
61-704 Poznań, Poland
ariel@man.poznan.pl

Michał Witkowski
Poznań Supercomputing and Networking Center
Applications Department
ul. Noskowskiego 10
61-704 Poznań, Poland
michal.w.witkowski@gmail.com

ABSTRACT

As the notion of Green Computing becomes more popular in the IT world, new power saving features start to appear in both computer hardware and software. The difficulty of leveraging existing means of power management and control increases with the scale of operations, making a networked approach to the problem a necessity. In this paper we present the SMOA Devices architecture, a distributed solution for monitoring and management of power consumption in computer systems. We discuss its features and characteristics, and explain what makes it suitable for a broad range of power saving and management applications.

General Terms

MANAGEMENT, MEASUREMENT

Keywords

power management, XMPP, power consumption, energy efficiency, distributed systems

1. INTRODUCTION

The rising energy prices and the increasing environmental awareness of the general public are playing an ever-increasing role in the world of IT and computational science. After years of constructing computer systems solely for their sheer performance, power consumption became one of the main criteria for evaluating a computer system and metrics, such as performance-per-watt [13], are receiving increasing attention.

The push for increased energy-efficiency is also present in the field of desktop computing and even home appliances. All major modern operating systems feature sophisticated power management capabilities based on open industry standards [9]. While these features can reduce the power consumption of a running machine, they do not address the problem of idling computers, which consume vast amounts of power without any gain in productivity. The issue can be mitigated by suspending or shutting down idling machines, although such solution entails the problem of waking the machines up afterwards.

Even the smallest power consumption reduction, achieved using the techniques already available in modern computer hardware and software, can cause enormous net-savings on company or cluster scales. However, introducing a power saving policy on a larger scale requires a sophisticated distributed mechanism of management and control, as well as a metering solution to evaluate its impact.

In this paper, we present an easily extensible, flexible and standards-based distributed management and control system for energy consumption. The paper is organized as follows. Section 2 deals with our proposed solution, while section 3 is a short description of its possible application scenarios. Section 4 discusses related work and section 5 deals with future work related to the system. In the final section, we conclude the description of our proposal.

2. SMOA DEVICES

This section describes our power consumption monitoring and control architecture, the SMOA Devices platform. The main motivation for its creation was to combine various power management related tools, solutions and monitoring equipment using a standardized, easy-to-deploy network protocol.

2.1 Usage of XMPP

The eXtensible Message and Presence Protocol (XMPP)[17] is best known for its application as the protocol backbone for the Jabber [6] and Google Talk [3] instant messaging services. However, because of its characteristics, it is well suited for areas other than chatting. The main feature of this protocol is that it is based upon a persistent client-server connection, making it ideal for application in heavily firewalled environments where opening local network ports is impossible. Moreover, XMPP heavily relies on XML[12] for all communications, eventually turning the client-server con-

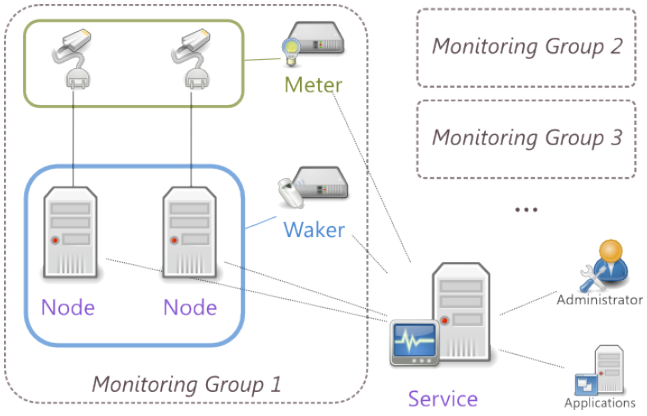


Figure 1: SMOA Devices Architecture

nection into two continuous XML document streams, one for each communication side. This feature makes XMPP easily-extensible and well-suited for transportation of other XML formats. Since XMPP was first created as an IM protocol, it also provides authentication and authorization capabilities based on SASL [15], which coupled with SSL/TLS encryption of the client-server connection, can be used to establish trust relations among nodes connected to the network.

2.2 RESTful XMPP

Since the architecture of Representational State Transfer (REST) was first introduced in the doctoral dissertation of Roy Fielding [14], resource-oriented approach to constructing web services is on the rise. Because such approach is natural for representing measurement data and device states, it became the data-access paradigm employed in the SMOA Devices architecture.

With HTTP being the predominant protocol for implementation of REST services, it seemed reasonable to mimic its behavior in XMPP with a protocol that is a mapping of HTTP logic to XML data structures. This approach has also an additional advantage of SMOA Devices elements being easily extensible to incorporate a HTTP transport in the future.

2.3 Architecture overview

The SMOA Devices architecture (Figure 1) integrates various tools and solutions across different levels ranging from integrated measuring devices to OS power management policies. Each of them is represented in the SMOA Devices platform as a separate *Node* having its own JID (Jabber ID) in the XMPP network. Every *Node* plays one of three roles: *Device* (machine) node, *Waker* node or *Meter* node. The first one is responsible for interacting with the machine's operating system, providing information about CPU load, memory consumption, power management policies, and providing means for safe shutdown/suspend of the system. The *Waker* node is responsible for bringing up normal nodes after their shutdown by the means of Wake-on-LAN [16] or IPMI [5]. *Meter* nodes are interfaces between the SMOA Devices platform and the power measurement/control hardware, providing mappings between JIDs and outlets the machines are connected to. Both the waker and meter nodes are responsible for handling groups of nodes, for example wake up of all nodes in one network segment. The *SMOA Devices Service* is the central point of the architecture integrating the capabilities provided by *Nodes, Wakers* and *Meters*, so that the management functions and information about a physical machine are provided in a coherent way, regardless of their source.

2.4 The software

The software components of the SMOA Devices architecture need to be highly portable and easy-to-extend, thus Python was chosen as the language of implementation. The *Node* code is available for all major platforms (Windows, Linux and Mac OS X), providing the same core functionality regardless of the OS involved. All components are highly modular, with each featuring the concept of a *handler*. Each *handler* provides some dedicated functionality, e.g. supplying information about CPU load or sending Wake-on-LAN requests. *Handlers* can be easily added or swapped-out to adapt the platform to specific scenarios. For example, in server environment one might want to write a *handler* to provide *Waker* and *Meter* capabilities using vendor-supplied solutions, such as IBM BladeCenter Advanced Management Module.

The platform's software also features a simple notification mechanism similar to a push-pull model, allowing software components or users to subscribe to various information sources. The information in these sources can be provided either periodically (e.g. power consumption statistics) or event-based (e.g. alarms). In both cases, XMPP provides the perfect protocol for such uses. For end-user interaction SMOA Devices packages contain a dedicated Pidgin [7] plugin (Figure 2). The aim of the plugin is to blur the gap between management tools and end-user applications, thus all managed machines are visible as "buddies" on the contact list. Since each machine has a dedicated JID, its "on-line" state represents its power state. The contact's status message contains the hostname for easier identification. The plugin also allows the user to change the powerstate of the machines via right-click menu items. While its functionality is currently limited to power-cycling machines and displaying state, the plugin will be enhanced to bring the management functionality closer to the desktop.

2.5 The hardware

Since the ideals of Green IT had only recently gained the interest of the computer hardware market, most of current computers lack any form of power consumption monitoring components. Although some blade-based solutions or smart server chassis provide such functionality, they are still rare. This is unfortunate because the ability to measure the savings provided by a power management policy is crucial to the evaluation of its viability.

One way of solving the problem is by using intelligent power strip solutions available on the market nowadays. These devices provide at least 8 separately and remotely managed power outlets. While the low-end models only measure total power consumption, more expensive units are equipped with per-outlet watt-meters. By USB, serial or Ethernet connectivity these devices can be interfaced by *Meter* nodes

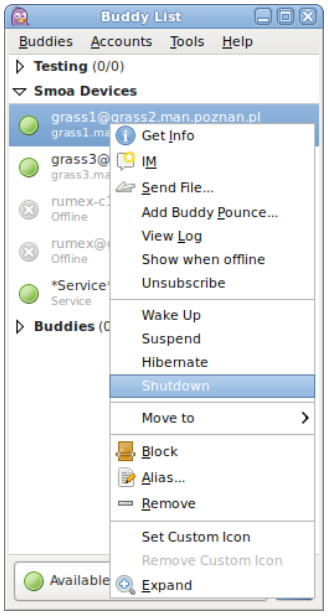


Figure 2: SMOA Devices Pidgin plugin in action

to provide power monitoring and control functionality for machines managed by the SMOA Devices platform.

An alternative is to use a dedicated solution developed for the SMOA Devices called *e-Sockets*. *e-Sockets* are small embedded devices containing a power outlet management chip, with watt-meter and on/off capabilities, and a low-power ZigBee [18] wireless component for communication. The *e-Sockets* are connected in a mesh, self-healing network with permitted node spacing of up to 15m. Of course, such a network of *e-Sockets* need to be controlled and interfaced by a control device. An ARM-powered [11] embedded computer with a ZigBee and Ethernet adapters is suitable for such a task. Thanks to the portability of Python, the *Meter* node can be run on the device itself, providing the SMOA Devices platform with access to up to 1000 *e-Sockets*.

3. APPLICATION SCENARIOS

While there is a significant effort to reduce the power consumption of individual components of computer systems [10], the most savings can be achieved by powering off computers altogether. As far as economy is concerned, an idling computer consumes quite a lot of power (above 100W for a standard PC) without producing much value. Even computers (or other appliances) that are turned off, but still connected to mains, draw a certain amount of power that may be significant in deployments of a larger scale. This chapter will show how SMOA Devices can be used in different environments and scenarios to counter these problems.

3.1 HPC environment

HPC environments consist of high-end computer equipment consuming vast amounts of power. The computing capacity provided by such deployments is usually managed by job schedulers, while system administrators manage the machines themselves. Though some effort has already been taken to address the problem, rarely are job schedulers aware of the power consumption impact of the jobs they launch. This often leads to situations when some machines are not being utilized to their full capacity or are even completely idling. SMOA Devices can be used by job schedulers to automatically bring up or down machines based on current computational needs and make advanced scheduling decisions by taking into account performance per watt metrics, thanks to live power consumption metering.

Thanks to its extensibility, our platform can take advantage of existing out-of-band management solutions provided by vendors for cluster environments. SMOA Devices can integrate their power-cycling and, if available, power-metering capabilities. Supplemented by the deployment of intelligent power strips, our platform can provide a coherent and uniform solution for power management and monitoring of cluster machines. By providing event notifications and an intuitive interface, SMOA Devices can bring management functionality closer to cluster administrators' desktops, making their task of managing the system easier.

3.2 Managing desktop machines in office environments

Computers became a commodity in modern business environment, with HPC being only one of their specialized applications. Much more common are desktop computers, present in virtually every company office, ranging from small businesses to large corporations. Since they are used as office tools for regular employees, and not execution environments for batched processing, their performance capabilities are rarely exploited to their full extent. This gives room for improvements in power consumption by fine-tuning OS power management settings. Taking into account the number of desktop computers in modern offices, this can yield a significant net power consumption reduction. SMOA Devices, being portable across all major desktop platforms, can provide a unified interface to control power management policies in an office environment.

IT staff can also take advantage of our platform's computer power-cycling capabilities to reduce power consumption by suspending machines which are left on by employees during off-office hours. Thanks to an innovative application of the popular Wake-on-LAN, this functionality of SMOA Devices can be used with little regard of the available network configuration. Given the general habits of office workers and the big difference in power consumption of an idling computer as compared to a suspended one, this can greatly influence the company's power bill. The solution can be further extended to gracefully shut down the unused machines and cut them off, along with their peripherals, completely from the mains thanks to functionality of *e-Sockets*. By doing so, the power draw related to sleep modes of all computer devices can be eliminated. While this amount of saving can be irrelevant for a single machine, it may be significant at the office or office building scale.

3.3 Domestic use

The SMOA Devices architecture can also be applicable to domestic power management and monitoring. In this scenario, a home owner could be interested in automated control of his/her appliances and reducing his/her power con-

sumption and bills. The provider of the power management and monitoring service would be a business interested in gathering fine-grained information about power consumption in given municipal area or among given type of users. It could be either the power distribution company, interested in spreading power consumption equally through the day to avoid peaks, or a company interested in statistics for marketing reasons.

By using *e-Sockets*, key energy consuming devices of the household can be monitored and controlled. Since the *e-well Sockets* require a dedicated control and data gathering device, a small ARM-equipped computer fitted with a SMOA Devices Node daemon would be provided to home owners. The devices can be connected to the home owner's Internet connection, in which case the client-server nature of the XMPP protocol would play a key role. These devices can come with a pre-programmed JID account set up by the provider's XMPP server and a complementary JID would be provided to the home owner. This way, the home owner could use his/her standard Instant Messaging client (Pidgin) to control and monitor the devices in his/her house and schedule their power cycles, e.g. turn off the home theater system in the middle of the night to avoid sleep mode power consumption. The home owner's JID could also be used as a communications channel for technical support or tips about reducing the power consumption and power costs.

4. RELATED WORK

In recent years, with the increasing environmental awareness of the society, there has been many initiatives aiming to reduce power consumption in different scenarios. To the best of our knowledge, none of these attempts were as flexible and extensible as SMOA Devices in covering so many different application scenarios.

Google PowerMeter [2] is aiming to bring the power monitoring information to the Web. It is meant to display the information from their utility smart-meters and in-home energy management system on the user's start page, and facilitate services for comparing and sharing information about one's power consumption.

The AlertMe [1] product range is a set of power consumption sensors similar to our *e-Sockets* communicating wirelessly via ZigBee to a hub connected to the home owner's Internet connection. The hub submits information to the AlertMe servers, via which a service of monitoring and scheduling is provided.

The Green-NET [4] explores the design of energy-aware software frameworks dedicated for large-scale distributed systems. The project aims to provide a framework which would collect energy usage information and provide them to other applications, such as resource managers or job schedulers.

The Oxford University's monitoring and wake on LAN services [8] is a project targeting the problem of idling computer nodes mainly during off-office hours. The system is intended as a monitoring infrastructure measuring the number of online nodes in each department or college, coupled with a service for remote wake-up using Wake-on-LAN.

5. FUTURE PLANS

SMOA Devices has already been deployed on a limited scale for testing purposes. Though the framework currently supports its basic functionality (power management, remote wake-up), in the near future features such as IPMI integration, *Node* group management and HTTP/WWW interfaces will be introduced. To be able to assess the power saving impact of the framework, larger scale deployments are planned in both office and HPC environments. For future production deployments, plans to enhance scalability and fault-tolerance of the system are made. These include *Nodes* being associated to more than one *Service*, increasing scalability by spreading the system load over many XMPP servers and increasing reliability due to XMPP server replication.

6. CONCLUSIONS

In this paper, we have presented the SMOA Devices architecture, a flexible and extensible distributed power monitoring and control solution. SMOA Devices platform faces the challenge of integrating methods and tools available on different abstraction levels ranging from power outlet control, through operating system calls, to networked remote wake-up. This kind of vertical integration is a novelty, which combined with easily-deployable communication protocol and wide-ranging monitoring and control capabilities, make SMOA Devices a well-suited solution for many application scenerios in the upcoming years of increased environmental awareness and the green computing revolution.

7. REFERENCES

[1] AlertMe Smart Energy. http://www.alertme.com/.

[2] Google PowerMeter. http://www.google.org/powermeter/.

[3] Google Talk. http://www.google.com/talk/.

[4] GREEN-NET Project. http://www.ens-lyon.fr/LIP/RESO/Projects/GREEN-NET/.

[5] Intelligent Platform Management Interface Second Generation Specification.

[6] Jabber.org. http://www.jabber.org/.

[7] Pidgin. http://pidgin.im/.

[8] The monitoring and wake on LAN services, Low Carbon ICT. http://www.oucs.ox.ac.uk/greenit/wol.xml.

[9] Advanced Configuration and Power Interface Specification, 2009. http://www.acpi.info/spec.htm.

[10] ENERGY STAR Program Requirements for Computers, 2009.

[11] ARM Ltd. ARM Processor Core Overview. 2008.

[12] T. Bray, J. Paoli, C. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible markup language (XML) 1.0. *W3C recommendation*, 6, 2000.

[13] CompuGreen, LLC. The Green500 List.

[14] R. Fielding. *Representational state transfer (REST). Chapter 5 in Architectural Styles and the Design of Networkbased Software Architectures*. PhD thesis, Ph. D. Thesis, University of California, Irvine, CA, 2000, 2000.

[15] J. Myers. Simple authentication and security layer (SASL). 1997.

[16] J. Oliveira. Wake on Lan mini-HOWTO.

[17] P. Saint-Andre et al. Extensible messaging and presence protocol (XMPP): Core. 2004.

[18] Zigbee Alliance. Zigbee specification. *ZigBee Document 053474r06, Version*, 1, 2005.

Characterization of energy consumption and energy efficiency

Alberto E Garcia, Karin A. Hummel
Chairs of IC0804 Working Group 2

I. Introduction

Characterization of energy consumption and energy efficiency are the main priorities of the Working Group 2 included into the IC0804 Cost Action. This Working Group proposes to find appropriate definitions for energy efficiency and to investigate the energy consumption and efficiency for system components (like routers, PCs, clusters, servers) to the whole distributed systems (taking into account the potential influence of one part to another). The relationship between resources sharing and energy consumption are evaluated establishing the corresponding cost models (including euro-costs), introducing a measurable metric to establish adaptive solutions.

One of the first results of the collaboration of different involved laboratories is the identification of key terms and keywords related to the different areas covered by the concept of energy efficiency and their relationships within the four main priorities above. The Fig.1 shows how the different terms may be associated with more than one of the main lines of action. It is also true that we only have taken into account the more obvious relationships, since in most cases a single term can be used in any of the lines by simply changing the context of its application.

During our previous Cost action meetings, involved laboratories made a presentation of their main characteristics, about their researching side, including their most important current researches. From a closer study of the different contributions, we identify four broad areas of study directly related to the research and keywords currently considered:

1. Networking: It is easily divided into two actual biggest worlds, wireless environments and wired one.
2. Architectures: It is a very generalist subject because it includes several differentiated issues depending of the network application. It includes subjects related with applications (e.g. virtualization, content distribution), and communications (e.g. P2P & DataCenter policies)

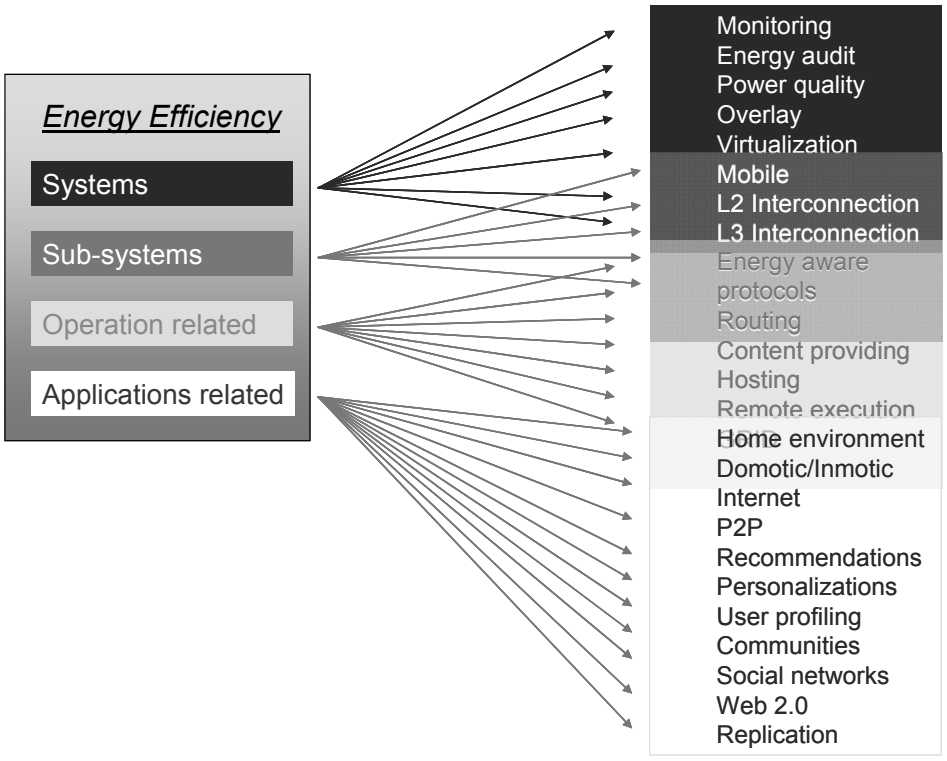


Fig. 1. Energy efficiency in large scale distributed systems: Keywords and main terms

3. Middleware/data management: These terms are more related with solutions that could be applied into the architectures and/or networking implementations, but they could be independently solved
4. Applications: These studies could be directly related to previous because they have to consider one or several associated environmental conditions. Energy awareness and efficient consumption are the main keys of the studies, including both mobile and wired environments

These four broad areas are independent but their interrelationship could be considered, with common areas of research, showed in Fig. 2.

The following documents are some of the most representative examples of work currently being developed within the COST action in relation to research issues associated with WG2. These studies represent the spearhead of the efforts being made by different laboratories during the first year of action. The joint work are starting to get results and new initiatives are leading to proposals for collaboration with other laboratories, companies and international projects within and outside the COST.

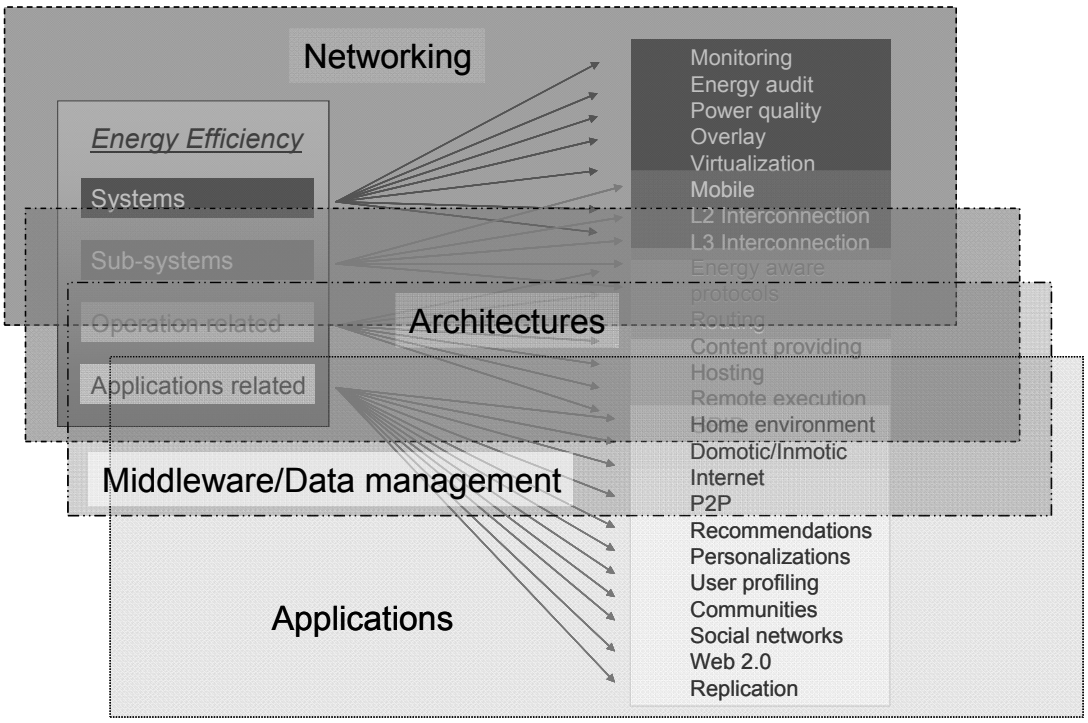


Fig. 2. Main areas of study inside the WG2

Green Internet File Sharing through Energy Efficient BitTorrent

Giuseppe Anastasi[*], Marco Conti[**], Ilaria Giannetti[*], Andrea Passarella[**]

[*]Dept. of Information Engineering, University of Pisa
Via Diotisalvi 2, 56122 Pisa, Italy
E-mail: {giuseppe.anastasi, ilaria.giannetti}@iet.unipi.it

[**]IIT-CNR
Via G. Moruzzi, 56124 Pisa, Italy
E-mail: {marco.conti, andrea.passarella}@iit.cnr.it

Abstract - **It is a common experience that many users leave their Personal Computer powered on for very long times to run Peer-to-Peer (P2P) file sharing applications. Since a large fraction of nowadays Internet traffic is originated by P2P applications, a significant energy reduction can be achieved increasing the energy efficiency of these applications. In this chapter we focus on BitTorrent – currently the most popular P2P platform – and propose EE-BitTorrent, a proxy-based version of the protocol optimized for energy efficiency. In EE-BitTorrent users delegate the download operations to a proxy, which can serve a large number of users, and switch off their PC during the download. We implemented and evaluated our solution in a real testbed. Our results show that, with respect to the legacy approach, EE-BitTorrent can provide up to 95% energy savings while reducing, at the same time, the average download time.**

I. INTRODUCTION

Many different studies show that the total energy consumption associated with the Internet is already very high and is expected to increase even more in the next years, as the Internet role in the society will expand. About 74 TeraWatts hours (TWh) of electricity per year are consumed in USA by Internet equipments (data centers, routers, switches, end hosts, etc.) [1]. This accounts for about 2% of the global electricity consumption in USA [2, 3]. Although the percentage is not so relevant, it nevertheless corresponds to $6 billions in energy cost and the emission of more than 50 million metric tons of CO2 per year [3]. In addition, it is estimated that roughly one third of this energy is wasted and could be saved using simple power management techniques on Internet-connected devices [1].

Most of the energy is consumed on the Internet edges, i.e., data centers and personal computing devices [4]. Therefore, the research efforts are currently concentrated on reducing the energy consumption in data centers and user equipments. In particular, Personal Computers (PCs) are largely widespread and very numerous. Typical power consumptions for desktop PCs and notebooks are in the order of 100 and 30 kW. In USA in 2007 office and home PCs accounted for approximately 16 TWh per year [4].

Furthermore, PCs are usually managed by common users who are not eager to address the energy wastage problem. Therefore, generally, they not apply any power management policy and very often leave their PC continuously powered on. This clearly emerges, for example, from the PC Energy Report released by the UK National Energy Foundation [5], which shows that about 21% of the PCs used at work are never switched off, thus causing an energy wastage of about 1.5 TWh of electricity per year, corresponding to about 700,000 tons of CO_2. This energy wastage could be easily avoided by just switching off PCs, e.g., using a centralized shutdown solution such as the NightWatchman [6].

However, in many cases PCs are intentionally left on by their users, especially at home, to perform networking activities like, for example, Peer-to-Peer (P2P) file-sharing. Recent studies [7] indicate that a very large fraction of nowadays Internet traffic is P2P (40-73%), and BitTorrent is the most popular P2P platform accounting for 50-75% of the overall P2P traffic. Hence, focusing on energy efficient P2P solutions is a very sensible research direction towards an energy-friendly Internet.

In this chapter we present ideas and concepts for saving the energy of PCs running a P2P application, customized for the BitTorrent platform. Specifically, we propose Energy Efficient BitTorrent (EE-BT), a proxy-based version of BitTorrent optimized for energy savings. Even if the proposed solution is customized to the BitTorrent platform, the ideas and concepts presented here are general enough and can be easily extended to other P2P platforms as well.

The rest of this chapter is organized as follows. Section II discusses the main approaches to energy efficiency. Section III presents the legacy BitTorrent protocol while Section IV introduces the EE-BitTorrent protocol. Section V provides some experimental results showing the effectiveness of the proposed solution. Finally, Section VI concludes the chapter.

II. GENERAL APPROACH TO ENERGY EFFICIENCY

The energy consumption by the Internet can be subdivided in two broad categories, i.e., *direct energy* and *induced energy* consumption [8]. Direct energy consumption accounts for the energy consumed by network equipments (i.e., routers and switches) and links, while *induced* energy consumption represents the energy consumed by end devices (e.g., PCs, printers, displays, etc.) to correctly implement the network protocols (e.g., to respond to an ARP request).

Solutions for reducing the direct energy consumption are thus aimed at reducing the energy consumption of links, routers and switches. Link power-management techniques adjust the link rate according to the real traffic needs. For example, the power consumption of an Ethernet network

interface card (NIC) increases from 1W for 10/100 Mb/s, to 7W for 1 Gb/s up to 15 W for 10 Gb/s [9]. Therefore, a large amount of energy can be saved by using the lowest link rate compatible with traffic conditions. The idea is known as Adaptive Link Rate (ALR) [9], Rapid PHY Selection (RPS) [10], Dynamic Link Control [11] or Energy Efficient Ethernet (EEE) [12].

Power management of network equipments relies on NICs with different power modes (from completely sleeping to completely active) and consists in switching the NIC between different power levels, depending on the network activity, using a sleeping algorithm, such as the Dynamic Ethernet Link Shutdown (DELS) [13, 14, 15].

Although the above-mentioned techniques for power management of links and network equipments can provide some energy savings, they can be used only when a NIC with appropriate hardware support is available. In addition, they do not seem the best approach for P2P file sharing, where downloading a file can take even several hours. Actually, the major energy consumption related with P2P file sharing is induced energy consumption due to end hosts (typically PCs) that must remain active for long times.

Solutions addressing the induced energy consumption include *remote wakeup* and *proxy-based* techniques. These techniques - often used jointly - rely on the experimental evidence that most of packets received by an end host (e.g., a PC) require a routine response or can be even ignored. Hence, the end host can be put in sleep mode, for energy conservation, while connectivity can be managed by a Network Connectivity Proxy (NCP). This allows to combine energy efficiency with permanent connection to the Internet [16, 17, 18].

An NCP is an entity capable of maintaining the network presence on behalf of a sleeping host, managing all packets destined to it. In practice, whenever receiving a packet the NCP performs one of the following actions, depending on the packet type [19]: (i) discards the packet; (ii) directly responds to the packet; (iii) re-directs the packet to another (active) computer for further processing; (iv) queues the packet for deferred processing by the host when it wakes up; or (v) wakes up the sleeping host and passes it the packet for appropriate processing. The NCP requires a remote wakeup mechanism on the sleeping host to wake up it when necessary, e.g., a Wake On LAN (WoL) NIC [16]. The latter is a special NIC with auxiliary source power, an external wakeup signal and the capacity to recognize wakeup packets in auxiliary power. The NCP can be implemented either as part of the computer's NIC [17], or as an external entity (e.g., a USB-connected device [20], or a software module running on a router [19], switch [17] or separate computer [16, 21, 22] in the same LAN).

NCPs provide a general framework for saving energy in Internet-connected PCs during idle periods, but they're not specifically tailored to P2P applications. An alternative approach is using a proxy customized to a specific application (e.g. P2P file sharing). This is the approach taken in this chapter where we propose a proxy-based version of the well known BitTorrent protocol, optimized for energy efficiency.

Energy efficiency in BitTorrent is also the main goal of Green BitTorrent [23]. It is a modified version of the legacy BitTorrent protocol where peers, that have already completed their download process and are not involved in any upload operation, are allowed to put their PC in sleep mode. From the viewpoint of a generic peer, the other peers in the same swarm can be in one of the following states: *connected*, *sleeping*, and *unknown*. When the number of connected peers is less than a pre-defined threshold, a peer can explicitly wakeup a sleeping peer by sending a special wakeup message to it. Unlike Green BitTorrent our proposal does not require any special hardware and introduces only small modifications in the BitTorrent protocol.

III. LEGACY BITTORRENT

BitTorrent implements an unstructured overlay network customized for file sharing [24]. Nodes of the overlay are called *peers* and the collection of peers involved in the distribution of a given file is called a *torrent* or *swarm*. The basic idea of BitTorrent is that peers both download and upload *chunks* of the shared files. Therefore, each peer receives chunks of a given file from a multitude of other peers, instead of downloading it from a single server as in the conventional client-server model. The resulting capacity of such cooperative process is higher than that of the traditional client-server architectures [24, 25].

We provide below a short description of the BitTorrent protocol (a detailed description can be found in [26]). A peer wishing to download a file first needs to get the corresponding `torrent` file. Torrents are very small files, typically hosted by conventional Web servers (torrent servers), and can be found through standard Internet search engines. A torrent file contains the name of the file's *tracker*. This is a node that constantly tracks which peers have chunks of the file (i.e., belong to the swarm). When a peer joins a swarm it registers with the tracker and, then, periodically informs the tracker that it is still in the swarm.

Once obtained the tracker's address, the peer opens a TCP/IP connection to the tracker and receives a random list of peers to be contacted for starting the download process. Then the tagged peer gets in touch with a set of peers, called *neighbors*, with which it exchanges parts of the file. The neighbor set changes dynamically since, as time elapses, some peers may leave the swarm and others may join. In addition, each peer preferentially selects, for downloading chunks, those peers from which it can achieve the highest download rate (see below). Furthermore, every 30 seconds neighbors are selected completely at random, as a way to discover new neighbors and allow new peers in a swarm to start-up.

At a certain point in time, each peer in the swarm has a different subset of chunks from the file. To figure out where missing chunks can be downloaded from, periodically the peer asks each of its neighbors for the list of chunks they have. To decide which chunks to request first the tagged peer uses the

Rarest First policy i.e., it gives priority to those chunks that are less spread. Finally, to decide which requests from other peers to respond to, the tagged peer uses the *Tit-for-Tat* (TAT) policy, i.e., it gives priority to peers from which it is downloading data at the highest rate. For each of its neighbors the tagged node measures the downloading rate and, then, selects the four peers that are providing to it the highest bit rate.

IV. ENERGY EFFICIENT BITTORRENT

The legacy BitTorrent architecture is not energy efficient. BitTorrent peers have to stay connected to the overlay network during the whole download process of the requested files, which, typically, may take several hours. Hence, the induced energy consumption can be very high. Periodically turning off peers without modifying the BitTorrent architecture is not a viable solution for several reasons. First of all, if a peer is downloading content, powering it off does not save any energy (related to the current download), as the download itself stops when the peer turns off. Also, powering off peers that are not downloading anything (but are sharing content) is also not an efficient solution in general, as this can result in decreasing the overall download performance of the swarms they participate to. Thinking at coordinated ways of powering those peers is also not appropriate, as it would require central control, and is thus at odds with the BitTorrent P2P design paradigm.

To increase the energy efficient of P2P file sharing, we propose EE-BitTorrent, a proxy-based version of the traditional BitTorrent protocol. We refer to a typical LAN environment where a certain number of users run BitTorrent peers on their PC. Our solution just requires that one computer in the LAN behaves as a proxy between the peers and the rest of the BitTorrent overlay. The proxy can either be a dedicated computer, or a machine that is already providing other network services (e.g., DHCP server, Web proxy, etc.). Obviously, if the BitTorrent proxy is a dedicated machine we need to take into account its additional energy consumption. Conversely, if the same machine is already used for other purposes, there is no additional consumption.

The basic idea of EE-BitTorrent is that peers "behind" the BitTorrent proxy ask the proxy itself to download the requested content on their behalf. The proxy participates to the conventional BitTorrent overlay, and takes care of all the downloads of the peers behind it. While downloads are in progress, the peers behind the proxy can be switched off without stopping the requested downloads. Finally, the requested files are transferred from the proxy to the peers upon completion.

We provide below a brief description of the different actions performed in EE-BitTorrent, by the various actors, during the file download process. A more detailed description of the EE-BitTorrent architecture and protocol can be found in [27].

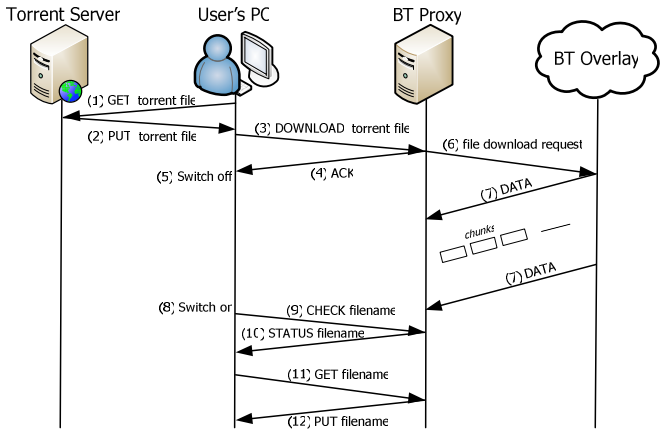


Fig. 1. EE-BitTorrent Protocol

With reference to Fig. 1, whenever a user wishes to download a new file, the client program running on his/her PC retrieves the `.torrent` file from a torrent server in the Internet, just like in the conventional BitTorrent architecture (steps 1-2). Then, it uploads the `.torrent` file to the BitTorrent proxy requesting the download of the desired file (step 3). The proxy acknowledges the received request (step 4) and starts the download operations according to the standard BitTorrent protocol (steps 6-7). Upon receiving an acknowledgement from the BitTorrent proxy, notifying that the download has started, the user's PC can be switched off to save energy (step 5). Later, on restart, the PC checks the status of all file downloads previously requested to the proxy (steps 8-10). If the download is over, the PC fetches the corresponding file from the proxy (steps 11-12). Since this file transfer is between two computers connected to the same LAN, it takes significantly less time than a typical BitTorrent download.

EE-BitTorrent is clearly suitable to save energy, and also keeps the underlying P2P principles of the legacy BitTorrent architecture. The overall BitTorrent network is not modified, as the proxy acts exactly as a standard BitTorrent peer. Modifications are just required at the proxy and at the user PCs behind the proxy, and are thus confined within a single LAN. Note that different proxies "masking" peers on different LANs are completely independent of each other. Therefore, this architecture is also scalable, as it does not require modifications of the BitTorrent global architecture, nor global coordination between (sets of) BitTorrent peers. Finally, EE-BT is also suitable to support mobile clients accessing the Internet, e.g., through WiFi Access Points connected to the LAN where the proxy is running, and, more in general, is a solution to enable *asynchronous BitTorrent download*, which is something not supported by the conventional BitTorrent architecture.

V. Performance Evaluation

To evaluate the effectiveness of the proposed solution, we implemented EE-BitTorrent in a real testbed and derived, through a set of measurements, the energy savings provided by the proxy-based protocol with respect to the legacy protocol.

The experimental environment was based on a set of PCs, which were interconnected by a Gigabit Ethernet LAN connected to the Internet via a high-speed 100 Mbps link. All PCs used Linux Ubuntu 8.04 (Hardy Heron). The BitTorrent client was a simple command line client provided with Rasterbar libtorrent.

Our experimental testbed consisted of two systems running in parallel: the legacy BitTorrent system, where each peer downloads files in standalone mode, and the EE-BitTorrent system, where peers delegate the file download to the proxy and, then, switch off. We performed a large set of experiments, measuring the overall time taken for downloading a set of a given number of files. To limit the variability in Internet conditions and in the swarm size, we interleaved the experiments with and without the proxy. Finally, we repeated the same experiment many times. For each experiment we used the same number of files, but we changed the set of files in different experiments. We selected files with similar popularity and size (about 4 GB), so as to achieve comparable results. The results presented below are averaged over the total number of replications.

To quantify the effectiveness of our proposal we measured the time required for downloading each file, and derived the total energy E_L and E_P consumed for downloading the same set of files with the legacy and proxy-based protocol, respectively (see [27] for details). Hence, the energy savings provided by EE-BitTorrent is given by

$$S = 1 - \frac{E_P}{E_l}. \quad (1)$$

Obviously, when using EE-BitTorrent, E_P includes not only the energy consumed by PCs but also the energy consumed by the proxy (unless this is a multi-server machine).

TABLE I shows the energy savings provided by our EE-BitTorrent, with respect to the legacy approach, for different number of PCs that are simultaneously downloading a file. If the BitTorrent proxy runs on a multi-server machine that must be active also for other purposes (and, hence, its energy consumption can be neglected), the energy savings is not affected by the number of simultaneous downloads, and it is approximately equal to 95% for each PC. When the proxy is running on a dedicated machine, its energy consumption must be taken into account as well. Therefore, the total energy consumed with EE-BitTorrent may be even larger than that consumed with the traditional protocol. Actually, this happens when there is a single active download. This is because the energy savings with a single download is negative in TABLE II.

TABLE I
Energy Savings for Different Number of PCs.

Number of PCs	Energy savings (%)	
	Dedicated Machine	Multi – Server Machine
1	-21 %	95 %
2	52 %	96 %
3	50 %	94 %
4	59 %	95 %
5	48 %	96 %
6	57 %	95 %

TABLE II
Average Download Time Experienced by Each Single File in the Legacy and Proxy – based Architecture.

Number of PCs	Average Download Times (s)	
	Legacy	Proxy – based
1	8844.89	8844.89
2	8927.82	5618.60
3	6168.82	4509.35
4	7719.75	7171.00
5	9261.91	6797.69
6	10967.55	9103.90
7	7188.39	5669.38

Obviously, as the number of simultaneous downloads increases, the additional energy consumed by the proxy is compensated by the reduction in the energy consumed by PCs. It can be shown that, for a very large number of simultaneous downloads, the energy savings asymptotically converges to the same value achieved with a multi-purpose machine (i.e., 95% in our testbed) [27].

Using our proxy-based BitTorrent architecture is not only beneficial in terms of reduced energy consumptions, but it also improves the performance of the download process. TABLE II shows the average download time of a file, for an increasing number of simultaneous downloads, with the legacy and our proxy-based architecture, respectively. The average download time varies significantly in the different experiments because the set of used files is different in the various experiments and, also the network conditions vary over time. However, in all the experiments the average file download time reduces significantly when using EE-BitTorrent. This is because the BitTorrent proxy is able to achieves more bandwidth than a single normal peer, as it shares more file than the single peer.

VI. Conclusions

In this chapter we have proposed EE-BitTorrent, a proxy-based version of BitTorrent optimized for energy efficiency. The goal of EE-BitTorrent is to reduce the energy consumption of user PCs during the file download, without introducing any degradation in the Quality of Service (QoS) perceived by the user, i.e., without increasing the file download time. We have implemented and evaluated EE-BitTorrent in a real testbed. The experimental results show that using our proxy-based approach can save up to 95% of the

energy consumed by each PC when using the legacy solution. We also observed that EE-BitTorrent does not introduce any QoS degradation. Rather, the average time to download a file *reduces* by approximately 22% when using the proxy-based architecture since the number of files shared with the overlay network by the proxy is greater then the number of files shared by any single peer.

REFERENCES

[1] K. Christensen and A. D. George, "Increasing the Energy Efficiency of the Internet with a Focus on Edge Devices", The Energy Efficient Internet Project, University of South Florida and University of Florida, Florida, 2005 – 2008.

[2] S. Ruth, "Green IT - More Than a Three Percent Solution", IEEE Internet Computing Magazine, Vol. 13, N. 4, July-August 2009.

[3] K. Christensen, B. Nordman, and R. Brown, "Power Management in Networked Devices", Communications column *IEEE Computer*, Vol. 37, No. 8, pp. 91-93, August 2004.

[4] C. Gunaratne, K. Christensen, S. Suen, and B. Nordman, "Reducing the Energy Consumption of Ethernet with an Adaptive Link Rate", IEEE Transactions on Computers, V.57, N.4, April 2008.

[5] National Energy Foundation (NEF), S. Karayi, "The PC energy report", 1E, London, 2007. http://www.1e.com/energycampaign/downloads/1E_reportFINAL.pdf

[6] H. Schulze and K. Mochalski, "The Impact of Peer-To-Peer file sharing, voice over IP, Skype, Joost, Instant Messaging, One-Click Hosting and Media Streaming such as YouTube on the Internet", IPOQUE – Internet Study 2007, Leipzig, Germany, September 2007.

[7] G. Anastasi, M. Conti, and A. Passarella, "Power Management in Mobile and Pervasive Computing Systems", Chapter 24 in Algorithms and Protocols for Wireless and Mobile 12 Networks, Azzedine Boukerche (Editor), CRC Computer and Information Science Publisher, October 2005.

[8] K. Christensen, "An Energy Efficient Internet: Ongoing Work" presentation, Center for Communications and Signal Processing, University of South Florida, Tampa, Florida, April 18, 2008. San Jose, California, March 5, 2008.

[9] C. Gunaratne and K. Christensen, "Ethernet Adaptive Link Rate: System Design and Performance Evaluation", Proceedings IEEE Conference on Local Computer Networks, pp. 28 – 35, November 2006.

[10] K. Christensen and F. Blanquicet, "An initial performance evaluation of Rapid PHY Selection (RPS) for Energy Efficient Ethernet", Proceedings IEEE Conference on Local Computer Networks, pp. 223 – 225, October 2007.

[11] Y. Fukuda, T. Ikenaga, H. Tamura, M. Uchida, K. Kawahara, and Y. Oie, "Dynamic Link Control with Changing Traffic for Power Saving", Proceedings of the IEEE Local Computer Networks Conference (LCN), October 2009.

[12] P. Reviriego, J. Hernandez, D. Larrabeiti, and J. Maestro, "Performance Evaluation of Energy Efficient Ethernet", IEEE Communications Letters, Vol. 13, No. 9, pp. 1-3, September 2009.

[13] M. Gupta, S. Grover, and S. Singh, "A feasibility study for power management in LAN switches", IEEE ICNP 2004, Berlin, Germany, October 2004.

[14] S. Singh and M. Gupta, "Using low-power modes for energy conservation in Ethernet LANs", INFOCOM 2007 26th IEEE International Conference on Computer Communications IEEE, Portland State University, Portland, May 2007.

[15] M. Gupta and S. Singh, "Dynamic Ethernet Link Shutdown for Power Conservation on Ethernet Links", Proceedings of IEEE ICC, June 2007.

[16] K. Christensen and F. Gulledge, "Enabling power management for Network-attached computers", International Journal of Network Management, vol. 8, N. 2, pp. 120 – 130, March – April 1998, pp. 1099 – 1190.

[17] C. Gunaratne, K. Christensen, and B. Nordman, "Managing Energy Consumption Costs in Desktop PCs and LAN Switches with Proxying, Split TCP Connections, and Scaling of Link Speed", International Journal of Network Management, Vol. 15, No. 5, pp. 297-310, September/October 2005.

[18] K. Christensen and B. Nordman, "Improving the Energy Efficiency of Ethernet - Connected Devices: A Proposal for Proxying", Ethernet Alliance White Paper, September 2007. http://efficientnetworks.lbl.gov/enet-pubs.html

[19] M. Jimeno, K. Christensen, and B. Nordman, "A network Connection Proxy to Enable Hosts to Sleep and Save energy", IEEE International Performance Computing and Communications Conference, pp. 101 – 110, December 2008.

[20] Y. Agarwal, S. Hodges, J. Scott, R. Chandra, P. Bahl, and R. Gupta, "Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage", Proceedings USENIX Symposium on Networked System Design and Implementation (NSDI, 2009), Boston, MA, USA, April 2009.

[21] S. Nedevschi, J. Chandrashekar, B. Nordman, S. Ratnasamy, and N. Taft, "Skilled in the Art of Being Idle: Reducing Energy Waste in Networked Systems", Proceedings USENIX Symposium on Networked System Design and Implementation (NSDI, 2009), Boston, MA, USA, April 22-24, 2009.

[22] Yuvraj Agarwal, Stefan Savage, and Rajesh Gupta, "SleepServer: A Software-Only Approach for Reducing the Energy Consumption of PCs within Enterprise Environments", Proceedings of the USENIX Annual Technical Conference (USENIX '10), Boston, MA, June 2010.

[23] J. Blackburn and K. Christensen, "A Simulation Study of a New Green BitTorrent", Proceedings First International Workshop on Green Communications (GreenComm 2009), Dresden, Germany, June 2009.

[24] J. Kurose and K. Ross, "Peer-to-Peer Applications", Computer Networking. A Top-Down Approach, IV Edition, Addison Wesley, 2007.

[25] D. Towsley, "The Internet is Flat: A brief history of networking over the next ten years", ACM PODC 2008, 2008.

[26] A. R. Bharambe, C. Herley, and V. N. Padmanabhan, "Analyzing and Improving BitTorrent Performance", Technical Report MSR-TR-2005-03, February 2005.

[27] G. Anastasi, I. Giannettia, and A. Passarella, "A BitTorrent Proxy for Green Internet File Sharing: Design and Experimental Evaluation", Computer Communications, Vol. 33, No. 7, pp 794-802, May 2010.

A Distributed Architecture for Energy-Efficient Data Mining over Mobile Devices

Carmela Comito, Domenico Talia, and Paolo Trunfio

***Abstract*—The dissemination and increasing power of wireless devices is opening the way to support analysis and mining of data in a mobile context. Enabling mobile data mining is a significant added value for nomadic users and organizations that need to perform analysis of data generated either from a mobile device (e.g., sensor readings) or from remote sources. A key aspect to be addressed to enable effective and reliable data mining over mobile devices is ensuring energy efficiency, as most mobile devices are battery-power operated. This paper proposes a general architecture for pervasive data mining over mobile devices focusing on energy efficiency. In such an architecture, a mobile device can play the role of data producer, data analyzer, client of remote data miners, or a combination of them. Stationary nodes provide the necessary support to enable mobile nodes to organize themselves into local groups to perform mobile-to-mobile data mining (M2M-DM) computations.**

***Index Terms*—Energy efficiency, distributed computing, mobile-to-mobile data mining.**

I. Introduction

An increasing number of cell-phone and PDA-based data intensive applications are starting to appear. Examples include cell-phone-based systems for body-health monitoring, vehicle monitoring, and wireless security systems. Monitoring data in small embedded devices for smart appliances, on-board monitoring using nano-scale devices are examples of such applications that we may see in the near future. Support for advanced data analysis and mining is necessary for such applications.

Data mining from such mobile/embedded devices faces various challenges because of several reasons such as (1) low-bandwidth networks, (2) relatively small storage space, (3) limited availability of battery power, (4) slower processors, and (5) small displays to visualize the results. We need to design algorithms and systems that can perform data analysis by optimally utilizing the limited resources.

A key aspect to be addressed to enable effective and reliable data mining over mobile devices is ensuring energy efficiency, as most mobile devices are battery-power operated and lack a constant source of power. Most commercially available mobile computing devices like PDAs and mobile phones have battery power which would last for only a few hours. Therefore, the next generation of data mining applications for such embedded and mobile devices must be designed to minimize the energy consumption. Software power utilization and minimization have been studied in various contexts [1], [2], [3], [4], [5] but, to the best of our knowledge, only very few studies have been devoted on energy requirements for data mining algorithms [6].

This paper proposes a general architecture for pervasive data mining over mobile devices focusing on energy efficiency. In such an architecture, a mobile device can play the role of data producer, data analyzer, client of remote data miners, or a combination of them. As such, we envision an architecture in which there are several distributed mobile devices and stationary servers where the mobile devices can run some steps of the data mining task, or some lightweight data mining algorithms.

The mobile devices cooperate in a peer-to-peer style to perform a data mining process tackling the problem of energy capacity and processing power limitations. Whenever a resource limited computing device (client) in such a cooperative environment has a set of tasks (or subtasks) to be executed (which may have dependencies and communication requirements among themselves), it uses all available resources in nearby computing devices (servers).

The remainder of this paper is organized as follows. Section II presents the overall architecture of the proposed framework. Section III describes the software components inside each mobile device. Finally, Section IV concludes the paper.

II. System Architecture

The system is designed to enable *Mobile-to Mobile Data Mining* (M2M Data Mining) applications having energy efficiency as the primary goal. In the following we present the overall architecture of the system.

A typical M2M data mining scenario includes *stationary nodes* (e.g., computer servers) and *mobile devices* (e.g., mobile phones, PDAs). Stationary nodes can act as server nodes for executing the data mining tasks submitted by mobile clients. On the other hand, the possibility of performing data mining over a mobile device may include several application scenarios in which a mobile device can play the role of data producer, data analyzer, client of remote data miners, sever or a combination of them. More specifically, we can envision five basic scenarios for mobile data mining.

1) The mobile device is used as a terminal for ubiquitous access to a remote server that provides some data mining services. In this scenario, the server analyzes data stored in a local or a distributed database, and delivers the results of the data mining task to the mobile device for its visualization.
2) Data generated in a mobile context are gathered through a mobile device and sent in a stream to a remote server to

C. Comito, D. Talia and P. Trunfio are with the Department of Electronics, Computer Science and Systems (DEIS), University of Calabria, Rende, Italy e-mail: {ccomito,talia,trunfio}@deis.unical.it.

D. Talia is also with the Institute of High Performance Computing and Networking of the Italian National Research Council (ICAR-CNR), Rende, Italy.

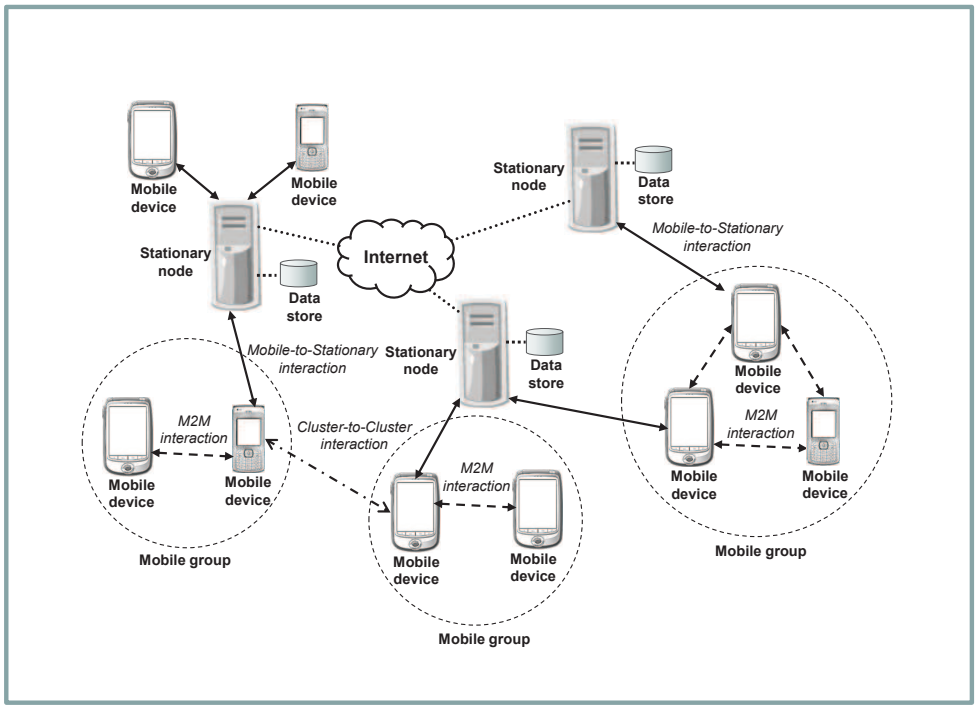


Fig. 1. The M2M-DM architecture. The arrows denote remote service calls.

be stored into a local database. Data can be periodically analyzed by using specific data mining algorithms and the results used for making decisions about a given purpose.

3) Mobile devices are used to perform data mining analysis. Due to the limited computing power and memory/storage space of todays mobile devices, it is not possible to perform heavyweight data mining tasks on such devices. However, some steps of a data mining task (e.g., data selection and preprocessing) or very simple mining tasks on small data sets can be executed on mobile devices.
4) The mobile device acts as a data mining server for other mobile clients. As stated earlier, data analysis provided by a mobile device may include either lightweight data mining algorithms or some steps of the whole process.
5) A mobile device acts as as a gateway for other mobile devices. In this case, even if the mobile gateway itself does not provide processing, it plays the fundamental role of linking poorly connected devices to a remote processing node.

The system architecture, depicted in Figure 1, has been designed to allow on-demand collaborations among mobile nodes. Examples of mobile-to-mobile collaborations regard several areas such as disaster relief, construction management and healthcare. In order to promote and easy collaborations when two or more mobile users meet each other, we let them grouping into *clusters* referred to as *mobile groups*. Consequently, the M2M-DM architecture includes a number of stationary nodes and a number of mobile groups.

Cluster formation is an important issue to be addressed. Clusters may be formed based on many criteria such as communication range, number and type of mobile devices, and their geographical location. In particular, we group the mobile devices on the basis of their transmission range. More precisely, when two or more co-workers standing within a given area meet, their mobile devices will discover each other and create an ad-hoc network in order to form a cluster. Each cluster has a node referred to as the *cluster head*, which acts as the coordinator for the cluster, manages the other nodes within the cluster, and interacts with the other local groups in the network.

Figure 1 shows the interactions among the different components of the architecture. Stationary nodes are connected through the Internet and can interact with the other nodes (including the mobile ones) in order to execute a data mining task. Mobile nodes within a group interact trough ad-hoc connections (e.g., wi-fi, bluetooth) that we refer to as *M2M connections*, represented as dotted arrows in Figure 1. Interactions among mobile groups (*cluster-to-cluster connections*) take place through ad-hoc connections among the cluster-heads of the groups and are represented as dot-dash arrows. Mobile groups are connected to stationary nodes through their cluster head (*mobile-to-stationary connections*) by exploiting an Internet connections (e.g, wi-fi, wi-max). All types of interactions take place whether to ask for a data mining request or to cooperate in order to collaboratively execute a data mining task.

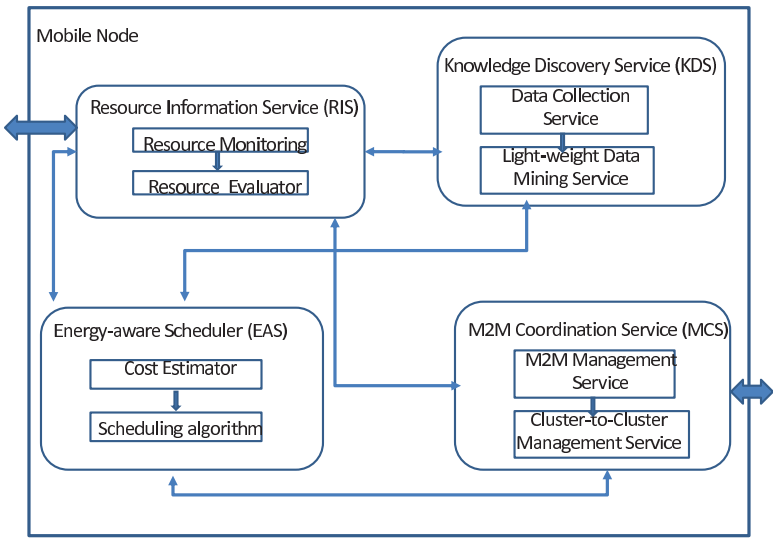


Fig. 2. Software components inside each mobile device.

In the M2M-DM architecture, both stationary and mobile nodes provide a specialized set of services, as detailed below.

Stationary nodes provide three groups of functionalities:

- *Knowledge discovery*, to execute or support the different steps of the knowledge discovery process (preprocessing, data mining, visualization, etc.).
- *Data management*, allowing to store and retrieve data (e.g., data generated either by mobile devices or by third-party data providers).
- *Coordination*, allowing mobile devices to organize themselves into groups and manage computations in a cooperative way (e.g., registration services, discovery services, etc.).

Mobile devices provide the following group of functionalities:

- *M2M knowledge discovery*, to execute knowledge discovery tasks that can be executed on limited resources, such as preprocessing, visualization, or lightweight data mining processes.
- *M2M resource management*, allowing to monitor the local resources (e.g., memory, CPU load, battery status) to establish whether the device is able to execute a data mining task.
- *M2M coordination*, enabling mobile devices to organize themselves into local groups on a temporary basis for on-purpose knowledge discovery applications.
- *M2M interaction*, allowing interactions with nodes inside or outside the group. The interactions with nodes external to the group are realized through the cluster head that acts as a gateway towards the outside of the group.

III. Mobile Device Components

Mobile nodes include a set of software components that cooperatively perform the functionalities introduced in the previous section. As shown in Figure 2, each node includes four software components: *Resource Information Service* (*RIS*), *M2M Coordination Service* (*MCS*), *Energy-aware Scheduler* (*EAS*), *Knowledge Discovery Service* (*KDS*).

The RIS is responsible to collect information about all the resources inside a mobile node and the context in which an application is running in order to adapt its execution. To this aim, the RIS is composed of two modules implementing the above cited features:

- *Resource Monitoring* module. It informs the system about the mobile device resources measurement such as the available memory, CPU utilization, battery consumption, battery level, remaining time to fill memory, network connectivity performance.
- *Resource Evaluator* module. This module acts as a resource measurement receiver from both local and environmental resources. It then takes some actions on the basis of the received measures, i.e. choosing the most suitable configuration for the data mining task. Moreover, the module is responsible for starting the data mining task with the appropriate parameters.

The MCS is responsible for the coordination among mobile devices and includes two modules:

- *Mobile-to-Mobile Management* module. It includes mechanisms aiming at the coordination of the nodes within a group such as cluster formation and maintenance, joining of a new node to a group, cluster-head election, cooperative data mining task execution.
- *Cluster-to-Cluster Management* module. This module provides mechanisms allowing mobile devices to organize themselves into clusters, cluster-to-cluster interactions to the end of a data mining task allocation, coordination to collaboratively execute a data mining task.

The EAS is the component responsible for task assignment among local groups. It implements a scheduling strategy aimed at prolonging network life-time by distributing energy consumption among local groups. In such an approach, whenever a resource-limited computing device (client) has a set of tasks

to be executed, it uses the energy resources in nearby computing devices (servers) and an efficient task assignment is found in such a way that the total consumed energy is minimized. The scheduler interacts with the RIS through its resource-monitoring and resource-evaluator modules. Moreover, the scheduler is also tightly related to the KDS component, as it is actually the scheduler that activates a data mining process. The EAS includes three modules:

- *Cost Estimator* module. This module exploits information about availability, performance and cost of resources collected by the RIS component. It deals with the actual calculation of the estimation functions on the basis of the perceived status of resources w.r.t. time, energy and load constraints.
- *Mapper* module. This module schedules the tasks. It embeds a scheduling algorithm, and a matchmaker that takes into account resource characteristics, incorporates interdependencies among resource groups or types, and computational and I/O cost evaluations to map the available resource units to newly scheduled tasks according to a pre-specified mapping objective function.
- *Scheduling Process* module. This module guides the scheduling activity. It receives jobs, requests the corresponding schedules to the mapper, and orders the execution of scheduled tasks.

The KDS is responsible for the execution of a knowledge discovery task over a mobile device. It includes two modules:

- *Data Collection* module. This module provides access/store mechanisms for data to be processed or generated as a result of a data mining process. Typically, only a limited amount of data can be stored on a mobile device. Therefore, this module will manage the interaction with a stationary node that will act as a storage node or as a source for data.
- *Lightweight Data Mining* module. It is responsible for managing the execution of a data mining task on the mobile device, if possible. If the mobile resources are not (or no more) sufficient to carry out the whole computation, this module can delegate the process to another node(s). As an example, this may happen when the resource measures indicate that the device can not achieve the required accuracy according to the incoming data rate. In such a case, the node sends a request to a data mining server (either stationary or mobile) to continue the current process with the specified accuracy;

IV. Conclusion

The development of software framework for running data mining tasks on cooperating mobile devices will allow to exploit such devices for novel data analysis applications. Handling the energy efficiency issue is a significant contribution for making mobile devices effective platforms for supporting complex applications in nomadic scenarios.

The architecture presented in this paper is a first step toward the implementation of a framework for energy-efficient mobile-to-mobile data mining. We are currently working in three directions:

1) defining a formal energy model for a mobile data mining scenario;
2) defining a scheduling strategy that takes into account the energy requirements of algorithms and the energy capability of the devices;
3) implementing a prototype of the system, starting from the implementation of the software components devoted to cluster formation, energy measurements, and scheduling.

References

[1] K. Barr and K. Asanovic. Energy aware lossless data compression. In Proceedings of USENIX/ACM International Confenrence Mobile Systems, Applications, and Services, pp. 231244, May 2003.

[2] Z. Li, C. Wang, and R. Xu. Computation offloading to save energy on handheld devices: a partition scheme. In Proceedings of ACM International Confenrence Compilers, Architecture, and Synthesis for Embedded Systems, pp. 238246, Nov. 2001.

[3] A. Rudenko, P. Reiher, G. J. Popek, and G. H. Kuenning. Saving portable computer battery power through remote process execution. SIGMOBILE Mobile Computing and Communication Review, 2(1):1926, Jan. 1998.

[4] J. Flinn, M. Satyanarayanan. Energy-aware adaptation for mobile applications. Proceedings of the Symposium on Operating Systems Principles, pp. 48 63, 1999.

[5] S. Gurun and C. Krintz. Addressing the energy crisis in mobile computing with developing power aware software. UCSB, Computer Science Department, Technical Report, 2003.

[6] R. Bhargava, H. Kargupta, and M. Powers. Energy Consumption in Data Analysis for on-board and Distributed Applications. Proceedings of the ICML'03 workshop on Machine Learning Technologies for Autonomous Space Applications, 2003.

Research line on power-aware computing by the High Performance Computing and Architectures Group

Manuel F. Dolz
Juan C. Fernández,
Enrique S. Quintana-Ortí,
Rafael Mayo, High Performance Computing and Architecures Research Group
Universitat Jaume I
Castelló de la Plana, Spain
Email: {dolzm,jfernand,quintana,mayo}@uji.es
Antonio Peña
Departamento de Informática de Sistemas y Computadores
Universidad Politécnica de Valencia
Valencia, Spain
Email: apenya@gap.upv.es

***Abstract*—High Performance Computing and Architectures (HPCA) Group is now opening a new research line on energy saving on high performance computing platforms. Our first work in this area has been the development of an energy saving module for the Sun Grid Engine queue system and Rocks Clusters operating system. By turning on only those nodes that are actually needed at a given time during the execution of a batch of jobs, the module yields substantial energy savings. On energy saving research area, HPCA is interested in the virtualization of specific resources (such GPU for GPGPU) to minimize the number of such resources in a specific installation, in the development of a model of the power requirement of an application running on a specific platform (assuming a heterogeneous computer platform), and the impact of selecting an algorithm (from a set of algorithms that can be used for the same problem) in the energy consumption.**

***Index Terms*—High performance computing, data centers, green computing, power consumption, virtualization.**

I. Introduction

A. About HPCA

The High Performance Computing & Architectures (HPCA) group was created in 2006 at the University Jaume I (Spain) from the fusion of the Parallel Scientific Computing group and the Advanced Computer Architecture and Reconfigurable Computing group of this university.

The HPCA group pursues the optimization of numerical algorithms for general purpose processors (superscalar and VLIW) as well as specific hardware (GPUs and FPGAs), and their parallelization on both message-passing parallel systems (mainly clusters) and shared-memory multiprocessors (SMPs, CC-NUMA multiprocessors, and multicore processors). The group is involved in the application of high-performance parallel computing techniques to the solution of problems arising in control theory, computational chemistry, electromagnetics, aeronautic engineering, and scientific and engineering applications in general. Current interests of the group also include power-aware computing, hardware-software codesign, reconfigurable architectures, and high-speed networks and QoS.

This research was supported by projects CICYT TIN2008-06570-C04 and FEDER, and P1B-2009-35 of the *Fundación Caixa-Castellón/Bancaixa* and UJI. Part of this work is done in collaboration with the Parallel Architecture Group of the Universitat Politècnica de València

B. People

The HPCA group is composed of 12 researchers, all of them faculty members of the ”Depto. de Ingeniería y Ciencia de los Computadores” of the University Jaume I (Spain). There are also four assistant researchers and one Ph.D. student that currently pursue their Ph.D. in the research lines of the group.

C. Collaborations

The members of the HPCA group have conducted research on the application of high-performance parallel computing techniques to scientific and engineering applications. Below is a list of a few of the challenges that have been confronted:

- Real-time Magnetic Resonance Imaging (Chicago hospital, 1995).
- Analysis of tensions of airplane components (Boeing Ltd.,1996).
- Fault-tolerant computing for aerospatial vehicles (Jet Propulsion Lab & NASA, 2001).
- Evaluation of the gravitational field of Earth (Dept. of Aerospace Engineering & Engineering Mechanics - The University of Texas at Austin, 2003).
- Model reduction for VLSI design and simulation (Philips Research Labs., 2004-).

- Analysis of tensions of ceramic materials (Instituto de Tecnología Cerámica - Universidad Jaume I de Castellón, 2006-).

Some of these activities have been carried out in collaboration with several national (Spanish) and foreign research groups. Currently, we cooperate with researchers from the following centers:

- Barcelona Supercomputing Center (Spain).
- Chemnitz University of Technology (Germany).
- ETH Zurich (Switzerland).
- IBM T. J. Watson Research Center (USA).
- Intel GmbH (Germany).
- Technical University Carolo-Wilhelmina at Brunswick (Germany).
- Technical University of Denmark (Denmark).
- Texas Advanced Computing Center (USA).
- The University of Texas at Austin (USA).
- Ume University (Sweden).
- University of California at Berkeley (USA).

II. Research Line on Power-aware Computing

A. Introduction

The High Performance Computing and Architecture Research Group of the "Jaume I" University of Castelln has been working on the development of high performance algorithms for engineering problems since its creation on 1991. We have acquired an in deep knowledge in this area. In the last two years we have open a new research line on energy saving for high performance HPC clusters. Our goal is to apply our experience in the development, maintenance and use of high performance clusters. We are working in the development of tools and new algorithms taking into account not only the performance from the view point of instructions per second but also from the view point of energy efficiency. We have two complementary lines of work:

- The development of algorithms and libraries that can be run with the less possible use of energy with the less possible impact on the performance.
- The study and development of strategies for managing datacenters, balancing the requirements of performance/throughput and energy efficiency.

B. Energy efficency in datacenters

There exists a lot of references in the last years about the energy consumption and the need of putting the energy efficiency as one of the requirements in the design, implementation and managing of datacenters. The term Green-Computing is becoming familiar for most of the computer scientists and there are several initiatives in USA and EU in this field. All studies show that energy consumption in datacenters is mainly due to the non-IT part of it. In fact the cooling subsystem is the responsible of more than the 50% of the energy consumption, but it is known too that the cooling requirements are due to:

- The building design and construction.
- The IT infrastructure.

We focus our attention in the IT infrastructure: design, deployment and managing. If the energy consumption is a key goal in the design, deployment and managing of the datacenter IT infrastructure it is possible to reduce the cooling requisites and then having a better relation between the energy used on the IT infrastructure and the overall energy used in the datacenter. First we have to adopt a measure of the efficiency of the energy used, PUE and PUEi have been defined and are a good start point. There are two complementary paths to minimize the energy usage. One is at code level, by the design of codes that use in an energy efficient form all the hardware of the system from the execution units to the memory hierarchy. The second one is to have middleware tools that yield to an energy efficient use of the overall system. We focus on two types of middleware.

One of them to schedule and balance work on the nodes to get the best use of the cooling system. High Performance Computing (HPC) clusters have been widely adopted by companies and research institutions for their data processing centers because of their parallel performance, high scalability, and low acquisition cost. On the other hand, further deployment of HPC clusters is limited due to their high maintenance costs in terms of energy consumption, required both by the system hardware and the air cooling equipment. In particular, some large-scale data processing centers consume the same energy power as 40,000 homes. Studies by the U.S. Environmental Protection Agency show that, in 2007, the power consumption of data centers in the United States was around 70 billion KWatt-hour, representing 5,000 million euros and the emission of 50 million tones of CO2. Since the benefits of HPC clusters are clear, scientists and technicians are currently showing special interest in all types of solutions and ideas to minimize energy costs in data processing centers.

Energy-awareness has spread among researchers from organizations like IEEE, which have analyzed HPC clusters, concluding that a significant part of the energy consumed by these systems is due to the interconnection of its components (switches, network cards, links, etc.); following this result, energy-aware algorithms have been developed which can disable idle interconnections in the cluster. Microsoft inspects the problem from a different viewpoint and one solution proposed is to share highly efficient power supplies among several nodes of the system, achieving significant energy savings.

In this context one of the most well-known energy management techniques is DVFS (Dynamic Voltage and Frequency Scaling). DVFS entails reducing the system energy consumption by reducing the CPU supply voltage and the clock frequency (CPU speed) simultaneously. This technique has a great impact on the development of work aimed at reducing consumption in this research context.

Alternative strategies to limit power consumption and required cooling of HPC clusters are based on switching on and shutting down the nodes, according to the needs of the users' applications. We have developed a tool, named EnergySaving [6], [7], that allows the definition of different conditions to activate and deactivate nodes for a full adaption to the requirements of the system administrator and/or the end user. A simulation of the module under real conditions will

show that its use combined with a reasonable policy deliver considerable energy savings compared with a conventional cluster in which all nodes are permanently active. This tool will be described in Section 3.

The other one is based on remote access and virtualization techniques to provide services of high energy demand hardware with the best relation between energy consumption and performance. One of this high energy demand hardware with an increasing use in today's HPC datacenters are the GPU's. The increasing computing requirements for GPUs (Graphics Processing Units) have favored the design and marketing of commodity devices that can nowadays also be used to accelerate general purpose computing. Therefore, high performance clusters intended for HPC (High Performance Computing) may include such devices. However, high-end GPU-based accelerators used in HPC feature a considerable energy consumption, so that attaching a GPU to every node of a cluster will have a strong impact on its overall power consumption.

Virtualization techniques may provide significant energy savings, as they enable a larger resource usage by sharing a given hardware among several users, thus reducing the required amount of instances of that particular device. As a result, virtualization is being increasingly adopted in data centers. In this way, virtualizing GPUs may report power and cost benefits. We have developed an initial rCUDA framework. This framework enables the concurrent usage of CUDA-compatible GPUs remotely. To enable a remote GPU-based acceleration, our framework offers remote access to virtualized CUDA-compatible devices from any node in the cluster. Thus, all of the nodes are able to concurrently access the whole set of CUDA accelerators installed in the cluster, independently of which nodes the GPUs are physically attached to. In another words, our solution aims at offering a noticeable reduction in execution time to computationally-intensive applications running in an HPC cluster equipped with only a few CUDA-compatible accelerators, enabling remote hardware acceleration. As far as we know, this is the first solution to enable CUDA remote acceleration in HPC clusters. Our experiments demonstrate that rCUDA leads to a reduction of the number of GPUs required in the system, thus attaining considerable energy savings.

III. rGPU. Virtualizing the access to GPU accelerators

In the last years hardware virtualization has become a way to reduce acquisition, administration, maintenance, space, and energy costs of high performance computing (HPC) clusters and data processing centers The idea behind this technique is to avoid attaching a given device to each of the nodes of a cluster by virtualizing those installed in a smaller number of nodes, that become servers that provide the virtualized device services to the rest of the cluster.

GPU virtualization is of particular relevance for HPC applications, which frequently make use of the GPU as a code accelerator, so that computationally intensive tasks are offloaded there. Remarkable speed-ups have been attained using

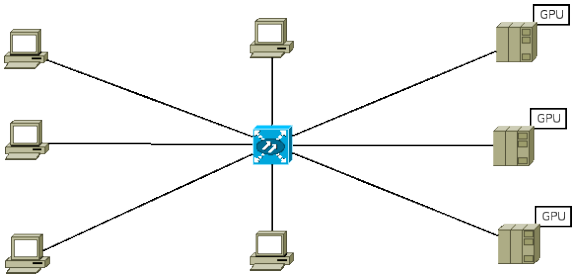


Fig. 1: Cluster configuration with only a few nodes equipped with a GPU-based code accelerator.

the NVIDIA CUDA framework in many research fields, such as computational fluid dynamics, image analysis or computational algebra, to name only a few.

Virtualizing GPUs provides several benefits. First, adding an accelerator to every node in an HPC cluster is not efficient neither from the performance point of view nor from the power consumption perspective.Second, in a configuration where all the nodes of a cluster are equipped with a GPU, it is unlikely that all GPU accelerators will be fully loaded all the time, and therefore it is reasonable to set up a system with a number of nodes larger than the amount of accelerators. Hence, by hopefully reducing only slightly the performance of GPU-accelerated applications, it is possible to reduce acquisition costs.

A number of upcoming commercial solutions allow that multiple servers in a rack share a few GPUs —such as NextIO's N2800-ICAby PCI-Express (PCIe) virtualization— ,enabling a cluster configuration where only a few of its nodes have an attached GPU, like in the scheme depicted in Figure 1. However, they do not allow multiple nodes to concurrently access the same GPU, as GPUs need to be assigned to a specific node at each time. Moreover, these proprietary hardware-supported solutions are non-standard and often substantially expensive.

Many efforts to virtualize GPUs have been conducted in the field of virtual machines and graphics, or CUDA. However, when GPUs are used as accelerators, and therefore there is no need to take care of visual output or virtual machine-related issues, the specifics of virtualizing GPUs change drastically. Therefore, this new target environment led us to adopt a different approach. To overcome the constraints of the previous work we adopted a software front-end GPU virtualization by API (Application Programming Interface) remoting, which avoids the need of hardware support.

Our proposal, is based on the use of a client at the local machine requesting GPU services that forwards the requests to the server owning the GPU and latter receives the results and delivers them to the original application. In our solution, the bandwidth between the main memory of the computer requesting acceleration services and the memory of the remote GPU is limited by the network interconnect, since the bandwidth of PCIe is in general higher than that of the network. Fortunately, GPU codes are frequently compute-intensive and I/O communication is usually not the bottleneck.

The most important publications of the HPCA group related

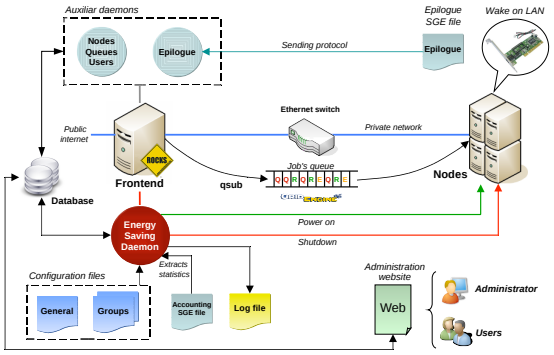


Fig. 2: Diagram of the energy saving module.

to virtualization techniques are [1], [2], [3], [4] and [5].

IV. EnergySaving Tool

A. Introduction

In this context a well-known energy management technique is DVFS (Dynamic Voltage and Frequency Scaling). DVFS entails reducing the system energy consumption by simultaneously decreasing the CPU supply voltage and the clock frequency (CPU speed). Complementary strategies to limit power consumption and required cooling of HPC clusters are based on switching on and shutting down the nodes, according to the needs of the users' applications. PowerSaving is a prototype example of this strategy which provide little functionality or are still under development. This tool allows the definition of different conditions to activate and deactivate the nodes for a full adaption to the requirements of the system administrator and/or the end user. The tool has been designed and implemented as a module (Roll) for Rocks®and employs the Sun® Grid Engine (SGE). The target hardware platform used in this work is an HPC cluster equipped with a front-end node that is responsible of the queue system and the new energy saving module (Roll).

B. Implementation of the Energy Saving Roll

In this section, we describe the energy saving module in detail; see Figure 2. The module includes the following major components:

- Three daemons in charge of managing the database, collecting statistics, and executing the commands that power on and shutdown the nodes. These daemons are implemented in Phyton.
- The website interface to configure and administer users' groups as well as set the threshold triggers that define the power saving policy.

1) Daemons:

a) Daemon for epilogue requests: The epilogue daemon employs the information provided by the *epilogue script* of SGE queuing system to perform a series of updates in the energy saving module database.

b) Daemon for the queues, users and nodes: This daemon is responsible for ensuring that all information on users, nodes and queues that are actually in operation in the SGE queue system is correctly reflected by the database.

c) Daemon for the activation/deactivation actions and statistics: This daemon, the most important of the module, activates and deactivates the nodes according to the users' requests. Specifically, a node can be turned on if a lack of resources for a particular job is detected, the average waiting time of the jobs is greater than a threshold, or the number of enqueued jobs exceeds a threshold. On the other hand, a node can be turned off if the idle time exceeds a threshold, the waiting time of enqueued jobs is lower than a threshold, or the current jobs can be served using a smaller number of nodes. In addition, there are also options to select candidate nodes to be powered on and strict levels which can produce different behaviors and energy savings. Node activation is done via WakeOnLAN while deactivation is performed with the `shutdown` command through `ssh` session on target nodes.

2) Website interface: The module has an interface that eases the administration of the energy saving module. Some possible operations using this website include:

- Check and modify configuration parameters of the energy saving system.
- Monitor the operation of the cluster through a series of diagrams which illustrate the active/inactive node times, the average waiting time and execution time of jobs, etc.
- Monitor the energy savings in terms of power consumption and economic cost.

Moreover, this interface seamlessly integrates with Rocks®, and facilitates remote access and administration.

3) Experimental Results: To evaluate the benefits of the system we have developed a flexible simulator that provides information on the system behavior for various platform configurations and under realistic workloads. Among many other statistics, the simulator reports the percentage of the time that each node in the cluster will be turned on/off and, therefore, offers an estimation of the energy consumption. We have configured this simulator to emulate the system of queues of the HPC computing facility service at the Universitat Jaume I (UJI). This facility is composed by 65 nodes with different architectures. On the other hand, the job benchmark, obtained from the real queue system logs of the same computing facility, is composed by 10,415 jobs corresponding to the load submitted during three full months. After evaluating the system with different policies, we have achieved significant energy savings.

The policy without the energy saving module, where all the nodes are powered on all the time, yields an average response time (latency) per job over 339 h, and consumes 65.37 MWh to execute all the jobs. With the application of our system, the job latency roughly increases to 461 h, but now the percentage of time that nodes are powered on is only 42.9%, and the power consumption is reduced to 29.51 MWh. The nodes were down a considerable period of time (roughly 1,856 h) for this particular workload. This high value indicates that nodes have been deactivated for long periods of time and, therefore, the decision of keeping them down is feasible. The

result also demonstrates that, for this particular workload, it is more convenient to turn nodes off than to keep them active using, e.g., DVFS, as the time needed to reactivate a node is negligible compared with the period of time it remains inactive.

REFERENCES

[1] José Duato, Antonio J. Pea, Federico Silla, Rafael Mayo, Enrique S. Quintana-Ortí. Modeling the CUDA Remoting Virtualization Behaviour in High Performance Networks. First Workshop on Language, Compiler, and Architecture Support for GPGPU. Bangalore (India) January 2010.

[2] José Duato, Antonio J. Pea, Federico Silla, Francisco D. Igual, Rafael Mayo, Enrique S. Quintana-Ortí. Accelerating Computing through Virtualized Remote GPUs. Actas de las XX Jornadas de Paralelismo 2009. pp 635-639. ISBN: 84-9749-346-8

[3] José Duato, Francisco D. Igual, Rafael Mayo, Antonio J. Pea, Enrique S. Quintana-Ortí, Federico Silla. CUDA Remoto para Clusters de Altas Prestaciones. II Workshop en Aplicaciones de Nuevas Arquitecturas de Consumo y Altas Prestaciones (ANACAP 2009). ISBN: 978-84-692-7320-3

[4] José Duato, Francisco D. Igual, Rafael Mayo, Antonio J. Pea, Enrique S. Quintana-Ortí, Federico Silla. Virtualized remote GPUs. Advanced Computer Proceedings of the Architectures and Compilation for Embedded Systems (ACACES 2009). pp 221-224 ISBN: 978-90-382-1467-2

[5] José Duato, Francisco D. Igual, Rafael Mayo, Antonio J. Pea, Enrique S. Quintana-Ortí, Federico Silla. An Efficient Implementation of GPU Virtualization in High Performance Clusters. Proceedings of the 4th Workshop on Virtualization in High-Performance Cloud Computing (VHPC 2009). Delft (Netherlands).

[6] Manel F. Dolz, Juan C. Fernández, Rafael Mayo, Enrique S. Quintana-Ortí. EnergySaving Cluster Roll: Power Saving System for Clusters. Proc. Architectures of Computing Systems (ARCS 2010). Lecture Notes in Computer Science 5974. pp 162-173. ISSN: 0302-9743.

[7] Manel F. Dolz, Shehed Kudama, Juan C. Fernández, Rafael Mayo, Enrique S. Quintana-Ortí, A. Ruíz, J.I. Sánchez. Monitorización y ahorro de energía en clusters de computadores. Actas de las XIX Jornadas de Paralelismo 2008. pp 591-597. ISBN: 978-84-8021-676-0.

Broadcasting Protocols in MANETs

K. Dar*, M. Bakhouya*, J. Gaber*, M. Wack*, P. Lorenz+
* Universite de Technologie de Belfort-Montbéliard
Rue Thierry Mieg, 90010 Belfort Cedex, France
{kashif.dar, maxime.wack, gaber}@utbm.fr, bakhouya@gmail.com
+ University of Haute-Alsace, France
lorenz@ieee.org

Abstract - Broadcast is an essential building block of any MANET, however, it requires substantial routing and re-transmissions that results in extra energy consumption. In this paper, we have studied several MANETs broadcast protocols and compared them with respect to the number of redundant transmissions. Simulations have been conducted, and based upon the analysis; we have prioritized the protocols with respect to energy effectiveness.

Key words - MANETS, broadcast protocols, energy efficient.

I. INTRODUCTION

The Mobile Ad Hock Networks (MANETs) are communication networks formed on the fly by radio-equipped mobile nodes without a fixed infrastructure. The broadcasting in MANETs is an important function e.g. for cooperative operations, group discussions, and common announcements. The core problem in multi-hop broadcasting is how to minimize the number of redundantly received messages in order to save transmission energy while, at the same time, maintaining good latency and reachability since rebroadcasting causes tradeoff between reachability and efficiency under different host densities. Therefore, the selection of relay nodes and their transmission power is a major design consideration in routing and broadcasting algorithms. Several broadcasting protocols (simply written as protocols, hereafter) for information dissemination have been proposed for MANETs [1, 3, 4, 5].

The energy efficiency problem in broadcast wireless network design has received significant attention in the past few years. Unlike the wired networks, in which the energy consumption is not a major concern, energy efficiency in wireless networks is very important e.g. in military networks or wide area voice and data networks. In such networks, the mobile hosts are powered by batteries; therefore the limited battery life time imposes constraints on the network performance [10]. Hence, in order to maximize the network lifetime, the traffic should be routed in such a way that the energy consumption is minimized [11].

In this paper, we will first classify the protocols used in MANETs for broadcasting purpose, a short description of each protocol will be provided in next section followed by their comparative discussion in section 3. Our main point of interest will be on energy efficiency in term of reduced number of re-transmissions.

II. PROTOCOLS CLASSIFICATION

We have classified each protocol into two main categories: *Statistical or geometric based* and *Network topology based* protocols. The former protocols category depends upon certain threshold (e.g. distance, redundant message counts, or broadcast probability) values to estimate the network density while protocols in later category use sophisticated structures or neighborhood information to construct the broadcast schedule. Figure 1 depicts the whole classification.

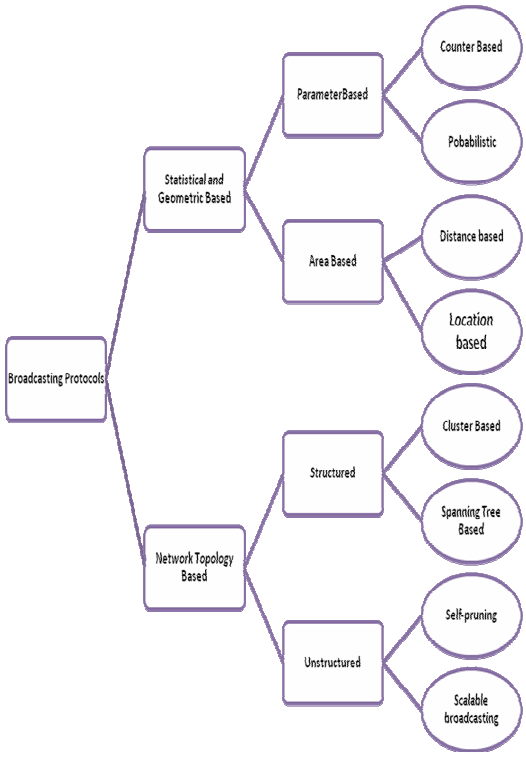


Figure 1. The classification of broadcasting protocols used in MANETs.

A. Statistical or Geometric Based Protocols

The statistical or geometric based protocols are also subdivided into: *parameter based and area based*. Parameter based protocols use certain parameters like broadcast probability, hope counters etc to reduce the number of redundantly received packets. The area based broadcasting techniques exploits the geographical location of the node to calculate the additional coverage area of the sender.

A.1. Parameter Based Protocols

The parameter based protocols basically extend the *flooding* technique, in which the source node disseminates a message to all its neighbors only if this message is seen first time. The classical flooding algorithm has several drawbacks; first it is rather costly in terms of air interface usage, secondly, it is not reliable since most of the nodes are expected to broadcast the message at the same time, thus collusions are likely to occur, thirdly, it causes broadcast storm problem [3] that severely affect the energy consumption due to redundant message re-broadcast. Therefore, flooding has been modified into following techniques.

A.1.1. Counter Based

In counter based broadcasting, a message will be rebroadcasted only if the number of received copies at a host is less than a threshold after RDT (Random Delay Time, which is randomly chosen between 0 and Tmax seconds) [7]. In [8], authors have modified the counter based protocol and named the new protocol as Hop Count Ad hoc Broadcasting (HCAB) protocol. In HCAB, upon receiving a broadcast message for the first time, the node initiates a flag R = true and records initial hop count value HC0 of this message. Meanwhile, this node sets a RDT value between 0 and Tmax. During the RDT, the node compares the hop count of redundantly received message HCx with HC0 and flag R is set to false if HCx > HC0. When the random delay expires, the node will relay this message if R is true. Otherwise, it just drops this message. The counter based techniques rely only on fixed counter and also the message delay at each hop is increased due to RDT and counter comparison.

A.1.2. Probabilistic Based

In probabilistic scheme, mobile hosts rebroadcast messages according to certain probability that is defined at the initial stage. The major drawback of this technique is that setting the probability dynamically in different traffic situations is not an easy task. In [2], authors have introduced a scheme for dynamic probabilistic broadcasting in MANETs. The fixed probabilistic approach reasonably reduces the number of redundant re-broadcasts; however, its performance suffers in less dense networks and need high probability to achieve good reachability.

A.2. Area Based

Area based information broadcasting schemes take advantage of the geo-graphical location of the nodes [6]. Two main approaches used in this category are discussed as follows.

A.2.1. Distance based

The distance based approach only the neighbor far away from the current node rebroadcasts the message i.e. a distance threshold value is defined. Upon reception of a previously unseen message, a RDT is initiated and redundant messages are cached. When the RDT is expired, all source node locations are examined to see if the node is closer than a threshold distance value. If true, the node doesn't rebroadcast.

A.2.2. Location based

The location based scheme uses a more precise estimation of expected additional coverage area in the decision to rebroadcast. In this method, the source node also appends its geographical position information with the message. The receiving node then calculates the additional broadcast coverage area with the help of positioning data sent by the source node. If the additional area is less than a threshold value, the node will not rebroadcast, and all future receptions of the same message will be ignored. Otherwise, a node assigns a RDT before delivery. If the node receives a redundant message during a RDT, it recalculates the additional coverage area and compares it with the threshold. This process is continued until the message is rebroadcasted or finally dropped. The major drawback of this scheme is that it assumes the node in a network should be equipped with a GPS device.

Moreover, area based methods also presume that visibility can be estimated merely from the position of the nodes i.e. it mainly depends upon the distance of the nodes. This is realistic only if there are no shading objects, e.g., users are in a plain field. Finally, the distance calculation also increases broadcast latency and computational overhead that ultimately cause energy in-efficiency.

B. Network Topology Based Protocols

The network topology based protocols are further categorized into structured and unstructured protocols. Structured protocols use geometrical shapes or data structure to make an information dissemination plan whereas unstructured protocols use neighborhood information to calculate the additional number of recipient nodes.

B.1. Structured Protocols

B.1.1. Cluster based

Despite its many applications, the cluster based approach is also used for broadcasting in which mobile hosts form clusters.

Within one cluster, each host is treated as a member, and there is one cluster head and one gateway node responsible to relay messages. However, maintaining such structure is too costly or even impossible especially when the nodes mobility is very high. Furthermore, in a clustered MANET, each node periodically sends 'Hello' message to advertise its presence which consume extra transmission energy.

B.1.2. Spanning tree based

In [9], authors have described a spanning tree based algorithm for broadcasting in ad hoc networks. The whole broadcasting mechanism is divided into two parts: (i) the maintenance of the broadcast tree, and (ii) the broadcast process itself using the tree. The spanning tree based broadcast scheme is considered to be inappropriate for ad hoc networks, being too difficult and resource consuming and being too sensitive to the link failures.

B.2. Unstructured Protocols

Unstructured protocols use neighbor nodes knowledge to make broadcast decision. Following two schemes are used for this purpose.

B.2.1. Self pruning

In self-pruning, each node maintains the knowledge of its neighbors by periodically exchanging the "Hello" messages. The receiving node first compares its neighbors list to that of sender's list, and rebroadcast the message only if the receiving node can cover additional nodes. The neighbor knowledge attached with the identity of the node from which the packet is received allows a receiving node to decide if it would reach additional nodes by rebroadcasting.

B.2.2. Scalable broadcasting

The scalable broadcasting further enhances the self-pruning scheme by gathering neighbors' information up to two hop distance. Thus, each node has a two hop topology information.

It is worth noting that neighbor knowledge based algorithms also consume extra transmission energy since they require substantial communication between the nodes to exchange the topological and geographical structure of the network.

III. PERFORMANCE ANALYSIS

In this study we have used a realistic mobility scenario using MOVE (MObility model generator for Vehicular networks) [12] and TRaNS (Traffic and Network Simulation Environment) [13]. A node mobility pattern defines its motions within the network area during a simulation time. The scenario generated by using these tools is a grid topology of 800x800 square meters with a block size of 200mx200m as depicted in Figure 2.

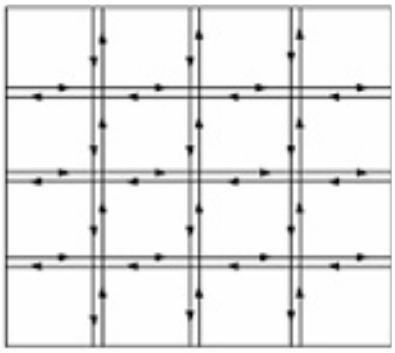


Figure 2. Map of the simulated scenario.

A simulation time of 100s is used, which is long enough to evaluate the broadcasting protocols by varying nodes speed and densities. Each node use IEEE 802.11 MAC protocol to send and receive messages. We used two-ray ground model for radio propagation [14]. Other simulation parameters are described in Table 1.

Table 1. Simulation parameters.

Simulation Parameter	Value
Network range	800 square meters
Transmission range	200 meters
Number of nodes	25-100
Nodes speed	1-25 meters/second
Bandwidth	2Mbps
Message size	1000 bytes
Simulation time	100 seconds
Number of trials	10

A. Simulation results

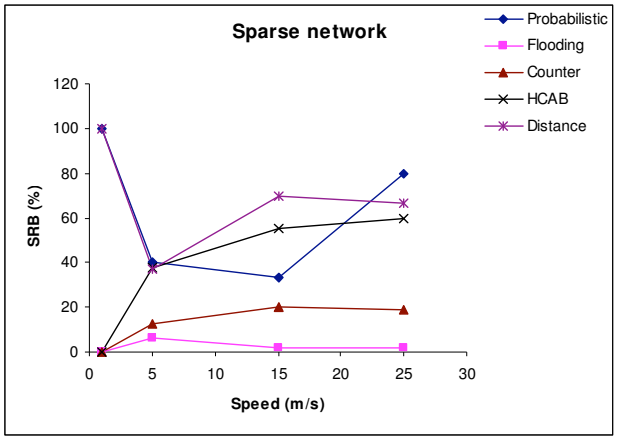
We measured the performance of some distinguished broadcasting protocols (Probabilistic, Flooding, Counter Based, HCAB, and Distance Based) under different nodes' speed, network topologies, and host densities. Different threshold values for the protocols under investigation are defined as follows: for Probabilistic, the relay probability is set to 0.5, for Counter Based, the counter threshold is set to 3, and for Distance Based, the distance threshold is set to 150 m (0.75× transmission range).

The key performance metric that we are interested to measure is the Saved ReBroadcast (SRB) which is the ratio between the number of host receiving the message and the number of hosts actually rebroadcasting the message. Since our goal is to measure the transmission energy consumption against each protocol, the number of SRB is inversely proportional to the transmission energy. So the greater number of SRB will represent the lesser transmission energy consumed during the broadcasting. We performed ten simulation trials for each scenario and calculated the average number of SRB against

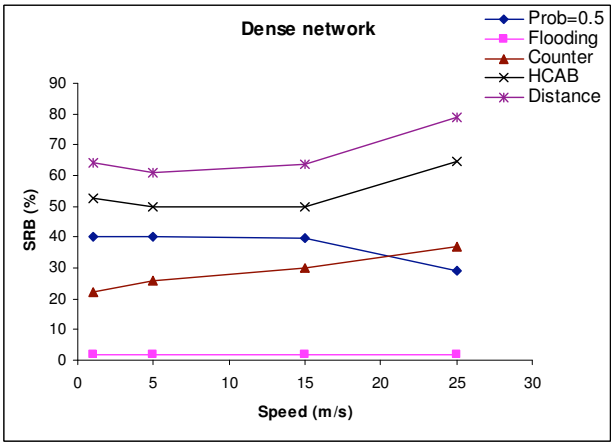
each protocol. In the following subsections we will analyze the obtained results in terms of both nodes speed and density within the network.

A.1. Speed effect

Figures 3 plot SRB of five schemes as function of nodes speed. We analyzed the effect of nodes speed under both sparse and dense networks. In Figure 3 (a), when nodes speed are very low (< 5 m/s), the Probabilistic and Distance Based schemes reveal gradual down in SRB value (although still high from other protocols), however, SRB value of Distance Based improves when nodes speed is above 5m/s but the SRB for Probabilistic still goes downwards and starts improving when nodes speed exceed the threshold of 15 m/s. HCAB gradually improves SRB as nodes speed is increased.



(a)



(b)

Figure 3. SRB vs. nodes speed: (a) sparse network, (b) dense network.

In Figure 3(b), due to greater nodes density, Distance Based outperforms the other protocols. HCAB (although not efficient as Probabilistic and Distance Based) also shows consistent increase in SRB. Unlike sparse network, the amount of saving SRB in Probabilistic decreases when nodes speed is increased while the performance of Counter Based protocol gradually improves. However, Flooding shows the worst behavior in both sparse and dense network. In conclusion, under different nodes speed, the Distance Based scheme works more efficiently, both in sparse and dense networks, in term of SRB as compared to other broadcasting schemes.

A.2. Density effect

In this sub-section, we will investigate the effect of nodes density within the range from 25 to 100 nodes where nodes speed is fixed to 15 m/s. Figure 4 depicts the number of SRB associated with number of nodes in the network. We can see that, Distance Based exhibited better performance when compared with other schemes and number of SRB increases proportionally with the increase in number of nodes proving once again that Distance Based scheme is the best candidate in the dense network situation.

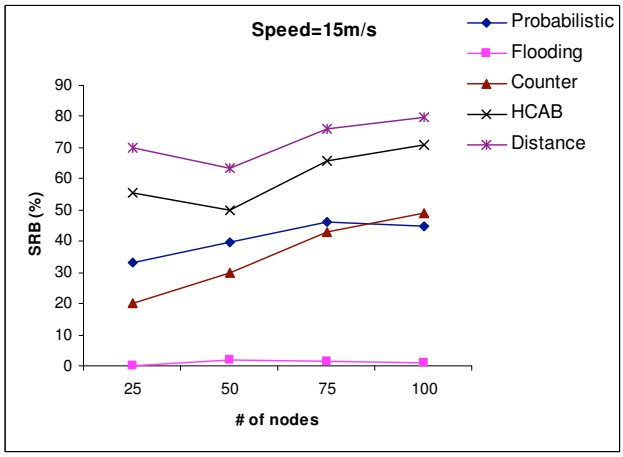


Figure 4. Number of SRB v/s number of nodes.

IV. SUMMARY AND CONCLUSIONS

MANETs differ from traditional wired and infrastructure-based wireless networks, due to their envisioned applicability and increased dynamics due to nodes motion. In addition, most of the times; MANETs nodes have limited battery charging capacity that requires energy efficient protocols especially for broadcast purpose since broadcast operation require subsequent routing and re-transmissions in order to disseminate the information among all network nodes.

In this paper, we first surveyed the most common broadcast protocols that are used in MANETs. Since our main objective was to find the most efficient protocol with respect to energy consumption. We selected 5 potential protocols to measure their performance in term of communication over-head i.e. the number of re-transmissions during the broadcast operation. We observed that, Distance Based scheme is the most efficient with respect to SRB both in term of network density and nodes

mobility except that its performance slightly degrades in sparse network. HCAB is the second best option, while Probabilistic protocol remains at third. In sparse network, when nodes speed are greater than 20 m/s, Probabilistic protocol shows significant improvement. The performance of Counter Based protocol is slightly lower than Probabilistic, however, this outperforms the Probabilistic when network become more dense (i.e. number of nodes > 80) or when nodes move at higher speed (i.e. speed > 20 m/s). Flooding remains the worst in all situations due to its static behavior. From these results, we conclude that no single protocol is able to give optimal broadcast solution with respect to energy efficiency.

Future activities involve the design of an efficient and adaptive broadcast protocol for MANETs that will provide the near-optimal solution in term of minimum energy consumption while maintaining good latency and reachability.

V. REFERENCES

[1] Hugo Miranda, PhD Thesis, "Gossip-Based Data Distribution in Mobile Ad Hoc Networks", DI-FCUL/TR-07-33.

[2] Q. Zhang, D. P. Agrawal, "Dynamic Probabilistic Broadcasting in MANETs", Journal of parallel and Distributed Computing, Volume 65, pp 220-233, 2005.

[3] S.Y. Ni, Y.Tseng, Y. Chen, and J. P. Sheu, "The broadcast storm problem in a mobile ad hoc network", International Conference on Mobile Computing and Networking, pp 151 – 162, ISBN:1-58113-142-9, 1999.

[4] B. Williams and T. Camp, "Comparison of Broadcasting Techniques for Mobile Ad hoc Networks", In Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), pp 194–205, 2002.

[5] K. Dar, M. Bakhouya, J. Gaber, M. Wack, "Information dessemination approaches in VANETs", ICPS 2010: The 7th ACM International Conference on Pervasive Services, Berlin, Germany, [To appear in July 2010]

[6] D. Kouvatsos, I. Mkwawa, "Broadcasting Methods in Mobile Ad Hoc Networks: An Overview", Proceeding of the HetNet, UK, 2005.

[7] K. W. Kim, K. K. Kim, C. Han, M. M. Lee, and Y. Kim, "An Enhanced Broadcasting Algorithm in Wireless Ad-hoc Networks", International Conference on Information Science and Security, 2008.

[8] Q. Huang, Y. Bai, L. Chen, "Efficient ligt weight broadcasting protocols for multihop ad hoc networks", The 17th Annual IEEE International Symposium on PIMRC'06.

[9] A. Juttner and A. Magi, "Tree based broadcast in ad hoc networks", Mobile Networks and Applications, vol. 10, no. 5, pp. 753–762, 2005

[10] M.X. Cheng, J. Sun, M. Min, and D.-Z. Du, "Energy Efficient Broadcast and Multicast Routing in Ad Hoc Wireless Networks," Proc. 22nd IEEE Int'l Performance, Computing, and Comm. Conf., 2003.

[11] D. Li, X. Jia, and H. Liu, "Energy efficient broadcast routing in static ad hoc wireless networks", IEEE transactions on Mobile Computing, Vol. 3, No. 2, June 2004.

[12] F. K. Karnadi, Z. H. Mo, and K. C. Lan, "Rapid Generation of Realistic Mobility Models for VANET", In ACM MOBICOMM (2005).

[13] M. Piorkowski, M. Raya., A. L. Lugo, P. Papadimitratos, M. Grossglauser, and J.P. Hubaux, "TraNS: Realistic Joint Traffic and Network Simulator for VANETs", ACM SIGMOBILE Mobile Computing and Communications Review (Vol. 12 , Issue 1, pp. 31-33, 2008).

[14] Network Simulator NS 2.34, avilable via website http://www.isi.edu/nsnam

Methodology of Measurement for Energy Consumption of Applications

Georges Da Costa
IRIT
Université Paul Sabatier
Toulouse, France
Email: Georges.Da-Costa@irit.fr

Helmut Hlavacs
University of Vienna
Department of Distributed and Multimedia Systems
Lenaug. 2/8, Vienna, Austria
Email: helmut.hlavacs@univie.ac.at

***Abstract*—Energy awareness can be improved in two ways for IT systems, in a static or in a dynamic way. First by building energy efficient systems, that will run fast and consume only a few watts. The second consist to react to instantaneous energy consumption, and then by taking decision that will reduce this consumption.**

In order to take decision, it is necessary to have a precise view of the energy consumption of each element of an IT system. Nowadays, it is only possible to measure energy at the outlet level. Thus, when several applications are run it is difficult to evaluate each application consumption.

This work aims at evaluating energy consumption of each application using indirect measurements. In this paper, we describe the methodology of data acquisition and firsts models based on those measurements. We evaluate that using a naive model we obtain already a precise model of energy consumption in function of indirect measurements[1].

I. INTRODUCTION

Energy awareness can be improved in two ways for IT systems, in a static or in a dynamic way. First by building energy efficient systems, that will run fast and consume only a few watts. The second consist to react to instantaneous energy consumption, and then by taking decision that will reduce this consumption.

Those two approaches are complementary, and we mainly address the second one, using autonomous systems. Currently our system reacts to several events, one of them being energy consumption of hosts. Current hardware technology allow only to measure energy consumption of a whole node. As we manipulate Virtual Machines, we are interested in the energy consumption of each application or VM inside the node.

To address this problem it is necessary to create a software captor able to extract the energy consumption of each application.

Several techniques are currently used, but they either focus on a particular subsystem (such as processor[1], memory[2], GPU[3]) or are not precise enough (in [4] a comparison of several model leads to an average error of 10%).

Generally some process related values are monitored, and based on those values we can then create a mathematical model linking those data to the energy consumption of an application.

In the following we will explain the current methodology used to acquire data, then we will present a naive model and conclude.

[1]This work was done during a short term Scientific Mission (Cost action 0804) of Georges Da Costa (IRIT, Toulouse, France) to the University of Vienna (Austria) from 8, January to 12, February 2010

II. METHODOLOGY

The global methodology is the following :

- We run several applications and synthetic benchmarks
- We log several measurements during the run (including energy consumption)
- We use the logged values to derive a mathematical model linking measurements and energy consumption

To achieve application energy measurement, we use indirect measurements such as performance counters, process related information (using pidstat) and host related information using (collectd). The two first information type are related to one process, and are to be the base of the produced model. Information based on collectd are machine-wide and thus not related to a particular process. But some measure (such as network) are currently difficult to obtain in a process-related way.

In order to reduce the bias, we did not choose the value to be monitored. Thus a part of the mathematical modeling will be to evaluate the impact of each value on the energy consumption. Values range from *number of instruction per second* to *Core number on which the process is running*.

The current implemented framework use two elements, a monitoring part and a synthetic workload one.

A. Monitoring

The monitoring part is split on two computers in order to reduce the impact of measures on the experiment. A first computer runs the application to be measured and make the process and machine related measurements. A second computer is linked to a wattmeter and logs the energy consumption. At start the two computers clocks are synchronized.

The current limit of the framework is that it cannot follow at the same time a process and its children. So it can currently only be used for applications that do not fork. As most benchmarks actually fork, we had to develop new ones to create the large datasets on which to deduce the models.

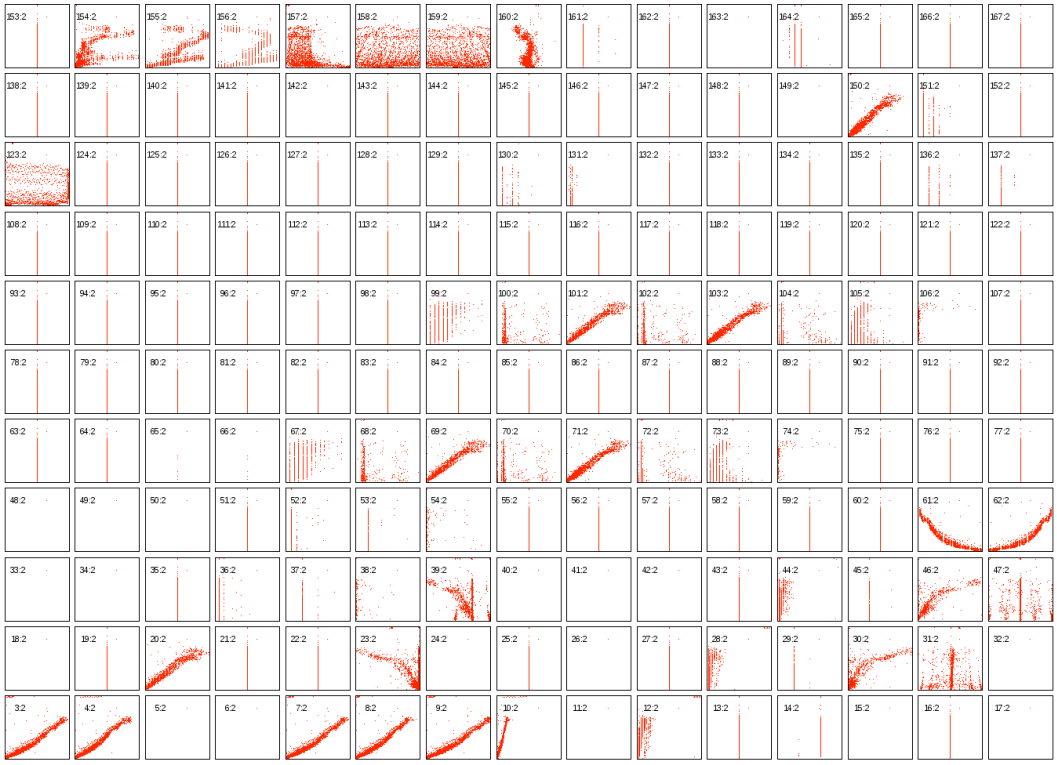


Fig. 1. Graph of the 165 measured values in function of energy consumption for an synthetic disk benchmark.

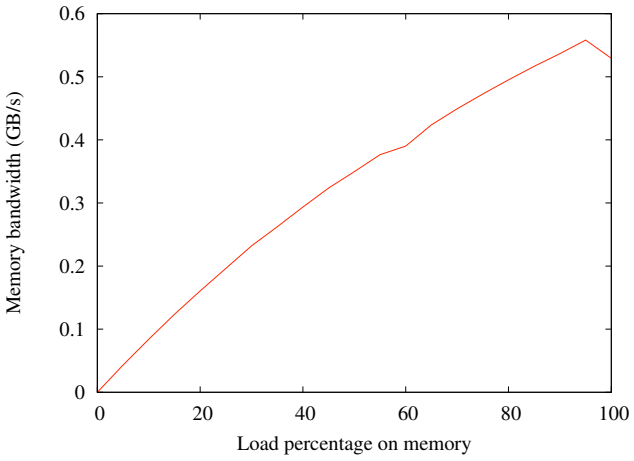


Fig. 2. Example of synthetic workload. Shows the memory bandwidth really used in function of the percentage requested.

B. Synthetic workload

We created several benchmark linked to synthetic workloads: Memory, CPU, network, Disk, Idle General behavior of each of them is always the same: They start at 100% of a particular resource and go down to 0% by step of 1%. Each step is 20s. Each second we measure around 200 captors (using pidstat, perfcounters, collectd and a wattmeter) so one experiment produce after data treatment around 2MB of data.

Starting at 100% and going down to 0% allows to reduce the impact of allocating resources as advised by SpecPower.

Figure 1 shows an synthetic workload example. It shows the memory bandwidth used by the memory benchmark as a function of the requested percentage of memory bus. This benchmark stresses memory bus and memory banks. It is linear, and the other produced benchmarks follow the same principle. It is important to be remarked that the framework does only require that the benchmarks span the whole space, not necessarily in a linear way.

III. Experiments

We finally created some tools to treat the data and to have a fast overview of the possible relationships between energy consumption and measured values.

As values come from different tools, we first developed tools to aggregate those data in a large csv file. Then we created some tools to have a fast overview of the possible correlation between energy and each measured value. For this, we plotted a graph of each of those values in function of energy. Figure 2 shows these graphs for the Read synthetic benchmark. In this Figure it is clear that some measurements are correlated with energy and that some are not. We use this tool only to have some insight on the possible correlation, but the final choice will be done using only statistical evaluation.

By instance, using a simple model using only the number of instructions per second (using performance counters) and read and write bandwidth on the disk we obtain (using the formula 70.74 + cp*2.252e-02 + rd*8.602e-05 + wr*8.140e-05) a standard error of less than 3 (ie around 5%) on our experiments. From an other point of view, it means that the total energy consumed is precise at around a few joules per second on a computer that consumes between 70 to 130W depending on the load). More precise models are currently worked on, taking into account more values, such as network or memory usage.

IV. Conclusion

We put in place a measurement platform at the University of Vienna, we created the software for it, developed several synthetic benchmarks and provided first example of simple models. The first models are already as precise as state of the art models. Future works include choosing the best values to measure in order to reduce the impact of measurement on the system. A second future work will take care of spawn processes in order to be able to work on a larger number of applications.

References

[1] R. Joseph and M. Martonosi, "Run-time power estimation in high performance microprocessors," in *ISLPED '01: Proceedings of the 2001 international symposium on Low power electronics and design*. ACM, 2001.

[2] I. Kadayif, T. Chinoda, M. Kandemir, N. Vijaykirsnan, M. J. Irwin, and A. Sivasubramaniam, "vec: virtual energy counters," in *PASTE '01: Proceedings of the 2001 ACM SIGPLAN-SIGSOFT workshop on Program analysis for software tools and engineering*. ACM, 2001.

[3] X. Ma, M. Dong, L. Zhong, and Z. Deng, "Statistical power consumption analysis and modeling for gpu-based computing," in *SOSP Workshop on Power Aware Computing and Systems (HotPower '09)*, 2009.

[4] S. Rivoire, P. Ranganathan, and C. Kozyrakis, "A comparison of high-level full-system power models." in *HotPower*, F. Zhao, Ed. USENIX Association, 2008.

Improving energy efficiency with adaptative load controllers on web servers

J. Vicens and C. Juiz
University of the Balearic Islands
Department of Mathematics and Computer Science
Palma, Spain
Email: {jaume.vicens, cjuiz}@uib.es

Abstract—In recent years, the use of the Internet has experienced tremendous growth, both on behalf of users, such as data size and complexity of calculations. Thus, one consequence is the excessive energy consumption caused by data centers, which contain thousands of servers. We propose a comprehensive study of energy consumption in relation to web server performance. We also propose the creation of new methods of assessment and monitoring of performance and consumption of Web servers, because existing tools are incomplete or propietary. With them, we would create smart algorithms improving the adaptability to Web systems for any types of loading conditions to self adjust its energy consumption to achieve, under the loading pattern detected, the maximum performance per watt.

***Index Terms*—energy efficiency, web performance engineering, green IT, web benchmarks**

I. INTRODUCTION

In last 15 years the number of Internet users has grown very rapidly. Evidence of this is that in 1995 the IDC had 16 million Internet users, however, last year there were about 1.596 million users. Looking at only the last 10 years this growth was also significant because since 2000 the number of users has grown exponentially. However, the number of users with Internet is not the only growth factor. More and more Web applications are used, in turn, they are increasingly complex. Also is increasing the bandwidth available to users and service providers, have markedly improved wireless connections, both at home and in cellular phones, making it possible to offer Web services that require more intensive use or move a greater volume of data over the network. One consequence of this is that users increasingly opt to replace most desktop applications for fully integrated applications on the Web, requiring services of cloud computing. It is this term (cloud computing) that has gained strength during the two or three years, proposing a return to a more centralized organization web servers. This aspect requires the creation of an infrastructure and several services online where size is determined by the needs of end users. With this schema, user data reside ”in the cloud”, and therefore increase the number of visits to these services every time they need to be consulted. However, not only cloud computing applications are growing. The traditional Web applications that use client-server paradigm have also experienced a large increase in recent years. Lately, corporate application servers were located very close to the end users, quite decentralized. But the need for quality control of these applications, some synergy costs, growth of the economies of scale and accelerated centralization of Internet infrastructure cause centralizing these services go back into what is known as large server farms, which have increased in size through use in the cloud of the applications that the user had downloaded before.

Far from reducing the energy consumption of these particular data centers and across the Internet in general, these have been growing at a fast pace as energy consumption [1]. This consumption has grown so much that can be likened to that of the Czech Republic to complete [2]. Data requirements will not stop growing and to build data centers are increasing all the ecological impact that this entails.

To reduce this consumption is necessary to study the energy efficiency of data centers, however, this is very difficult given the diversity and complexity of their infrastructures. There are five industry segments that affect the energy consumption of a data center: (1) servers and storage systems, (2) power conditioning equipment, (3) cooling and humidification systems, (4) network equipment, and (5) physical security lighting. Therefore, in the creation of a sustainable data center can act from mechanical engineers to software engineers, as well as through electrical engineers, which divide the challenge into multiple layers and disciplines [3].

This position paper describes how to improve the energy efficiency of Web servers and their study and procedure for such improvements. The work aims to improve overall energy efficiency coating application servers.

A. State of Art

Energy use has implications for density, reliability and reliability of server data center. As data centers house more servers and consume more energy, more heat is generated, which consumes a lot of energy in refrigeration systems and, due to temperature increase, the reliability of all components decreases.

These problems represent large problems that cause data centers are designed near hydroelectric plants to waste as little energy during transportation and take advantage of the cooling water that a river can provide, in addition to be located in places with a temperature half low enough throughout the year [4]. Of course, all these concerns are reflected in the upgrading of the cooling infrastructure and efficiency of servers [5].

To help improving these servers, so that energy efficiency is concerned, some institutions have created benchmarks that focus on energy efficiency of a server [6], to establish common metrics to assess progress reducing the energy consumption.

While there have been great advances in energy efficiency of systems in general, it is notable that the I/O subsystems has been very little emphasis [7]. As demonstrated by a benchmark of great recognition as the SPECWEB'09 only incorporate a I/O simulator backend of a web server that provides simulated delays, taking into account only the performance of these subsystems, rather than incorporating the actual measurement of their energy consumption, since in reality these subsystems represent a large potential demand for high workloads.

At server level, is necessary to understand the relationship between resource use and consumption by power systems, i.e., to maximize the energy efficiency of a set of web servers, it is imperative to keep off as many servers as possible while ensuring a minimum quality of service [8][9][10], because the current hardware is very inefficient at low utilization [11].

One of the improvements that are being extended at the server level is to add a layer of virtualization. In this way a set of physical machines can host a much larger set of virtual machines that can be added or removed very easily, keeping the same physical infrastructure. This allows greater utilization of physical servers, because the insertion of multiple virtual servers with a lower average utilization in a single physical server causes the hardware is used in a greater degree of utilization, which promotes energy efficiency.

At the level of implementation, however, must be made aware of the impact energy of the applications with the following objective: to build applications that consider real-time energy measurements and energy policies on the server set to dynamically auto-adjust their productivity and power consumption [12].

Use of data mining techniques can be very useful energy, for example, make classifications energy states or to identify usage patterns so as to establish a policy or other energy consumption according to the pattern detected. At this level has more on the power consumed in relation to productivity granted, allowing more aggressive adjustments in certain states of the system load. For example, an application can "decide" their energy consumption dynamically [13], temporarily disabling low-level device if it determines that at certain moment are unnecessary. Of course, it is essential that all such decisions always take consider quality of service as this will be very fragile these adjustments energy.

II. METODOLOGY

To improve the efficiency of a Web server we propose the use of adaptive algorithms sensitive to the energy state of a system. These algorithms are designed after a previous study of energy consumption of a Web server, detailed in the following subsections.

A. Determine patterns of energy consumption and performance of Web applications by modeling.

The different types of Web systems, their architecture and implementation can be very heterogeneous. It is therefore necessary to conduct a study where they get models that show the energy efficiency of Web servers with respect to its performance. For these preliminary studies are normally used benchmarks, monitors and other tools available today. From these studies one can infer the model performance and energy consumption together. For example, regression models can infer how energy consumption varies as the performance offered by the system. It is also possible to build simulation models, which although they are not as accurate as to emulate more complex situations typical of web servers through benchmarking and monitoring techniques.

B. Define and build the tools necessary for better assessment of energy efficiency.

Following the pattern of the preceding paragraph, it should be possible to design and to establish the requirements for evaluation and monitoring of energy efficiency that are not covered in the benchmarks and current monitors. Thus, it is essential to build or adapt the tools for a more advanced level of customization and cover all the existing needs that, for example, SPECWEB'09 not covers in the case of the benchmarks.

This last benchmark cannot be customized since it has only three methods of evaluation for specific load scenarios. Therefore, it could be difficult to adapt it to the needs of a particular data center, or cover all aspects of a Web system, for example, including the storage system which it is responsible of high energy consumption.

Therefore, it is necessary to define how to customize a web benchmark that can adapt to most existing systems and web applications in data centers for energy and performance measurement.

C. Designing a Web planning tool that allows the server to ensure performance and energy consumption both.

After obtaining some assessment tools adaptable to different needs, and once the models that have been obtained allow us to know what situations are present when a lower energy efficiency is produced and what more, how we can proceed to design a web-oriented planning tool for energy efficiency.

This tool should complement the study of the capacity that has traditionally been done and will try to predict how users access and Web applications are using the information from the data center.

The study of capacity planning has always considered the maximum number of users or tasks that can simultaneously access the system, however, designing a data center in this way is detrimental to the interests of

"Green IT" where, as already mentioned, the ideal case is to have just the machines (on) that are strictly necessary to service users connected at any given time. Therefore it is essential to combine a green policy with the performance, dependability and other issues so that you can design and plan the way in which, for example, physical resources can be turned off during periods of low activity.

To all these factors it may be added the concept of server virtualization, particularly in web systems with heterogeneous services offered. This can be used to reduce the number of physical servers to provide different services to end users that you can combine into different virtual servers within the same machine. The price to be paid in virtualization will be the resulting software overhead managing high server utilization situations.

D. Build a prototype application that can autoconfigure awarded based on performance and energy consumption.

It is important to study the feasibility and benefits of improving the adaptation of performance or workload in terms of energy consumption or energy efficiency. Portable computers and mobile phones can adjust its operation depending on the load at any time. It was first developed in such devices, as this adaptability is crucial to the life of the battery.

However, currently these schemes are being implemented also in the large-scale servers. We propose to apply such settings from the application layer, as mentioned, is where they are known more accurately, the data given in reference to performance and energy use.

This layer can identify patterns of use of the Web application, is expected on current consumption, the performance offered, and the level of charge. For example, today's processors incorporate technologies that allow a system to adjust its frequency and voltage [15][16]. It would be desirable for hardware devices to adjust their output and consumption exactly according to the performance needs of the moment, using exactly the needed energy. However, this adaptation, i.e. in the CPU, is usually not accurate and therefore is wasting energy. There are applications that adjust consumption and CPU performance according to performance needs [14] but, as mentioned, it is necessary to know as much detail as possible the actual system and Web application to decide how it can be adjusted the performance of hardware devices (not only the processor), for example inserting modules in Web applications themselves.

One way to adjust the maximum workload in a given time with the resources available would be the delay of requests in situations of intermediate load or low load to resolve later. That is, the treatment in "batch" of requests that exceed the current capabilities of the server depending on configuration, may be delayed for what, by increasing the resources available to the server, these are used on the largest possible. For example, if a server is set to operate at a level of 60% yield and energy, and has peaks at certain times, instead of increasing its capacity to serve the remaining requests would be slightly delayed, obviously until it exceeds a certain pending requests threshold, at which time it will be increased system capacity.

III. CONCLUSIONS AND FUTURE WORK

Efforts in the area of server hardware to reduce energy consumption and to improve energy efficiency have been extensively and intensively researched in recent years. These efforts have been made in different layers from the server level up throughout a data center, as well as through the middleware layer.

However, it is also necessary to create intelligent applications that self-regulate their energy consumption based on performance requirements that have to bear at any time. That is why the web servers in particular must adapt to this kind of solutions to minimize the energy to be consumed.

For this reason we first focus our efforts in learning and modelling the performance of these systems together with their consumption. Only when we know how these systems behave, we may design tools to plan the energy consumption of a Web server depending on its workload.

Without new assessment methods and energy performance tools, adaptable to any type of solution, it is also impossible to make reliable energy studies, so that, our priority is to design a customized benchmark, taking into account all aspects that influence today both the performance and consumption of a Web system, including the storage subsystems.

A future goal to research is the creation of log modules concerning the status of a Web system to auto adjusts its energy state according to the needs of the moment, offering the optimal ratio of performance per watt.

ACKNOWLEDGMENT

This work was partially funded by the Ministry of Education and Science of the Government of Spain through the project TIN2007-60440 and European Concerted Research Action Designated as Cost Action IC0804.

REFERENCES

[1] U.S. EPA, “Report to congress on server and data center energy efficiency,” *Tech. Rep.*, Aug. 2007.

[2] J. Mankoff, R. Kravets, and E. Blevis, “Some computer science issues in creating a sustainable world,” *IEEE Computer*, vol. 41, no. 8, Aug. 2008.

[3] P. Banerjee, C. D. Patel, C. Bash, P. Ranganathan. Sustainable Data Centers: Enabled by Supply and Demand Side Management. DAC’09, July 2009, San Francisco, *ACM Press*.

[4] R. Katz. Tech Titans Building Boom. *IEEE Spectrum*, February, pp. 36-43, 2009

[5] C. D. Patel and P. Ranganathan. Enterprise power and cooling. *ASPLOS Tutorial*, Oct. 2006.

[6] S. Rivoire, M. A. Shah, P. Ranganathan, C. Kozyrakis. JouleSort: A Balanced Energy-Efficiency Benchmark http://www.hpl.hp.com/environment/datacenters.html

[7] Standard Performance Evaluation Corporation (SPEC). SPEC power. http://www.spec.org/

[8] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao. Energy-aware server provisioning and load dispatching for connection-intensive internet services. In NSDI, San Francisco, CA, April 2008.

[9] HEATH, T., DINIZ, B., ET AL. Energy conservation in heterogeneous server clusters. In *Proceedings of the Symposium on Principles and Practice of Parallel Programming* (PPoPP) (June 2005).

[10] PINHEIRO, E., BIANCHINI, R., ET AL. Load balancing and unbalancing for power and performance in cluster-based systems. In Proceedings of the Workshop on Compilers and Operating Systems for Low Power (COLP) (2001).

[11] BARROSO, L. A., AND H¨OLZLE, U. The case for energy proportional computing. IEEE Computer 40, 12 (Dec. 2007), 33–37.

[12] D. Kusic, J. O. Kephart, J. E. Hanson, N. Kandasamy, G. Jiang: Power and performance management of virtualized computing environments via lookahead control. Cluster Computing 12(1): 1-15 (2009).

[13] J. Flinn and M. Satyanarayanan. Energy-aware adaptation for mobile applications. In SOSP '99: Proceedings of the seventeenth ACM symposium on Operating systems principles, pages 48–63, 1999.

[14] Miserware MiserWare ServerMiser ES.

[15] Enhanced Intel SpeedStep Technology. http://www.intel.com/cd/channel/reseller/asmo-na/eng/203838.htm

[16] AMD PowerNow. Technology. Advanced Micro Devices Inc. http://www.amd.com/us/products/technologies/amd-powernow-technology/Pages/amd-powernow-technology.aspx

[17] X. Molero, C. Juiz, M. J. Rodeño. *Evaluación y Modelado del Rendimiento de Sistemas Informáticos*. Pearson Prentice Hall. 2004.

WG3 : Adaptive actions for distributed systems

Laurent Lefèvre
INRIA - LIP Laboratory - Université de Lyon - École Normale Supérieure
46, allée d'Italie - 69364 LYON Cedex 07 - FRANCE ,
laurent.lefevre@inria.fr

Hermann de Meer
University of Passau - Germany
Hermann.deMeer@Uni-Passau.De

1 COST IC804 Working Group 3

The question of energy savings has been a matter of concern since a long time in the mobile distributed systems and battery-constrained systems. However, for large-scale non-mobile distributed systems, which nowadays reach impressive sizes, the energy dimension (electrical consumption) just starts to be taken into account.

The Working Group 3 on the European IC804 COST action (chaired by Laurent Lefèvre and Hermann de Meer) is exploring some possible actions that can be taken when energy cost is known, at the middleware, network and applications levels. The efficiency of the different actions to be undertaken will be evaluated thanks to the cost models introduced by WG2 when these become available.

This Working Group is creating several Focus Groups investigating in parallel several approaches at the different levels of a distributed system infrastructure. Two focus groups have been created : Green Wired Networks, and GPU and energy efficiency.

2 Workshop Proceedings

The following articles will show those current researches that were presented in Toulouse and Passau during the first year of the IC804 COST Action in the activities of the Working Group 3. These papers are organized in different groups :

- adaptive actions for targeted infrastructures : Clusters [6], Grids [7], Clouds[10] and Networks [11];
- energy awareness in allocation strategies [9, 4];
- energy aware resource management systems [12, 8];
- energy efficiency with Service Level Agreements [5, 1];
- targeted functionnalities [2] and scenario [3].

References

[1] R. Basmadjian, C. Bunse, V. Georgiadou, G. Giuliani, S. Klingert, G. Lovasz, and M. Majanen. FIT4Green Energy aware ICT Optimization Policies.
[2] R. Basmadjian and H. de Meer. An Approach to Reduce the Energy Cost of the Arbitrary Tree Replication Protocol.
[3] A. Berl and H. de Meer. Energy-Efficient Office Environments.
[4] D. Borgetto, G. D. Costa, J.-M. Pierson, and A. Sayah. Energy-Aware Resource Allocation.
[5] I. Brandic, V. C. Emeakaroha, M. Maurer, and S. Dustdar. Including Energy Efficiency into Self-adaptable Cloud Services.
[6] S. Contassot-Vivier, S. Vialle, and T. Jost. Optimizing computing and energy performances on GPU clusters: experimentation on a PDE solver.
[7] M. Krystek, K. Kurowski, A. Oleksiak, and W. Piateket. Energy-aware simulations with GSSIM.
[8] G. Lovasz, F. Niedermeier, and H. de Meer. Energy-Efficient Management of Physical and Virtual Resources - A Holistic Approach.
[9] M. Mazzucco. Profit-Aware Allocation Policies for Power and Performance.
[10] A.-C. Orgerie and L. Lefèvre. Greening the Clouds!
[11] C. Phillips, L. Dittmann, and M. Collier. Energy Saving in Wired Communications Networks: How and Why.
[12] J. Torres, E. Ayguad, D. Carrera, J. Guitart, V. Beltran, Y. Becerra, R. M. Badia, J. Labarta, and M. Valero. BSC contributions in Energy-aware Resource Management for Large Scale Distributed Systems.

Optimizing computing and energy performances on GPU clusters: experimentation on a PDE solver

S. Contassot-Vivier S. Vialle and T. Jost

***Abstract*—We present an experimental comparison between a synchronous and an asynchronous version of a same PDE solver on a GPU-cluster. In the context of our experiments, the GPU-cluster can be heterogeneous (different CPUs and GPUs). The comparison is done both on performance and energetic aspects.**

I. Motivations and objectives

Distributed architectures, like PC clusters, are a current and extensible solution to implement and to execute large and distributed algorithms and applications. However, modern PC clusters cumulate several computing architectures in each node. A PC cluster node has several CPU cores, each core supplies SSE units (small vector computing units sharing the CPU memory), and it is easy to install one or several GPU cards in each node (large vector computing units with their own memory). So, different kinds of computing *kernels* can be developed to achieve computations on each node. CPU cores, SSE units and GPUs have different computing and energy performances, and the optimal solution depends on the particular hardware features, on the algorithm and on the data size.

According to the algorithm used and to the chosen implementation, communications and computations of the distributed application can overlap or can be serialized. Overlapping communications and computations is a strategy that is not adapted to every parallel algorithm nor to every hardware, but it is a well-known strategy that can sometimes lead to serious performance improvements.

Moreover, although a bit more restrictive conditions apply on *asynchronous parallel algorithms*, a wide family of scientific problems support them. Asynchronous schemes present the great advantage on their synchronous counterparts to perform an implicit overlapping of communications by computations, leading to a better robustness to the interconnection network performances fluctuations and, in some contexts, to better performances [1].

So, some problems can be solved on current distributed architectures using different *computing kernels* (to exploit the different available computing hardware), with *synchronous* or *asynchronous* management of the distributed computations, and with *overlapped* or *serialized* computations and communications. These different solutions lead to various computing and energy performances according to the hardware, the cluster size and the data size. The optimal solution can change with these parameters, and applications users should not have to deal with these parallel computing issues.

LORIA, University Henri Poincaré, Nancy, France
IMS SUPELEC group, and AlGorille INRIA project team, France
AlGorille INRIA project team, France

Our long term objective is to develop auto-adaptive multi-algorithms and multi-kernels applications, in order to achieve optimal runs according to a user defined criterion (minimize the execution time, the energy consumption, or minimize the energy delay product...). However, the development of this kind of auto-adaptive solutions remains a challenge. The first step of our approach is to develop and experiment different versions of some classical HPC applications. Then, we will attempt to identify pertinent benchmarks, performance models and generic optimization rules. They will be the foundations of an auto-adaptive strategy for multi-algorithms and multi-kernels applications. The SPRAT framework [5] investigates this approach to dynamically choose CPU or GPU kernels at runtime, but considers only one computing node. We aim at being able to dynamically choose between CPU or GPU kernels and between *synchronous* or *asynchronous* distributed algorithms, according to the nodes used in an heterogeneous CPU+GPU cluster.

This article focuses on the development and experiment of a PDE solver on a heterogeneous GPU cluster, using synchronous or asynchronous distributed algorithms. We have already shown in [6] that the use of GPUs for the inner linear solver provides substantial gains. In this paper, we aim at finding the best communication scheme (sync or async) and implementation solution (CPU or GPU) according to a given context of GPU cluster (number of machines and homogeneity or heterogeneity). Both computing performances and energy consumption have been measured and analyzed in function of the cluster size and the cluster heterogeneity. Finally, different optimal solutions have been identified in this multi-parameter space: GPU cluster always appears more efficient up to 16 nodes and probably up to 33 nodes (see section A), but more experiments are required to validate this hypothesis. Moreover, depending on the respective numbers of fast nodes and slow nodes used, the most efficient solution will be either synchronous or asynchronous (see section B).

Section II introduces the synchronous and asynchronous algorithms of our distributed PDE solver, then sections III and IV introduce the heterogeneous GPU cluster we used and the experimental performance we measured. Section V summarizes our results and suggests the next steps of this research project.

II. Distributed PDE solver algorithm

Our benchmark application performs the resolution of PDEs using the multisplitting-Newton algorithm and an efficient linear solver. It is applied to a 3D transport model, described in [3], which simulates chemical species in shal-

low waters. To achieve this, the PDE system representing the model is linearized, discretized and its Jacobian matrix is computed (on the CPU). The Euler equations are used to approximate the derivatives. Since the size of the simulation domain can be huge, the domain is distributed among several nodes of a cluster. Each node solves a part of the resulting linear system and sends the relevant updated data to the nodes that need them. The general scheme is as follows:

- Rewriting of the problem under a fixed point problem formulation:
  $x = T(x), x \in \mathbb{R}$ where $T(x) = x - F'(x)^{-1}F(x)$
  $\Rightarrow$ We get $F' \times \Delta X = -F$ with F' a sparse matrix
- F is distributed over the available nodes
- Each node computes a different part of ΔX using the Newton algorithm over its sub-domain
- F is updated with the entire X vector
- X is itself updated via messages exchanges between the nodes

In this process, most of the time is spent in the linear solver required for the computation of ΔX. So, it was implemented on GPU, using the biconjugate gradient algorithm. This algorithm was chosen because it performs well on non-symmetric matrices (on both convergence time and numerical accuracy), it has a low memory footprint, and it is relatively easy to implement.

A. GPU implementation of the linear solver

As GPUs have currently a limited amount of memory, the data representation is a crucial factor which requires very special care. Thus, our sparse matrices are stored in a compact way. Moreover, the memory accesses are treated carefully. To get coalesced memory accesses, our data structures are padded so that every line of a matrix starts on a multiple of 16 elements. When coalesced reads cannot be achieved in a vector, 1D texture cache is used to hide latencies as much as possible. We also use shared memory as a cache memory whenever it is possible in order to avoid costly slower reads to the device global memory. The different kernels used in the solver are divided to reuse as much data as possible at each call, hence minimizing transfers between the global memory and the registers.

B. Synchronous and asynchronous aspects

The asynchronism is inserted in the process depicted above at the level of the data exchanges of the X vector between the inner iterations performed within each time-step of the simulation. One synchronization is still required between each time step, as illustrated in Fig. 1. At this moment, the Jacobian matrix is locally updated for the computation of the next time-step.

The communications management is a bit more complex than in the synchronous version as it must enable sending and receiving operations at any time during the algorithm. Although the use of non-blocking communications seems appropriate, it is not sufficient, especially concerning receptions. This is why it is necessary to use several threads. The principle is to use separated threads

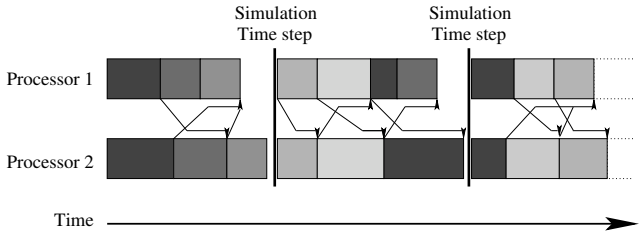


Fig. 1. Asynchronous iterations inside each time step of the computation

to perform the communications, while the computations are continuously done in the main thread without any interruption, until convergence detection. In our version, we used non-blocking sendings in the main thread and an additional thread to manage the receptions. It must be noted that in order to be as reactive as possible, some communications may be initiated by the receiving thread (for example to send back the local state of the unit).

Subsequently to the multi-threading, the use of mutex is necessary to protect the accesses to data and some variables in order to avoid concurrency and potentially incoherent modifications.

Another difficulty brought by the asynchronism comes from the global convergence detection. Some specific mechanisms must replace the simple global reduction of local states of the units to ensure the validity of the detection [2]. The most general scheme may be too expensive in some simple contexts such as local clusters. So, when some information about the system are available (for example bounded communication delay), it is often more pertinent to use a simplified mechanism whose efficiency is better and whose validity is still ensured in that context. Although both general and simplified schemes have been developed for this study, the performances presented in the following section are related to the simplified scheme which gave the best results.

III. Testbed introduction

The platform used to conduct our experiments is a set of two clusters hosted by SUPELEC in Metz. The first one is composed of 15 machines with Intel Core2 Duo CPUs running at 2.66GHz, 4GB of RAM and one Nvidia GeForce 8800GT GPU with 512MB per machine. The operating system is Linux Fedora with CUDA 2.3. The second cluster is composed of 17 machines with Intel Nehalem CPUs (4 cores + hyperthreading) running at 2.67GHz, 6GB RAM and one Nvidia GeForce GTX 285 with 1GB per machine. The OS is the same as the previous cluster. As the 8800GTs do not support double precision arithmetic, our program has been compiled with the `sm_11` flag for all the experiments.

Concerning the interconnection network, both clusters use a Gigabit Ethernet network. Moreover, they are connected to each other and can be used as a single heterogeneous cluster via the OAR management system.

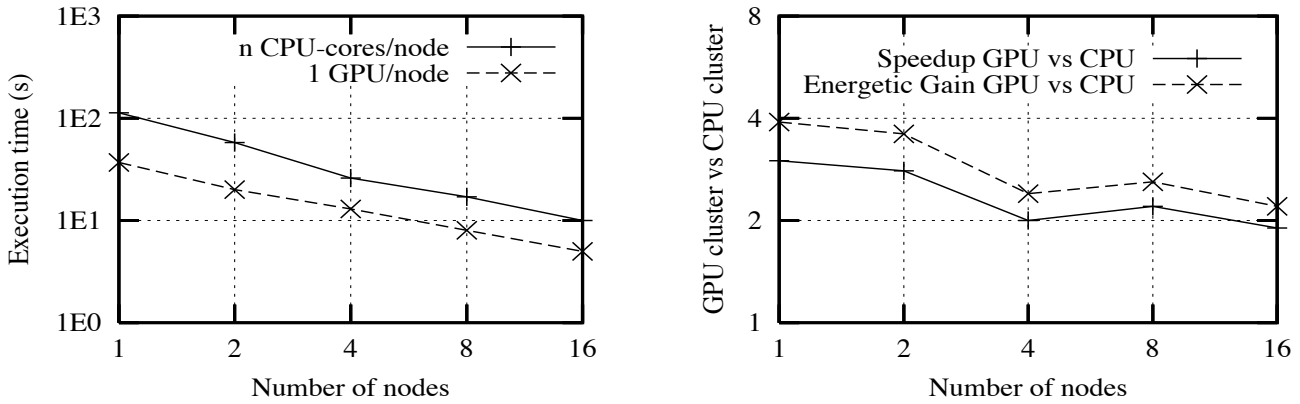


Fig. 2. Execution time of our PDE solver benchmark (synchronous version) on the multicore CPU cluster and on the GPU cluster (left), and speedup and energetic gain of the GPU cluster compared to the multicore CPU cluster (right)

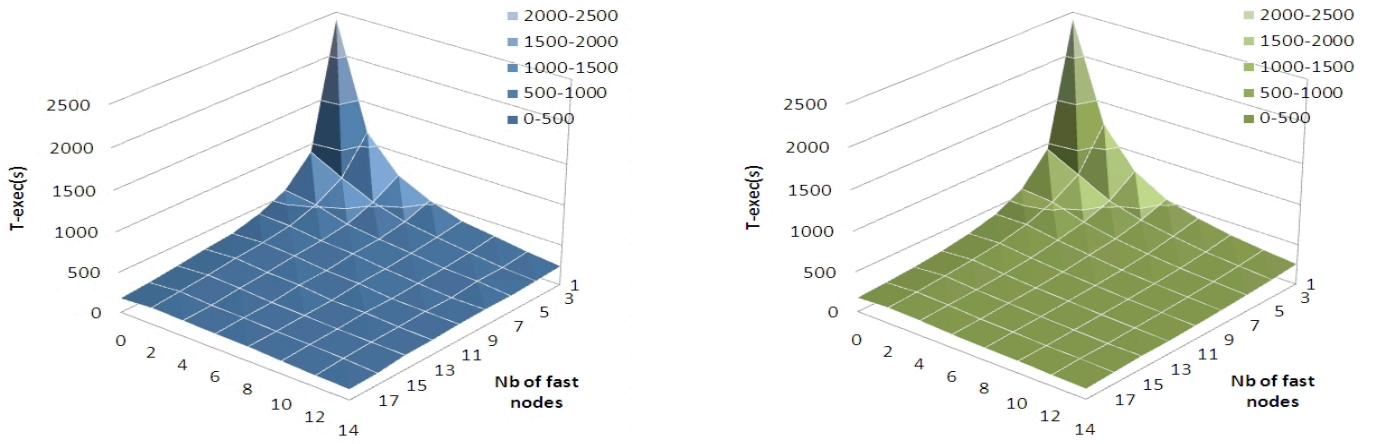


Fig. 3. Execution time of our PDE solver on a $100 \times 100 \times 100$ problem, on the heterogeneous GPU cluster, with synchronous (left) and asynchronous (right) schemes

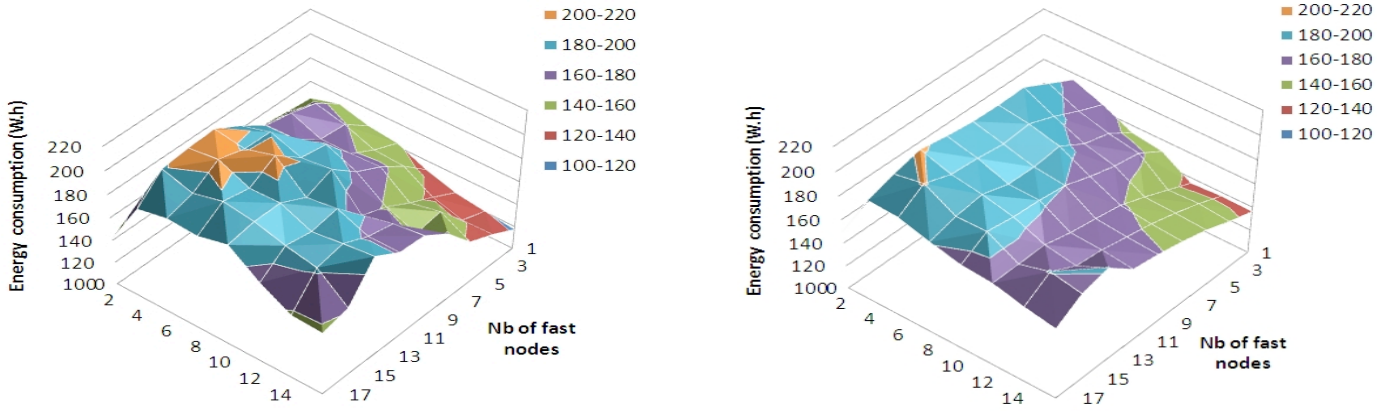


Fig. 4. Energy consumption of our PDE solver on a $100 \times 100 \times 100$ problem, on the heterogeneous GPU cluster, with synchronous (left) and asynchronous (right) schemes

IV. Experiments

A. *GPU cluster* vs *CPU cluster*

Figure 2 (left) shows the execution times of our PDE solver benchmark (in synchronous mode) using either the multicore CPUs of the cluster (all the CPU cores on each node) or using the GPUs of the cluster (one CPU core and one GPU per node). We used only the most recent nodes of our cluster, composed of Intel Nehalem CPUs and Nvidia GeForce GTX 285 GPUs, appeared on the market approximately at the same date. Our benchmark runs faster with the GPUs, scaling up to 16 nodes, and consumes less energy (not shown on Fig. 2).

However, Fig. 2 (right) shows the performance and energetic gains of the GPU cluster *vs* the multicore CPU cluster. It can be seen that substantial gains are achieved with the use of GPUs instead of CPUs. But, a slight decrease appears when the number of processors increases. This is due to the fact that computation times are smaller on GPUs whereas the inter-node communication times remain unchanged and an additional overhead is induced by the data exchanges between GPUs and CPUs. Thus, the ratio

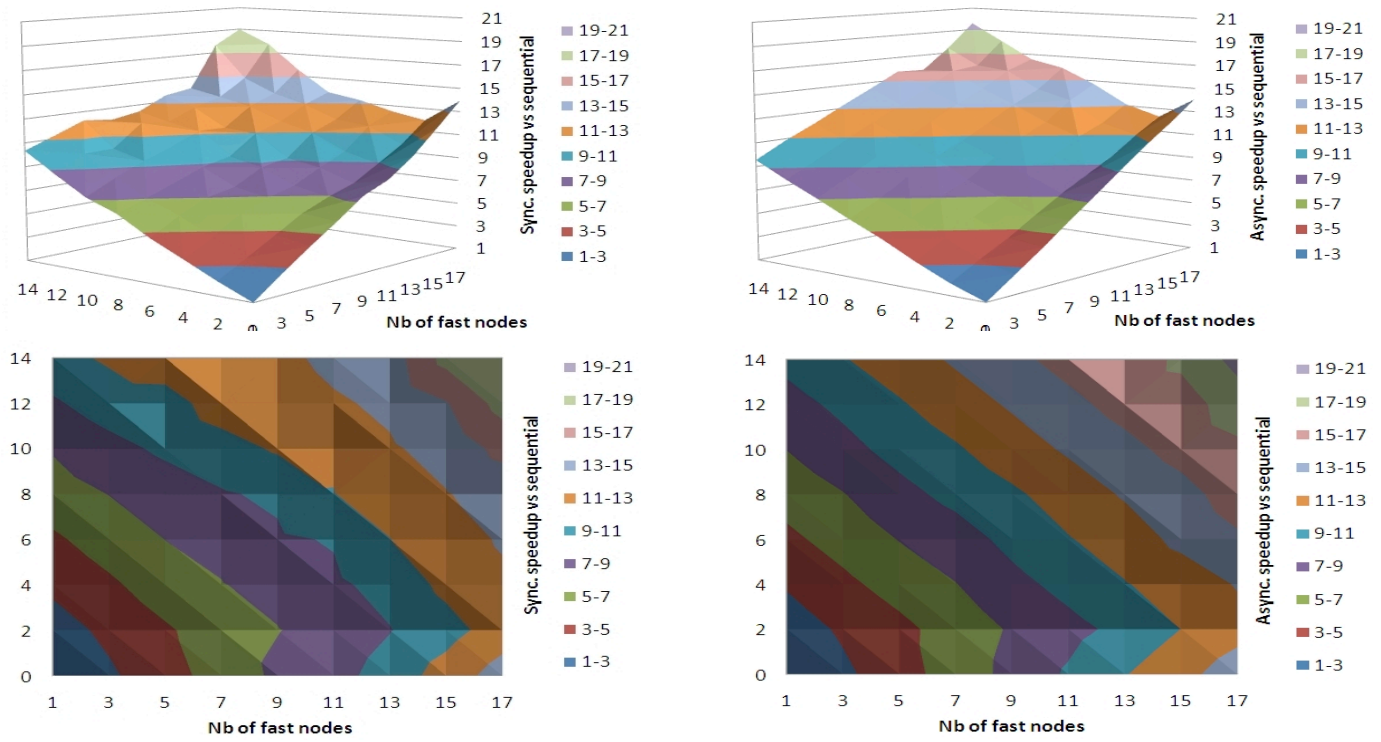


Fig. 5. Speedup of our PDE solver on a $100 \times 100 \times 100$ problem, on the heterogeneous GPU cluster, with synchronous (left) and asynchronous (right) schemes, compared to the sequential version

of communications over computations is larger on the GPU cluster. This results in a regular decrease of the speedup and energetic gain of the GPU cluster compared to the CPU cluster: GPU cluster supremacy decreases when the number of nodes increases.

B. *Synchronous* vs *asynchronous run on heterogeneous GPU cluster*

The first aspect addressed in our experiments is the evolution of the execution times according to the number of machines taken from the two available GPU clusters. As can be seen in Fig. 3, both surfaces are quite similar at first sight. However, there are some differences which are emphasized by the speedup distribution according to the sequential version, presented in Fig. 5. There clearly appears that the asynchronous version provides a more regular evolution of the speedup than the synchronous one. This comes from the fact that the asynchronous algorithm is more robust to the degradations of the communications performances. Such degradations appear when the number of processors increases, implying a larger number of messages transiting over the interconnection network and then a more important congestion. Thus, as in the previous comparison between GPU and CPU versions, the asynchronism puts back the performance decrease due to slower communications in the context of a heterogeneous GPU cluster.

In order to precisely identify the contexts of use in which the asynchronism brings that robustness, we have plotted in Fig. 6 (left), the speedup of the asynchronous GPU algorithm according to its synchronous counterpart.

First of all, it can be seen that asynchronism does not always brings a gain. This is not actually a surprise because when the ratio of communications time over computations time is negligible, the impact of communications over the overall performances is small. So, on one hand the implicit overlapping of communications by computations performed in the asynchronous version provides a very small gain. On the other hand, the asynchronous version generally requires more iterations, and thus more computations, to reach the convergence of the system. Finally, the computation time of the extra iterations done in the asynchronous version is larger in this context than the gain obtained on the communications side. That context is clearly visible on the left part of the speedup surface, corresponding to a large pool of slow processors and just a few fast processors.

As soon as the communication-times to computation-times ratio becomes sensible, which is the case either when adding processors or taking faster ones, the gain provided by the asynchronism over the communications becomes more important than the iterations overhead, and the asynchronous version becomes faster. Unfortunately, it can be observed on Fig. 6 (left) that the separation between those two contexts is not strictly regular and studying the relative speedup map would be necessary in order to deduce a general rule to apply on this kind of PDE solver.

C. *Energetic aspects*

Concerning the energetic aspect, the first point concerns the gains obtained by the use of GPUs in place of CPUs, given in Fig. 2 (right). It can be seen that those gains

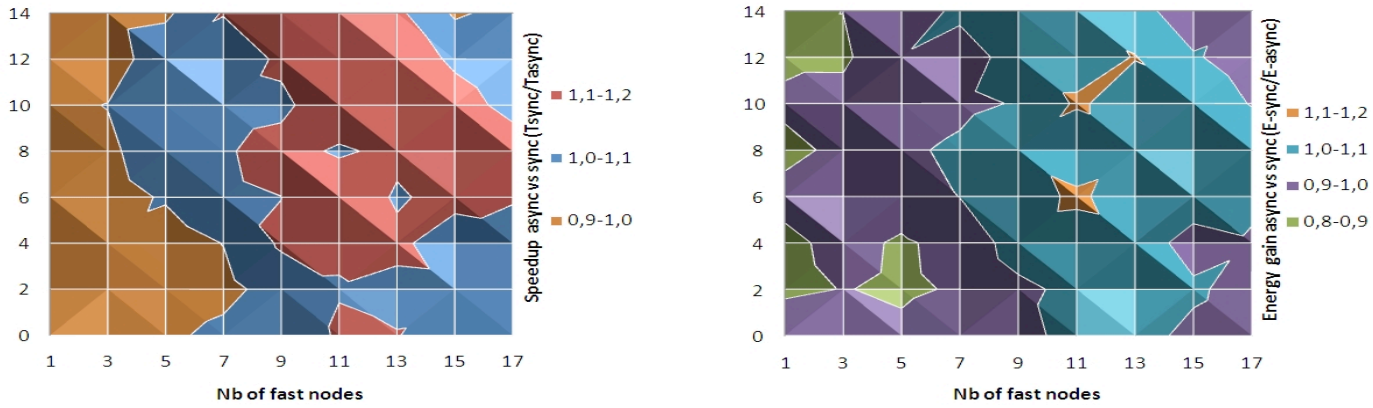


Fig. 6. Speedup (left) and energy gain (right) of asynchronous version *vs* synchronous version, on our heterogeneous GPU cluster

closely follow those of the speedup, with a nearly constant factor. That similarity of the two curves is quite obvious as the energy spent directly depends on the time of use. However, the vertical offset between the curves is a bit more surprising. This comes from the fact that the GPU and the CPU have different energy consumptions and computational powers at full load. Although the current GPUs have generally a larger consumption than the CPUs, they also have a larger computational power, and that last ratio is greater than the first one ($\frac{E_{GPU}}{E_{CPU}} < \frac{C_{GPU}}{C_{CPU}}$) at full load. Moreover, the total amount of energy used by one node is not fully spent in computations. In fact, two parts can be identified: the one actually used for the *computations* and another one for the *system* (minimal energy at idle time). So, the relative ratio of the *system* part over the *computation* part is lower when using a GPU than when using a CPU. Those two factors explain why the GPU version obtains a better energetic efficiency than the CPU one.

In order to get the complete energetic behavior of the couple algorithmic scheme - cluster, we have measured the energy consumption in function of the number of nodes and the algorithmic scheme used. The results are presented in Fig. 4. The first interesting point is that the energy consumption does not follow the performance behavior. This is explained by the performance speedup which follows a sub-linear trajectory when adding nodes in the system (see Fig. 5). Thus, multiplying the number of nodes by two does not reduce the computation time by two and the global energetic efficiency of the cluster decreases when the number of nodes increases. The second interesting point is the comparison between the synchronous and asynchronous energetic surfaces. As for the performances, asynchronism tends to be more robust as the surface is smoother and globally lower. However, here again there is no simple separation between the synchronous and asynchronous gains, as illustrated in Fig. 6 (right).

So, as for the performances, the study of such energetic gain maps will be necessary to design an optimization strategy for this kind of computing problem.

V. Conclusion and perspectives

A complete experimental study of a parallel PDE solver on a heterogeneous GPU cluster has been presented. The results show that GPUs are interesting both in terms of performance and energy consumption when the number of processors is not too high. Also, our experiments have pointed out that asynchronous algorithmic tends to bring a better scalability in such heterogeneous contexts of multi-level parallel systems, on the energetic side as well as on the performance one.

Finally, that study has also pointed out that the optimal choice of algorithmic scheme and hardware to use is not simple and requires a deeper study of the performance and energetic maps. Those results are a first step towards the design of performance and energetic models of parallel iterative algorithms on GPU clusters. In order to help this task of optimal choice, we also plan to study the *Energy Delay Product* [4] that allows to track a compromise between computational and energy efficiency.

References

[1] J. Bahi, S. Contassot-Vivier, and R. Couturier. Evaluation of the asynchronous iterative algorithms in the context of distant heterogeneous clusters. *Parallel Computing*, 31(5):439–461, 2005.

[2] J. Bahi, S. Contassot-Vivier, and R. Couturier. An efficient and robust decentralized algorithm for detecting the global convergence in asynchronous iterative algorithms. In *8th International Meeting on High Performance Computing for Computational Science, VECPAR'08*, pages 251–264, Toulouse, June 2008.

[3] J. Bahi, R. Couturier, K. Mazouzi, and M. Salomon. Synchronous and asynchronous solution of a 3D transport model in a grid computing environment. *Applied Mathematical Modelling*, 30(7):616–628, 2006.

[4] R. Gonzalez and M. Horowitz. Energy dissipation in general pupose microprocessors. *IEEE Journal of solid-state circuits*, 31(9), September 1996.

[5] K. Sato H. Takiza and H. Kobayashi. SPRAT: Runtime processor selection for energy-aware computing. In *Cluster Computing*, 2008.

[6] T. Jost, S. Contassot-Vivier, and S. Vialle. An efficient multi-algorithms sparse linear solver for GPUs. In *EuroGPU mini-symposium of the International Conference on Parallel Computing, ParCo'2009*, pages 546–553, Lyon, September 2009.

Energy-aware simulations with GSSIM

Marcin Krystek
Poznan Supercomputing and Networking Center
Noskowskiego 10
61-704 Poznan
mkrystek@man.poznan.pl

Krzysztof Kurowski
Poznan Supercomputing and Networking Center
Noskowskiego 10
61-704 Poznan
krzysztof.kurowski@man.poznan.pl

Ariel Oleksiak
Poznan Supercomputing and Networking Center
Noskowskiego 10
61-704 Poznan
ariel@man.poznan.pl

Wojciech Piatek
Poznan Supercomputing and Networking Center
Noskowskiego 10
61-704 Poznan
piatek@man.poznan.pl

***Abstract*— Growing public environmental awareness and rising energy prices led to significant interest in energy saving methods for distributed computing systems. Nevertheless, studies of impact of distributed computing on energy consumption require a large effort and are difficult to perform in real environments, especially for large-scale infrastructures. To overcome these issues we introduce the Grid Scheduling Simulator (GSSIM) - a simulation framework for automated experiments with various scheduling algorithms and distributed architectures. In this paper we present the GSSIM extension that provides a comprehensive support for energy-aware scheduling experiments including: definition of energy consumption models for resources, implementation of energy profiles that model impact of resource utilization and application performance models on energy consumption, providing power management operations, and visualization of results.**

I. Introduction

The growing importance of energy saving concept in information and communication technologies should be taken into account by the scheduling algorithms. Energy usage optimization is a subject of intensive research over last few years. The results of this research find an application in modern hardware design and software implementation of energy management policies. The existence of new hardware and software functionality and new resource interfaces which allow changing resource energy state (*on/of/sleep/stand by*) or dynamic voltage scaling (DVS) [1][2] in modern processors can be exploited by energy-aware scheduling algorithms. Due to difficult access to such large scale computing environments and to avoid hardware and software administration problems, we would like to introduce the simulation environment - *GSSIM* enabling simulations with advanced energy consumption model.

Grid Scheduling Simulator (GSSIM) is a comprehensive and advanced Grid simulation tool [3]. It is designed as a framework which enables easy-to-use experimental studies of various scheduling algorithms. The main advantage of GSSIM is a support for creating and managing multilevel schedulers. Both grid and local brokers are using functional, easy to implement and replace plug-ins, defining task allocation policies.

Configuration of the simulation environment includes detailed computing resource and network description. It is possible to define a number of parameters which characterize each resource such as a number of available processors, network link bandwidth, and resource energy consumption profile. Two standard SWF[4] and GWF[5] formats are supported for the workload description. Additionally, GSSIM supports an extended xml job description, which may contain further details such as time execution constraints, not available in standard workload archives. GSSIM provides also a workload generator tool. Synthetic workload parameters are directly defined and their value distribution is strictly controlled.

In this paper we introduce the extension of GSSIM that allows researchers to take into account energy consumption in distributed computing simulations. In particular, GSSIM provides a functionality to define energy efficiency of resources, dependency of energy consumption on resource load and specific applications, and manage power modes of resources.

II. Architecture

The main GSSIM goal is to enable researchers to effectively perform experiments that contain simulations of Grid environments. Therefore, it assumes a distributed infrastructure with multiple administrative domains (called also sites in this paper) and scheduling entities. GSSIM models two generic types of scheduling entities: Grid resource brokers and resource providers.

1) Grid broker: Grid resource broker, or Grid scheduler, is responsible for scheduling jobs to resources that belong to different administrative domains. To this end, it must interact with multiple sites including retrieving information about resources, submitting jobs, or creating reservations depending on specific settings and a type of considered scheduling problem.

2) Resource provider: A resource provider is responsible for managing resources within a single administrative domain (site). It retrieves tasks and reservation requests from Grid schedulers. Each resource provider has its local scheduler that schedules tasks to local resources. Therefore, we also use the ”local scheduler” term instead of ”resource provider” in this paper. The most common example of a site is a computing cluster under control of one of popular queueing systems, such as PBS, LSF, SGE, etc.

Having these two generic entities, GSSIM can be configured to model a large scope of architectural patterns. To this end, users may define for each Grid scheduler to which local schedulers or other Grid schedulers it can submit tasks and/or reservation requests. To define a fully decentralized architecture a user must, since local schedulers cannot interact

with each other, define couples of assigned Grid and local schedulers. Each Grid scheduler can submit tasks to one local scheduler only and to all other Grid schedulers.

Multiple scheduling strategies may be plugged into both levels. In addition to scheduling algorithms users can also define application performance models as execution time estimation plugins. Input data can be read from real traces or generated using the generator module. Results of experiments are collected, aggregated, and visualized using a statistics tool. GSSIM has a modular and plug-able architecture that enables adapting it to specific scheduling problems and users requirements.

The key elements of the presented architecture are plugins. They allow a researcher to configure and adapt the simulation framework to his/her experiment starting from modeling job performance, through scheduling policies of local schedulers, up to implementation of Grid scheduling algorithms. Information about the use of particular plugins is defined in configuration files and description of resources (since each resource provider may apply different local scheduling policies). Each plugin can be implemented independently and plugged into a specific experiment. Plugins access all required information and functionality of other components through well defined interfaces that model an abstraction of the external world. Each interface is responsible for providing some specific and dedicated functionality. In the context of energy-aware simulations and scheduling the new energy management interface is defined. It provides methods to determine detailed information about each resource and its components energy state and allows to change its current energy state. This interface is accessible from scheduling plugins, therefore energy management may be admitted as a part of whole scheduling process.

III. Energy management concept

The main aim of introducing energy management into GSSIM is to give a researcher an overview about how much energy is required to perform his/her computation. The scheduling framework in GSSIM, realized in form of easy to exchange plugins, provides an interface to obtain current resource energy state. Two available mechanisms (i) information about the way energy is consumed by the resource during simulation process and (ii) the resource administration interface allow to implement advanced energy management policies. By adoption of these mechanisms in task scheduling policies new plugins enable practical studies on energy-aware scheduling and development of energy saving strategies.

Energy management concept in GSSIM consists of the four basic elements:

A. Resource configuration

Resource configuration, provided in a simple and clear xml format, is a part of the experiment configuration input. Detailed description of the resource contains a number of properties such as CPU count, memory, CPU speed, and others, which are parts of the resource characteristics. Enabling energy management requires further information about energy states which are supported by the resource and its parts (such as CPUs), amounts of energy consumed in these states, as well as general energy profiles that provide means to calculate the total energy consumed by the resource during runtime. The above parameter categories may be defined for each element of a computing resource system as well as for the whole computing resource.

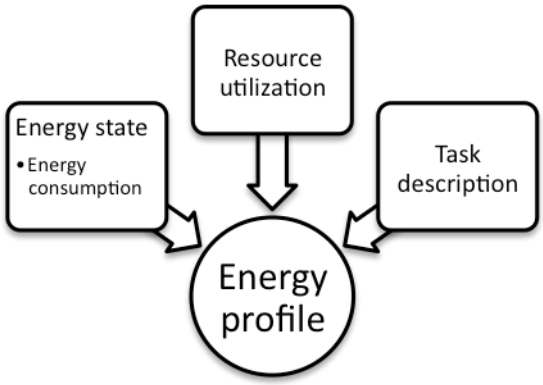


Fig. 1. Estimation of energy consumption with the energy profile

1) *Energy states* are defined separately for each component of the computing resource system such as processor, memory, disk, power adapter, etc. By default, similar to processor C-States[6], *on/off/sleep/stand by* basic states are supported. However user can define any number of new, resource specific, states. For the processor it is also possible to define frequency levels, so called P-States[9], in which processor can operate with specific power usage levels. Changing energy state of each computing resource component is part of the energy management policy and, due to further energy consumption definition, affects overall energy used by the computing resource.
2) *Energy consumption* is directly connected with energy state and describes average power usage by the resource working in current state. For the processors which can operate on more than one frequency level, such description must be provided separately for each level.
3) *Energy profile* definition of the amount of energy used used by the resource working in one of the predefined energy states is static and does not take resource utilization into the consideration. Resource energy profile, accomplished as an implementation of predefined interface, should perform calculations which estimate direct and close to real amount of used energy. Basic implementation will only sum up energy used by all components of the computing resource system according to their current state and adequate energy consumption definition. However, detail analysis of currently running tasks allows to determine exactly which components

of the computing system are used and in what way. Involving resource utilization in energy consumption calculations allows to highlight differences in amount of energy required to execute various types of tasks, e.g. CPU intensive and data intensive.

Relation between above elements is illustrated in Figure 1. Energy profile estimates energy consumption based on information about resource energy consumption levels, resource utilization, and application performance model.

B. Resource management

Information provided by the resource configuration, both static resource description and values calculated by the resource energy profile, can be used to perform advanced resource management. GSSIM provides specialized interfaces on global (grid) and local (resource) scheduling level, which allows to gain detail information about computing resource components and change their energy states. Accordingly, two types of methods are available: *getResourceComponent()* - to obtain information and *changeResourceComponentState(State)* to manipulate current state of the resource component. For the processors it is also possible to dynamically change the frequency level of single processor. The activities performed with this interface finds a reflection in total amount of energy consumed by the resource during simulation. Availability of these interfaces in global and local schedulers allows to implement different strategies such as centralized energy management, self-management of computing resources and mixed models.

C. Energy aware scheduling

Presence of detailed resource usage information, current resource energy state description and functional energy management interface enables an implementation of energy-aware scheduling algorithms. Resource energy consumption become in this context an additional criterion in scheduling process. There are various approaches to decrease energy consumption and they usually apply the following technics:

- minimize number of used computing resources
- move tasks between resources to reach full load on one resource and zero load on the other or to balance the load
- turn off unused resources
- turn off unused processors
- slow down processors, by cutting down their frequency

D. Analysis of results

It is expected, that energy management process and efficiency of used policies will be subject to further analysis after experiment is performed by GSSIM. Therefore detailed data about each resource component state and the energy consumed by it is collected. To ensure appropriate level of details each change of the resource component energy state is logged along with time stamp. Additionally, each value returned by the resource energy profile is also logged. Based on the collected data, GSSIM calculates minimum, maximum and basic aggregates like average amount of energy consumed by the resource. The results are also visualized in the form of timeline graph, which allows to observe variability of the resource energy consumption. Example of energy consumption analysis is illustrated in Figure 2. It shows power usage in time so the whole area under the plot reflects a total energy consumption.

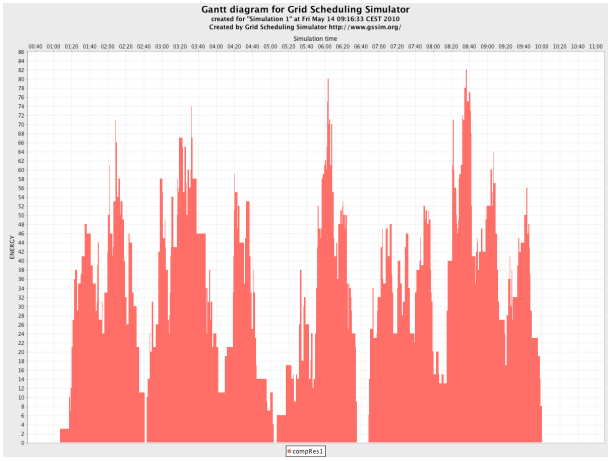


Fig. 2. Energy consumption chart

IV. Energy consumption models

Energy consumption models allow to introduce the energy consumption, a significant aspect of the high performance real-life processing, to the GSSIM simulation environment. The main aim of the models is to emulate the behavior of the real computing resource and the way it consumes energy. Due to reach functionality and flexible environment description, GSSIM can be used:

- to develop new energy consumption models. This is particularly useful to verify theoretical assumptions. Possibility of repeating the experiment in the same conditions unlimited number of times gives a researcher ability to find and test the best model and compare received results to simulation experiment.
- to develop energy management strategies - which, based on the existing models, trying to reduce overall energy consumed by the computing resources.

Energy consumption models provided by default in GSSIM, can be divided because of its accuracy, generality and complexity into following groups:

A. Constant

Constant model calculates total amount of energy consumed by the computing resource system as a sum of energy consumed by all its components (processors, disks, power adapters, etc.). This model does not take resource (component) utilization into consideration, however it follows resources energy states changes and respects amounts of energy defined for each state. The greatest value of the constant model is its simplicity, therefore it may be used as a base line, kind of reference for other utilization based models.

B. CPU utilization

CPU utilization model relies on assumption, that total amount of energy consumed by the computing resource is a function of energy used by its processors. This model ignores parameters and usage of other resource components (like disk, memory). In particular case, it may assume, that processors are the most significant energy consumption element and therefore skip existence of other parts of the computing system.

This model is also the first, which uses resource utilization as a base of its calculations. Unlike previous constant model, CPU utilization expands static energy state description and includes realtime usage of the processor. This allows to distinguish amount of energy used by the processor which is turned on, but doing nothing and the one which is turned on and full loaded.

C. Application specific

Application specific model is a generalization of CPU utilization model to all computing resource components. It removes the greatest drawback of the previous model, which is hidden and quite hard to follow and understand method of expressing all resource components usage as a function of processor usage. Application specific model considers all defined system elements (processors, memory, disk, etc.) as significant in total energy balance. It also assumes that each of these components can be stressed in different way during the experiment, thus have different impact in total energy consumption. The information about how resource will be used by task must be derived from the task description, therefore basic task processing is required. Because of availability of task characteristics and current resource components utilization, the application specific model can express differences in amount of energy required for executing various types of task, e.g. differences between CPU intensive and data intensive tasks.

V. Summary

GSSIM allows researcher to effectively perform unlimited number of experiments, providing equal conditions for each experiment execution. Until now many valuable experiments following GSSIM scheduling concept and architecture have been performed. Introducing energy management expands possible applications of GSSIM by allowing to track amounts of energy used to perform virtual computation. The greatest advantage of GSSIM energy management architecture are however resource energy consumption models. Easy to implement, replace and reuse, enable researcher to look for the models which suits best and perfectly reflects real computing resource behavior. On the other hand, existing models may be used to perform energy-aware scheduling and in this way make a significant contribution in saving energy.

References

[1] M.Weiser,B.Welch,A.J.Demers,andS.Shenker - Scheduling for reduced CPU energy. In Operating Systems Design and Implementation, pages 1323, 1994.

[2] T. Burd and R. Brodersen - Design issues for dynamic voltage scaling. In Proc. International Symposium on Low Power Elec- tronics and Design, pages 914, July 2000.

[3] Kurowski, K., Nabrzyski, J., Oleksiak, A., Wglarz, J. (2008). Grid Scheduling Simulations with GSSIM. In Proceedings of the International Conference on Parallel and Distributed Systems, 2:18, IEEE.

[4] Parallel Workload Archive. http://www.cs.huji.ac.il/labs/parallel/workload/

[5] Grid Workloads Archive. http://gwa.ewi.tudelft.nl/

[6] http://software.intel.com/en-us/blogs/2008/03/27/update-c-states-c-states-and-even-more-c-states

[7] http://software.intel.com/en-us/blogs/2008/03/12/c-states-and-p-states-are-very-different

[8] http://www.intel.com/technology/itj/2006/volume10issue02/art03_Power_and_Thermal_Management/p03_power_management.htm

[9] http://software.intel.com/en-us/blogs/2008/05/29/what-exactly-is-a-p-state-pt-1

[10] The Grid Scheduling Simulations Portal, http://www.gssim.org

[11] S. Rivoire, P. Ranganathan, C. Kozyrakis - A Comparison of High-Level Full-System Power Models 2008r.

Greening the Clouds!

Anne-Cécile Orgerie and Laurent Lefèvre
INRIA RESO - Université de Lyon - École Normale Supérieure
46, allée d'Italie - 69364 LYON Cedex 07 - FRANCE ,
annececile.orgerie@ens-lyon.fr, laurent.lefevre@inria.fr

Abstract

The question of energy savings has been a matter of concern since a long time in the mobile distributed systems and battery-constrained systems. However, for large-scale non-mobile distributed systems, which nowadays reach impressive sizes, the energy dimension (electrical consumption) just starts to be taken into account.

In this paper, we present Energy-Aware Reservation Infrastructure (EARI) which is a Grid framework that manages the Grid resources in an energy-efficient way. Then, new technologies are studied such as virtualization and live-migration of virtual machines in terms of power consumption. This study leads us to propose an energy-efficient Cloud framework: Green Open Cloud (GOC) and to validate it through experimentations with different scenarios.

Keywords: **Energy efficiency, large-scale distributed systems, Grids, Clouds, live migration of virtual machines, green computing, energy-awareness.**

1 Introduction

The question of energy savings has been a matter of concern since a long time in the mobile distributed systems and battery-constrained systems. However, for large-scale non-mobile distributed systems, which nowadays reach impressive sizes, the energy dimension (electrical consumption) just starts to be taken into account.

Energy consumption of data centers worldwide doubled between 2000 and 2006 [5]. Incremental US demand for data center energy between 2008 and 2010 is the equivalent of 10 new power plants [5]. These alarming figures lead to think about new technologies and infrastructures in order to increase the energy efficiency of large-scale distributed systems such as Grids and Clouds.

Section 2 presents our Energy-Aware Reservation Infrastructure (EARI) for Grids. In Section 3, we study the cost of virtual machines while executing basic operations (boot, run, halt) and live migration. This analysis leads us to propose the Green Open Cloud (GOC) framework in Section 4. An experimental validationn of GOC is conducted in Section 5, before concluding in Section 6.

2 Adapting a Grid Energy-Aware Reservation Infrastructure (EARI) to Clouds requirements

This work was part of the INRIA ARC Green-Net initiative[1] [1].

The main leverage to make large-scale distributed infrastructures more energy-efficient is to reduce energy wastage. Indeed, resources are always fully powered on even when they are not used. So, Grids require energy-aware frameworks capable of switching of unused resources without impacting user applications in terms of both performance and usage. That's why, we propose the Energy-Aware Reservation Infrastructure (EARI) [8, 7, 3].

The main features of EARI are to:

- Switch off the unused computing resources;
- Predict the next use;
- Aggregate the reservations by giving green advice to the users.

EARI is devoted to Grid infrastructures that support in-advance reservations: the users submit reservation requests. They specify the duration they want, the number of resources and the wished start time. When a reservation is accepted, the scheduler inscribes it on its agenda and cannot move it after.

[1] Green-Net initiative (http://www.ens-lyon.fr/LIP/RESO/Projects/GREEN-NET/) lasted from 2008 to 2009 and involved four partners: INRIA RESO project-team in Lyon (France), INRIA Mescal project-team in Grenoble (France), IRIT in Toulouse (France) and Virginia Tech (USA).

To aggregate the reservations, the idea is to propose several possibilities to the user instead of just accepting or refusing their request. EARI proposes to the user to put its reservation after or before a reservation which is already in the agenda. If the user accepts, he/she will avoid on/off cycles and thus save the corresponding energy.

Figure 1 presents the architecture of EARI. It is composed of a classical Grid infrastructure: users, a portal, a scheduler and resource manager, and the Grid resources. However, it is also composed of energy-aware components: a set of energy sensors plugged to the resources and an energy-aware manager which is responsible for applying the green policies of EARI.

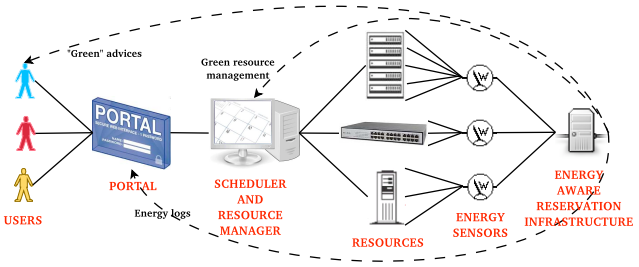


Figure 1. Architecture of EARI

In order to increase the energy-efficiency, EARI embeds also some prediction algorithms. They are used at the end of each reservation to know if the freed resources will be used soon (and so should remain on) or not (and so should be switched off). These prediction algorithms are presented and evaluated in [8]. The global EARI is evaluated in [7] by simulating a replay of one year of logs of Grid'5000, a french experimental platform. On the Lyon site of Grid'5000 (150 nodes), we have deployed energy sensors that fully monitor the site [2]. This provides an experimental testbed where we can test our frameworks with real energy measurements.

Among Cloud's most famous features are: virtualization, accounting, scalability, reliability and security. The Resources as a Service (RaaS) philosophy leads to a more flexible management of the physical nodes: Clouds provides a strong isolation that allows users to share the same physical resources. Thus, this strong virtual machine (VM) isolation can also leads to energy savings. Indeed, physical resources can be more exploited by doing workload consolidation.

So, we would like to adapt EARI to Clouds environments in order to benefit from Cloud's features. Yet, some differences between Grids and Clouds have to be taken into account:

- agenda (no reservations in advance in current Cloud infrastructures),
- virtualization,
- possibility to use live migration,
- usage and thus predictions,
- resource management.

Cloud computing seems to be a promising solution to the increasing demand of com- puting power needed by more and more complex applications. However, the studies often lack real values for the electric consumption of virtualized infrastructures.

3 Energy cost of virtual machines

Our experimental Cloud consists of HP Proliant 85 G2 Servers (2.2 GHz, 2 dual core CPUs per node) with XenServer 5.0[2] installed on them. Each Cloud node is linked to an external wattmeter that logs the power consumption of the node each second. These energy logs are sent to an energy data collector which stores them. The energy data collector is linked to the Cloud portal, so users can access these logs. The measurement precision of our experimental platform is 0.125 watts, and the maximum frequency is one measure per second, which is precise enough to have a good idea about the consumption of virtual machines.

Figure 2 shows the energy profile of a typical virtual machine usage: at $t = 10$, the VM is started and is booting until $t = 30$, then a *cpuburn*[3] is running on the VM from $t = 40$ to $t = 100$, and finally, the VM is shutting down from $t = 110$ to $t = 122$.

One can notice that the average power consumption from $t = 30$ to $t = 40$ is equal to the idle consumption (P_{idle}) which is the consumption of the node when it does nothing. So, an idle VM (with nothing running on it) does not consume energy. Furthermore, the boot and the shutdown consume really less energy than a *cpuburn*.

In Figure 3,six cpuburn on six different VMs on Cloud node 1 are launched one by one. The first starts at $t = 10$, the consumption increases to 209 watts. Then the second starts and the consumption reaches 230 watts. The third starts, and the node consumes 242 watts. The fourth leads to 253 watts. The apparition of the fifth and the sixth jobs does not increase the consumption. Indeed, as the jobs are CPU intensive (cpuburn uses 100% of a CPU capacity) and as there are only four cores on the node (2 dual core CPUs), they are fully used with the first four VMs. The fifth VM appears as free in terms of energy cost because it shares already fully used resources. Obviously, these energy-free VMs have a cost in terms of performances.

[2]XenServer is a cloud-proven virtualization platform that delivers the critical features of live migration and centralized multi-server management (`http://citrix.com/English/ps2/products/product.asp?contentID=683148`).

[3]cpuburn is a software designed to apply a high load to the processor (`http://pages.sbcglobal.net/redelm/`).

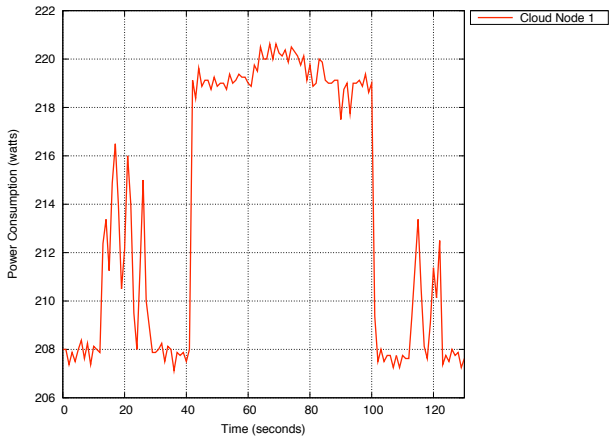


Figure 2. Power consumption of boot, cpuburn and halt of a VM

Each cpuburn job lasts 300 seconds (Fig. 3). At $t = 110$, the migration of the 6 VMs from Cloud node 1 to Cloud node 2 is launched. The migration requires sustained attention from the hypervisor that should copy the memory pages and send them to the new host node. So, each cpuburn ends 5 seconds late due to the VM's migration.

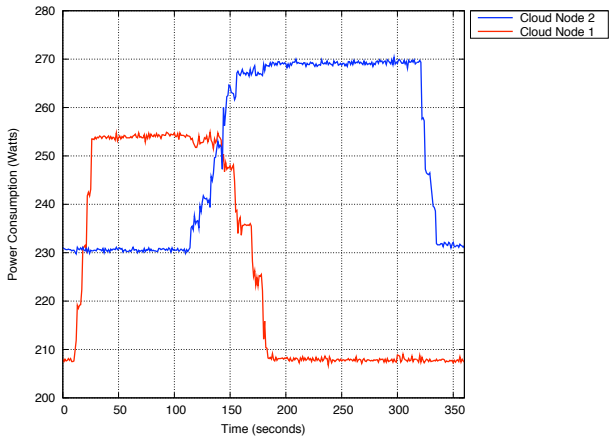


Figure 3. Power consumption during a live-migration

These analyses lead us to propose the Green Open Cloud (GOC) framework to manage Cloud resources in an energy-efficient way. Among the components of a Cloud architecture, we have decided to focus on virtualization, which appears as the main technology used in these architectures. We also use migration to dynamically unbalance the load between the Cloud nodes in order to shut down some nodes, and thus to save energy.

4 Green Open Cloud

As EARI, GOC supports the "do the same for less" approach, and deals with energy efficient on/off models combined with prediction solutions [6, 4].

The main features of GOC are to:

- switch off unused resources,
- predict usage,
- aggregate reservations,
- green policies for the users,
- network presence proxy.

GOC's framework embeds a network presence proxy to deal with the Cloud resource monitoring tools: a node which has been switched off to save energy should not be considered as dead. So, the network presence proxy embeds the basic services of the switched off resource and answers instead of it to the resource manager. The node is switched on again when it is required.

GOC's architecture is presented by Figure 4

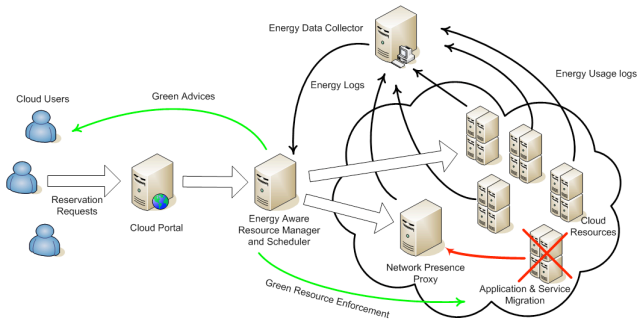


Figure 4. Architecture of GOC

When a user frees some virtual machines, a consolidation algorithm is used to aggregate the remaining VMs on the smaller number of nodes. This consolidation process is launched in coordination with predictions algorithms in order to avoid to switch off physical resources that will be required just after. GOC provides also green advices to the users as EARI in order to aggregate the reservations in time. This double aggregation in time and space is the core functionality of GOC.

The resource manager is a key component in a Cloud infrastructure. To be compatible with the broader range of resource management system, GOC's resource manager is built as an overlay of existing Cloud's resource manager. The components of GOC's resource manager are the green boxes on Figure 5. The yellow ones are the usual Cloud's resource manager ones.

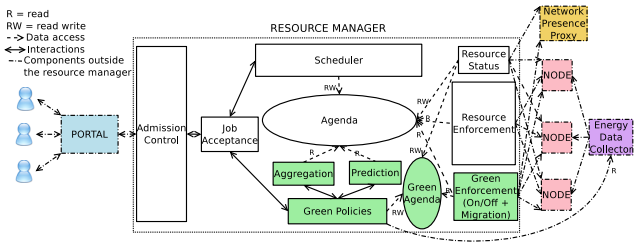


Figure 5. Architecture of GOC's resource manager

5 Experimental validation

Our Cloud platform consists of HP Proliant 85 G2 Servers (2.2 GHz, 2 dual core CPUs per node). XenServer 5.0 is installed on each node.

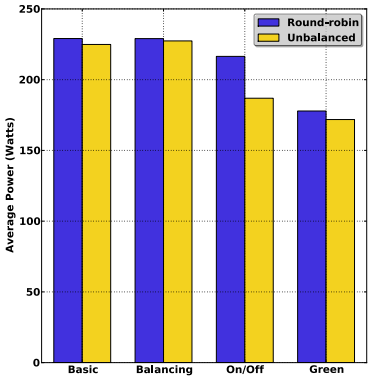


Figure 6. Comparison between the scenarios

We have tested two different schedulings: round-robin and unbalanced to show the adaptability of GOC to any kind of Cloud resource manager. Four scenarios are used to compare GOC with other classical resource management systems:

- *basic*: nothing is changed.
- *balancing*: migration is used to balance the load between the Cloud nodes.
- *on/off*: the unused nodes are switched off.
- *green*: the unused nodes are switched off and migration is used to unbalance the load between Cloud nodes. This allow to aggregate the load on some nodes and switch off the other ones. This is the scenario that corresponds to GOC.

Each scenario is launched on a Cloud job arrival example for each scheduling. All the results are provided in [4]. Figure 6 presents the average consumption for these 8 experiments.

As expected, the green scenario is the less consuming. With the unbalanced scheduling, it consumes 25% less energy than the basic scenario.

6 Conclusion and perspectives

After proposing EARI for an energy efficient management of Grid resources, we have proposed the first analysis of energy usage of VM in Cloud context. This analysis leads us to propose GOC: an energy-aware Cloud framework. Real tests have been launched and show that 25% of energy is saved with GOC on a basic Cloud usage example. This proves that significant energy savings are achievable. GOC can be integrated in current and future Cloud infrastructures (with reservations, accounting, etc.).

References

[1] G. Da-Costa, M. Dias de Assuncao, J.-P. Gelas, Y. Georgiou, L. Lefèvre, A.-C. Orgerie, J.-M. Pierson, O. Richard, and A. Sayah. Multi-facet approach to reduce energy consumption in clouds and grids: The green-net framework. In *e-Energy 2010 : First International Conference on Energy-Efficient Computing and Networking*, Passau, Germany, Apr. 2010.

[2] M. Dias de Assuncao, J.-P. Gelas, L. Lefèvre, and A.-C. Orgerie. The green grid5000: Instrumenting a grid with energy sensors. In *5th International Workshop on Distributed Cooperative Laboratories: Instrumenting the Grid (INGRID 2010)*, Poznan, Poland, May 2010.

[3] L. Lefèvre and A.-C. Orgerie. Towards energy aware reservation infrastructure for large-scale experimental distributed systems. *Parallel Processing Letters - Special Issue on Clusters and Computational Grids for Scientific Computing*, 19(3):419–433, Sept. 2009.

[4] L. Lefèvre and A.-C. Orgerie. Designing and evaluating an energy efficient cloud. *The Journal of SuperComputing*, 51(3):352–373, Mar. 2010.

[5] McKinsey & Company. Revolutionizing data center efficiency. Technical report, 2009.

[6] A.-C. Orgerie and L. Lefèvre. When clouds become green: the green open cloud architecture. In *Parco2009 : International Conference on Parallel Computing*, Lyon, France, Sept. 2009.

[7] A.-C. Orgerie, L. Lefèvre, and J.-P. Gelas. Chasing gaps between bursts : Towards energy efficient large scale experimental grids. In *PDCAT 2008 : The Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies*, pages 381–389, Dunedin, New Zealand, Dec. 2008.

[8] A.-C. Orgerie, L. Lefèvre, and J.-P. Gelas. Save watts in your grid: Green strategies for energy-aware framework in large scale distributed systems. In *ICPADS 2008 : The 14th IEEE International Conference on Parallel and Distributed Systems*, pages 171–178, Melbourne, Australia, Dec. 2008.

COST IC0804 WG3: Adaptive Actions for Distributed Systems

Energy Saving in Wired Communications Networks: How and Why

Chris Phillips, Lars Dittmann and Martin Collier

***Abstract*— Information and Communication Technology currently accounts for about 4% of the world's energy consumption. With greater Internet uptake forecast, this demand is expected to increase. Perhaps influenced by socio-economic and political pressures Operators are now exploring means of using their networks more efficiently.**

It is readily observed that traffic load varies significantly over a daily cycle; this gives considerable opportunity for savings by better matching energy consumption with the immediate information transfer demands of end-users. This paper reviews the motivation for saving energy in wired communication networks and introduces a potential framework that could be used to support dynamic energy management.

***Index Terms*—Energy Efficient Networking, Reducing Power Consumption, Sleep Modes**.

I. Introduction

ICT is directly responsible for about 3-4% of the carbon emissions to-date. Changing political, environmental and economic circumstances are inducing commercial organisations to consider operating their equipment in a more energy-efficient manner. This is already the case with data centres where environmental stabilization costs and power supply limitations pose particular challenges. For example, 50% of energy consumed by data-centres is devoted to power and cooling infrastructure [1].

In terms of the networking equipment itself, manufacturers like Juniper propose that networking equipment should be subjected to an energy rating similar to that used for domestic "white goods", in the guise of the Energy Consumption Rating Initiative. Another example is the Alliance for Telecommunications Industry Solutions (ATIS) who introduce the Telecommunications Energy Efficiency Ratio (TEER) for measuring the network-element efficiency of telecommunication equipment. Indeed, manufacturers are already considering measures that would enable their equipment to operate more efficiently, by altering the speed of cooling fans, selectively powering down layers of the internal switch fabric and so forth. Indeed around a 60% reduction in energy consumption is achievable via sleeping and rate-adaptation for lightly utilized networks (10-20%) provided the equipment is capable of millisecond changes [2] by exploiting traffic variability [3]. As well as the environmental benefits, reducing energy consumption saves money. Japan uses over 50TWh each year on Telecoms [4] as shown in Fig 1.

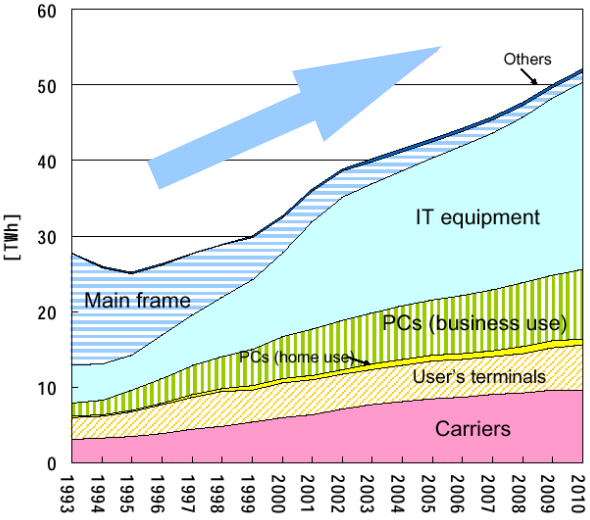


Figure 1: Energy Trends for Information and Communication

Exploring energy efficiency measures in wireless networks and data centres has received considerably more attention than the wired communications infrastructure. Nevertheless, as an example, with BT's network assuming a 22Tb/s traffic load, moving to all-optical network is estimated to translate into OPEX savings of up to 93% [5], although this figure includes factors such as rack space and labour in addition to power consumption costs. Replacing an electronic switching fabric and buffering with photonic-based equivalents may only reduce energy consumption by a modest 9% [6].

Despite these potential savings it must be borne in mind that commercial telecommunications providers operate very conservative regimes in the sense that they are risk averse. Thus, whatever energy-saving schemes are proposed, to be adopted in a business context, they must do nothing to degrade Quality of Service, availability and other Service Level Agreement metrics. Balancing these caveats are an awareness that it is "good to be green" both in terms of the perceived company profile as well as the economic benefits.

Chris Phillips is with Queen Mary, University of London, Mile End Road, London, E1 4NS, UK. (e-mail: chris.phillips@elec.qmul.ac.uk). Lars Dittmann is with the Institut for Fotonik, Danmarks Tekniske Universitet, 2800 Kgs. Lyngby, Denmark (e-mail: ladit@fotonik.dtu.dk). Martin Collier is with Dublin City University, Dublin 9, Ireland (martin.collier@rince.ie).

II. Background

From a technological perspective a number of researchers have considered the energy consumption benefits of exploiting features of optical networks. For instance, power-aware Routing and Wavelength Assignment (RWA) in optical networks can make large savings compared with normal load-spreading techniques [7] though this may raise issues of increased connection blocking. More radical examples include contributions like the CANARIE proposal of a next generation Internet to reduce global warming where a dense optical network connects massively distributed virtual routers [8]. The FP7 BONE project is also exploring how optical technologies can improve the energy efficiency of networks [9]. Using an IP over optical layered network architecture, efficient multilayer traffic engineering has been estimated to save about 10% energy [10].

Other work has focused on adapting existing architectures, by adding functionality to equipment to better exploit the natural variation that exists in the traffic load characteristics. [11] show that over a 20% energy saving can be potentially achieved when nodes and links in the core network are turned off during off-peak periods, though they only consider highly connected synthetic topologies. Certainly, power aware protocols can be used for routing, by putting components to sleep, adjusting rates. By doing so, substantial savings are obtainable [12]. However, if we just control the infrastructure and let the routing protocol respond to the resulting topology changes this could lead to unacceptable disruption. [13] [14] look at avoiding transient loops during the convergence of link-state routing protocols.

A different approach has been taken by other researchers. Since the access network and the devices attached to it constitute a major source of energy consumption, it can be argued that activities to reduce the energy consumption should focus on access networks first [15]. Clearly these approaches capitalise on the appreciable accumulated gains obtained when large numbers of devices are considered. However, they raise an interesting paradox that the savings would not necessarily reward the network operator as the Customer Premises Equipment (CPE) power supply is provided and financed by the customer.

Rather than focussing on the CPE device hardware alone researchers have also considered possible energy savings at the application layer, e.g. sleep/wake-up protocols for peer-to-peer file sharing, too [16]. Rather than continuously streaming packets across a network, a more energy efficient approach might be to bulk traffic into bursts and allow the IT infrastructure to sleep between these discrete burst transmission events.

III. Energy Management Architecture

Studies of the core network energy savings typically focus on individual equipment. However this is unlikely to produce the best possible cost savings. It is well understood that globally optimal configurations do not necessarily arise if local devices seek to optimise their own situation. In fact, to achieve globally optimal network configurations in terms of reduced energy consumption is likely to require some devices to be burdened to a greater degree in order to permit other devices to be "switched off". This is particular true as the energy consumption of a lightly loaded interface is little different from one that is heavily loaded. Appreciable savings only tend to materialise if link interfaces or entire nodes can be deactivated.

Traffic flowing across a network varies across multiple timescales, including generally regular, daily variations. The extent of these fluctuations is influenced by the location of the links within the topology and the traffic demand matrix. For certain "trunk" links the extent of the variation may be marginal. Closer to the CPE the variation tends to be more pronounced. However, for commercial networks there is little public-domain data available regarding the per-link daily load variations, or even the topologies themselves. This information is regarded as too sensitive. Nevertheless data pertaining to specific links or averaged over multiple links is available and suggests that the daily variation in load may be considerably in excess of 20%.

Dynamic power-consumption regulation mechanisms may be devised that can capitalise on these fluctuations, to permit equipment to be configured in an energy-efficient way, whilst delivering appropriate service. However, superimposed on this roughly sinusoidal daily load characteristic, significant fluctuations are observed, even over intervals of a few milliseconds. This poses a particular challenge to a coordinated energy management scheme. For example, a centralised approach, co-locating the energy-regulation controller with an operator's Network Management Station (NMS) may appear desirable; however, this increases signalling latency that may limit actions to responses to larger timescale traffic variations. It suggests that fine-grain energy management would need to be decentralised.

By coordinating power management decisions and by better tracking actual traffic demand, energy consumption may be significantly reduced since it is no longer tied to the peak capacity of hardware. The resulting architecture could feature some or all of the following:

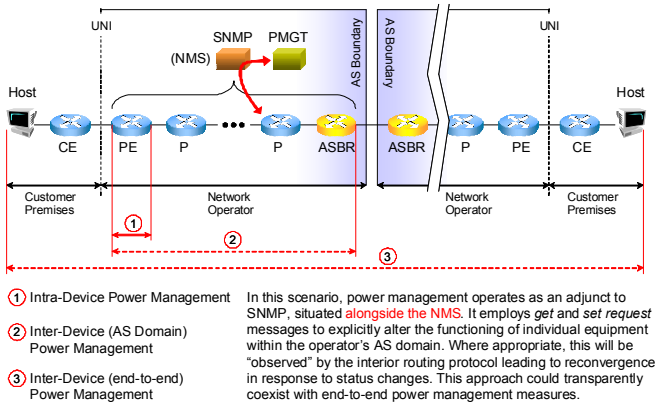
- A mechanism to collect the metrics and an overall control algorithm to orchestrate responses
- Mechanisms for changing network configuration by interacting with existing systems for management. For example, tools already allow an operator to switch individual devices on and off (i.e. SNMP for configuration)
- Interaction with, or enhancement of, existing interior routing protocols to include energy conservation as a routing constraint
- Adaptation mechanisms that permit end-user equipment to interact with network elements

With such an architecture in place the goals would be to:

- Examine how line rate scaling, switch fabric changes etc affect the device throughput and the corresponding energy savings
- Explore how actual / predicted traffic information can be exploited by power-aware routing / configuration
- Identify the risks and quantify the benefits.

An example architecture is presented in Fig 2. In this case energy management is considered as operating across several geographical regions. Within the smallest scope intra-device energy management would be used to respond to transient fluctuations in load using "sleep" states. These decisions would be carried out in a localised manner and buffering should be sufficient to accommodate any reconfiguration delay. Across the scope of an AS domain, energy management is shown as acting as an adjunct to SNMP with a centralised unit being co-located with the NMS. This Power Management (PMGT) entity would possess global awareness and use an algorithm to determine a suitable overall configuration so as to save a (near) optimal amount of energy. However, using SNMP to implement these configuration changes, such as by disabling interfaces, and relying on an existing routing interior gateway routing protocol to redistribute the traffic is unlikely to be acceptable. During reconvergence service disruption would likely be unacceptable. The means by which dynamic reconfiguration is implemented and under what circumstances, requires further investigation.

The final scope would be to explore schemes that operate end-to-end. However, given the lack of trust between Operators and the financial models they use, it is unlikely that they would cooperate to provide inter-AS energy management. Nevertheless, it is feasible that end-system applications could be modified so the traffic they inject into the network is influenced explicitly or implicitly by the network status. This becomes a more realistic option when it is considered that a growing proportion of terminal equipment, such as set-top boxes are leased to the end-user by service providers and so remain under their control.



① Intra-Device Power Management

② Inter-Device (AS Domain) Power Management

③ Inter-Device (end-to-end) Power Management

In this scenario, power management operates as an adjunct to SNMP, situated alongside the NMS. It employs *get* and *set request* messages to explicitly alter the functioning of individual equipment within the operator's AS domain. Where appropriate, this will be "observed" by the interior routing protocol leading to reconvergence in response to status changes. This approach could transparently coexist with end-to-end power management measures.

Figure 2: Example Scenario: PMGT Adjunct to SNMP

IV. The "Real World"

Two cautionary factors should be considered when devising schemes for dynamic energy management. Firstly, traffic usage patterns on the Internet are changing. Set-top boxes for media services operate "around the clock" – along with ambient intelligence devices. This means that the extent of the anticipated daily variations in load may not be present to the same extent in future years. However, recent published link utilisation figures suggest that appreciable variations currently persist. The second factor is the behaviour of Operators. Operators are EXTREMELY resistant to introducing mechanisms that may jeopardise customer (perceived) SLAs. Disruption due to reconfiguration must be avoided. Furthermore the cost-savings in response to demand fluctuations must be sufficient to warrant any change. One approach would be to confine changes to transparent ones. These involve per-device modifications that do not impact on their performance within the overall system. Examples include: variable fan speeds, idle modes – backplane fabric adjustments. These avoid the need for systematic migration and are regarded as low-risk as the energy saving mechanism(s) they employ do not impact on their surroundings. However, they suffer from omitting the gains that might be achievable by employing a coordinated energy management strategy. Of course, co-ordination implies signalling, which in turn adds to the complexity of the system and the potential sources of failure. In addition, new functionality must not interfere with existing coordinated activities such as fault diagnosis and resilience.

V. Relevance to COST IC0804

COST IC0804 is tasked with exploring means to reduce energy consumption of distributed systems from data centres, processing clusters to networking and home environments. Although the wired communication infrastructure is a minority energy consuming component with the overall ICT sector, it does represent an appreciable target for improved efficiency. Putting aside the environmental motivation, there remains a strong commercial case for "greening" network devices. Given the limited attention that this topic has received to-date, it is important that this COST Action explore possibilities for topology and service adaptation whilst guaranteeing quality of service. To do so, it is necessary to examine mechanisms that are available at the device-level, across an Autonomous System and even end-to-end. Furthermore, the combination of short and long-term variations in load imply a single mechanism will fall somewhat short of the achievable savings. It is likely that a coordinated management of long-term variations should be used in concert with schemes that can locally react to transient fluctuations. However, whatever approach is taken, it is essential that appropriate service delivery is not jeopardised.

REFERENCES

[1] M. Hodes et al, "Energy and power conversion: A telecommunication hardware vendor's perspective", Power Electronics Industry Group Presentation, Alcatel-Lucent, 2007. http://www.peig.ie/pdfs/ ALCATE~1.PPT (last visited: May 2009).

[2] S. Nedevschi et al, "Reducing network energy consumption via sleeping and rate-adaptation", 5th USENIX Symposium on Networked Systems Design and Implementation, 2008.

[3] Y. Luo et al, "Conserving Network Processor Power Consumption by Exploiting Traffic Variability", ACM Transactions on Architecture and Code Optimization (TACO), 2007.

[4] H Ikebe et al, "Green energy for telecommunications", Telecommunications Energy Conference, INTELEC 2007.

[5] A. Lord, "OPEX savings of all-optical core networks", 35th European Conference on Optical Communication, ECOC 2009.

[6] J. Baliga et al, "Photonic Switching and the Energy Bottleneck," Photonics in Switching, 2007, vol., no., pp.125-126, August 2007.

[7] Y. Wu et al, "Power-Aware Routing and Wavelength Assignment in Optical Networks", 35th European Conference and Exhibition on Optical Communication, ECOC 2009.

[8] B. St. Arnaud, "New Internet routing architectures to reduce carbon emissions", BELnet Conference, Brussels, Belgium, December 2007.

[9] E. Bonetto et al, "Optical Technologies Can Improve the Energy Efficiency of Networks", 35th European Conference and Exhibition on Optical Communication, ECOC 2009.

[10] B. Puype et al, "Energy Efficient Multilayer Traffic Engineering", 35th European Conference and Exhibition on Optical Communication, ECOC 2009.

[11] L. Chiaraviglio et al, "Reducing Power Consumption in Backbone Networks", IEEE International Conference on Communications (ICC'09), Dresden, Germany, 2009.

[12] J. Chabarek et al, "Power Awareness in Network Design and Routing", 27th IEEE Conference on Computer Communications, (INFOCOM) 2008.

[13] P. Francois, O. Bonaventure, "Avoiding Transient Loops During the Convergence of Link-State Routing Protocols IEEE/ACM Transactions on Networking, Vol. 15, No. 6, December 2007.

[14] J. Fu et al, "Loop-Free Updates of Forwarding Tables", IEEE Transactions on Networking and Service Management, Vol. 5, No. 1, March 2008.G. O. Young, "Synthetic structure of industrial plastics (Book style with paper title and editor)," in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.

[15] C. Lange et al, "Energy Consumption of Telecommunication Networks", 35th European Conference and Exhibition on Optical Communication, ECOC 2009.

[16] A.K. Leung, Y.-K. Kwok, "Community-based asynchronous wakeup protocol for wireless peer-to-peer file sharing networks", 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005., vol., no., pp. 342-350, July 2005.

Chris Phillips received a PhD on concurrent discrete event-driven simulation from Queen Mary, University of London in 1991. Until his return to Queen Mary as a Reader in 2000, he worked in industry as a hardware and systems engineer with Bell Northern Research, Siemens Roke Manor Research and Nortel Networks, focusing on broadband network protocols including SDH and G-MPLS, resource management and resilience. A common theme that underpins his research is how management mechanisms can be developed to enable limited resources to be used effectively in a changing environment. This has been addressed in the context of all-optical carrier networks, wireless sensor networks, and nomadic computing.

Lars Dittmann received the M.Sc. EE and Ph.D. from the technical university of Denmark in 1988 and 1994, and is currently Professor at the University within the area of integrated networks. Lars Dittmann has since January 99 been heading the network competence area (covering both optical and electrical networks) within the Research Center COM (now DTU Fotonik) at the Technical University of Denmark and was prior to that responsible for electronic switching and ATM networks at the Center for Broadband Telecommunication. Lars Dittmann has been involved in a number of EU and national project working on optical networks and has leading experience from several national and international research projects.

Martin Collier received the M.Eng by research in 1988 followed in 1993 by a PhD on Switching Techniques for Broadband ISDN, both from Dublin City University. He is currently a Senior Lecturer and Director of the Network Innovations Centre at DCU. Current research includes wireless sensor network routing protocols, routing in packet-switched networks, optical burst switching and the switching theory of multistage networks. He has been Principle Investigator on many projects including an Enterprise Ireland investigation into advanced router architectures using Network Processors.

Profit-Aware Allocation Policies for Power and Performance

Michele Mazzucco
University of Cyprus
Cyprus

***Abstract*— A server farm is considered, where a number of servers are used to offer a service to paying customers. Every completed request generates a certain amount of profit, while running servers consume electricity for power and cooling. A dynamic allocation policy aiming at satisfying the conflicting goals of maximizing the users experience while minimizing the cost for the provider is introduced. The results of some experiments are described, showing that the proposed scheme performs well.**

I. Introduction

A lot of of data centers, purpose-built facilities composed of thousands of servers and providing storage and computing services within and across organizational boundaries, have been built in the last few years. These massive architectures provide advantages for both users and service providers, as their size makes it possible to achieve substantial economies of scale that are simply not possible for enterprise data centers.

Unfortunately, high electricity consumption associated with running a data center not only increases greenhouse gas emissions, but also increases the costs of running the server farm itself. Therefore, it becomes obvious to think about solutions towards the next generation of server farms, *i.e.*, data centers that are energy efficient. Unfortunately, because of the increasing use of the Internet as a provider of services and a major information media have changed significantly, the expectations in terms of performance and responsiveness. Hence, service providers face the problem of reducing operating costs while delivering an acceptable level of performance.

Most researchers are focusing on optimizing the energy efficiency at the hardware level. However, despite considerable interest in designing servers whose power consumption is proportional to their utilization [1], the reality is that the amount of power consumed by an idle server is about 65% of its peak consumption [2]. Thus, the only way to significantly reduce data centers' power consumption is to improve the server farm's utilization, *e.g.*, by tearing down servers in excess whenever that can be justified by demand conditions, as existing hardware components offer only limited controls for trading power for performance [3].

In this paper we propose and evaluate techniques aiming at turning on and off servers on demand in order to ensure stable operation, *e.g.*, meeting performance requirements, while minimizing the number of running servers and thus maximizing the revenues of service providers.

It is assumed that the reader is familiar with the queueing theory terminology. The interested reader is referred to [4] and [5] for a discussion of the basic concepts and terminology.

II. System Model for the Cloud

The provider has a cluster of S identical processors/cores (*servers*, from now on), n running and $(S-n)$ switched off. The provider offers each server for a lease, and a customer who rents a server (*e.g.*, by running a virtual machine on it) is essentially creating a job whose size is unknown to the provider (the size of the job is the length of the lease). Servers are not shared, so each server can handle at maximum one job at any given time (since the power drained by each CPU is a linear function of the load [6], this approach can be applied to a scenario where multiple *virtual* machines are running on a physical CPU). If, once a server has finished processing a request, no other jobs enter the system, the server begins to idle (*i.e.*, it consumes energy without generating any revenue).

The contract that regulates the provisioning contract states, among the other clauses, that for each job *a user pays a charge which is proportional to the job size*, while the cost the provider charges for renting a server is c \$ per unit time. Determining the amount of the charge is outside the scope of this paper, and it could also include the costs related to the use of storage space or network bandwidth. Finally, an arrival finding all n servers busy is blocked and lost, without affecting future arrivals, see Figure 1, while running servers consume energy, which costs r \$ per kWh.

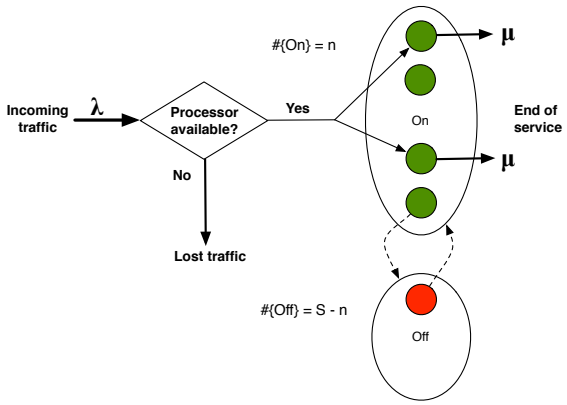


Fig. 1. System model for cloud providers. Jobs enter the system with rate λ and have an average service time of $1/\mu$. Incoming traffic is lost if no server is available.

Within the control of the provider is the 'resource allocation' policy, which decides how many servers to run in order to optimize the provider's profit. Because of the random nature of user demand, static policies would under-perform. Hence the provider should be able to dynamically change the number of running servers in response to changes in user demand. The problem is how to do that in a sensible manner.

During the intervals between consecutive policy invocations, or 'configuration intervals', the number of running servers remains constant. Those intervals are used by the controlling software to collect traffic statistics and obtain current estimates of the average arrival rate (λ) and service time ($1/\mu$) as well as the squared coefficients of variation (the variance divided by the square of the mean) of the above values . These values are used by the allocation policy at the next decision epoch.

While different metrics can be used to measure the performance of a computing system, as far as the service provider is concerned the performance of the server farm is measured by the average revenue, R, earned per unit time. That value can be estimated as

$$R = \frac{c}{\mu}T - rP, \tag{1}$$

where c/μ is the average charge paid by a customer for having his/her job run, T is the system's throughput and P is the amount of energy consumed per unit time by the data center. How to estimate T is described later, while P can be easily computed by assuming a linear relationship between power consumption and CPU utilization [6]

$$P = ne_1 + m(e_2 - e_1), \tag{2}$$

where e_1 is the energy consumed per unit time by idle servers, e_2 is the energy drawn by each busy server, and m is the occupancy of the system ($m \leq n$)

$$m = \left\lceil \frac{T}{\mu} \right\rceil. \tag{3}$$

For the following it will be convenient to indicate explicitly the dependency of Equation (1) on the parameter n by introducing the notation

$$R = r(n), \tag{4}$$

where $r(n)$ stands for the right-hand side of (1).

It is perhaps worth nothing that, although no assumption regarding the relative magnitudes of charges and costs parameters is made, the most interesting case is when they are close to each other. If the charge for executing a job is much higher than the cost paid by the provider to run a server, one could guarantee a positive (but not optimal) revenue by switching on all servers, regardless of the load. On the other hand, if the charge is smaller than the cost, than it would be better to switch all servers off.

In order to effectively control the energy consumption it is necessary to have a quantitative model of user demand and service provision. Suppose than n servers have been allocated to serve user demand. If jobs enter the system according to an independent Poisson process, than the average number of jobs inside the system can be modeled as an Erlang-B system (see [4] for more details) with n trunks and traffic intensity $\rho = \lambda/\mu$, augmented with the economic parameters described above. Therefore, the blocking probability, p_n, *i.e.*, the probability that all servers are busy and thus further requests are lost, can be computed efficiently and quickly [7].

If the arrival process is not Poisson, then the model becomes sensitive to the distribution of job sizes. Hence, the most appropriate model would be the $G/GI/n/n$ queueing model, for which there is no exact solution. However, a reasonable approximation for the blocking probability exists (see Whitt, [8]). This enables us to estimate the blocking probability, and thus the number of jobs entering the system (and completing service) per unit time

$$T = \lambda(1 - p_n), \tag{5}$$

with $(1 - p_n)$ being the probability that an incoming job finds an idle server. Hence, using Equations (1), (2) and (5) it is possible to compute the average revenue, R, efficiently and quickly.

The numerical experiment reported Figure 2 examines how the number of running servers affects R under different loading conditions. The data center consists of 100,000 severs, the settings for the energy consumption under different usage patters are those reported in [6], and the potential offered load ranges between 30% and 90% by varying the arrival rate. The figure illustrates that (i) in each case there is an optimal

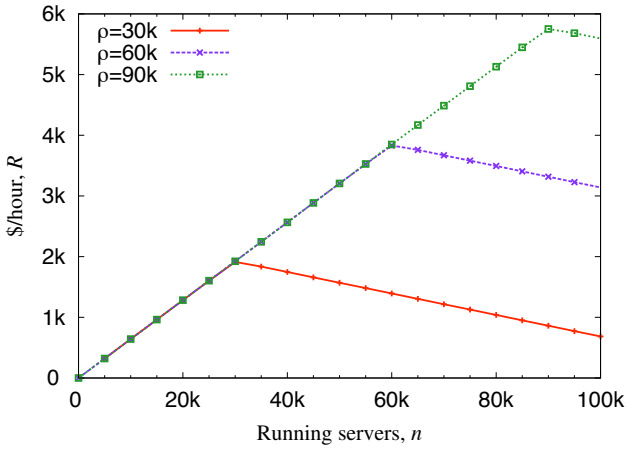


Fig. 2. Revenue as function of the running servers.

number of servers that should be switched on; (ii) the heavier is the load, the higher is the optimal number of servers as well as the maximum achievable revenue; and (iii) when $n > n_{opt}$, the system under-performs because the cost of running idle servers erodes revenues, while when $n < n_{opt}$, the system under-performs because it misses potential revenues.

III. Server Allocation for Internet Services

Next, we consider the case where the server farm is used used to offer a service to impatient customers. Every completed request generates a certain amount of profit. For example, it can be a profit coming from advertisements or from sales (in case of online merchants, such as Amazon).

As before, running servers consume electricity for power and cooling, but now jobs finding all the servers busy are not lost. Instead, they are parked into an external first-in-first-out (FIFO) queue whose size is assumed to be infinite. However, waiting customers might leave the system before receiving service if they experience excessive delays, see Figure 3. If, once a server has finished processing a request, the queue is empty, the server begins to idle. Otherwise, it removes the leading job from the waiting queue and starts processing it.

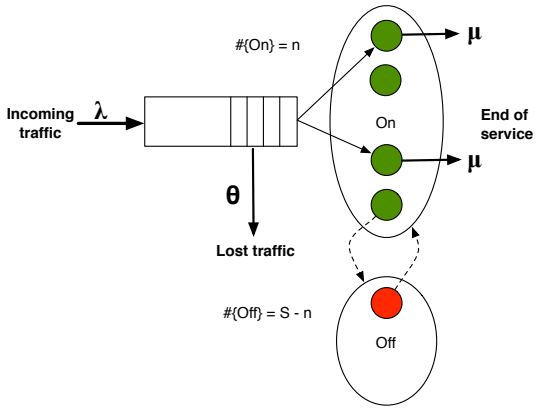


Fig. 3. System model. Jobs enter the system with rate λ, have an average service time of $1/\mu$, and abandon the system with rate θ while waiting.

The average revenue, R, earned by the service provider per unit time can now be estimated as

$$R = cT - rP, \tag{6}$$

where all the variables are the same as those used in Equation (1). However, now the revenue generated by completed jobs does not depend on the job size anymore. In this model, we assume that jobs enter the system according to an independent Poisson process with rate λ, operating servers accept one job at a time, and the service times are exponentially distributed with mean $1/\mu$. One might think that such a system could be easily modeled as an $M/M/n$ queue [4] with n trunks and traffic intensity $\rho = \lambda/\mu$. Unfortunately, the Erlang-C model does not acknowledge customers abandonment; hence, it either distorts or completely fails to provide important information. The model discussed in this section allows customers abandonment by assuming that a time-out policy is in operation: if a job entering the system does not acquire a server before its time-out period expires, the job is terminated and leaves the system without generating any revenue. This can be used to model HTTP time-outs as well as impatient customers. The latter is of particular importance, as [9] reports that 75% of people would not go back to a web site that took more than 4 seconds to load.

In order to estimate the throughtput, which is needed to find R, it is necessary to compute the steady state analysis of the Markov chain associated to the model sketched in Figure 3. A tractable way to model this system is the $M/M/n+M$ queue, also known as Erlang-A (where the 'A' stands for 'Abandonment') [10]. Having computed the stationary distribution of jobs present

$$p_j = \begin{cases} \dfrac{\rho^j}{j!} p_0 & \text{if } j \leq n \\ \dfrac{\rho^n}{n!} p_0 \left[\displaystyle\prod_{k=n+1}^{j} \dfrac{\lambda}{n\mu + \theta(k-n)} \right] & \text{if } j > n \end{cases}. \tag{7}$$

with p_0 being the probability that all powered up servers are idle, it is possible to compute the delay probability, $P(W > 0)$, and the conditional abandonment probability, $P(Ab|W > 0)$. Hence, by means of the Bayes formula the probability of abandonment, $P(Ab)$, can be expressed as

$$P(Ab) = P(W > 0)P(Ab|W > 0). \tag{8}$$

Having computed the abandonment probability for a given set of parameters (*i.e.*, load and number of available servers) the average number of requests served per unit time can be expressed as

$$T = min(n\mu, \lambda[1 - P(Ab)]). \tag{9}$$

It is perhaps worth noting that as the size of the server farms grows, the system achieves economies of scale that make it more robust against traffic variability. Hence, while violating the Markovian assumptions about the arrival, patience and service processes affects the average queue length, it does not substantially change the abandonment rate [11].

IV. Allocation Policy

Consider now a decision epoch. At that time the state of the system is defined by the number of servers which are not powered down and by the potential offered load. If the allocation does not change, the expected revenue for the next configuration interval is simply $r(n)$, see Equation (4). If the number of running servers change, things are more challenging, as tearing them up/down is not an instantaneous process. Instead, it takes on average k units time, during which servers consume energy without generating any revenue. To make things more complicated, system's reliability is affected by state changes, as hardware components tend to degrade faster with frequent power on/off cycles than with continuous operation. Therefore, each state change involves the following cost

$$Q = \frac{|\Delta n|}{t} (\sum_{i=1}^{l} d_i + kre_{max}), \tag{10}$$

where t is the length of the observation windows, $|\Delta n|$ is the number of servers that are switched on/off, e_{max} is the power consumed per unit time during state changes, k is the average time required to switch a server on/off, d_i is the cost for a hardware component's state change, and l is the number of hardware components.

The expected *change* in revenue resulting from a decision to change the number of running servers can be expressed as

$$\Delta r(n', n) = r(n') - r(n) - Q, \tag{11}$$

where $r(n)$ stands for the right-hand side of (6).

When R is computed for different values of n, it becomes clear that the revenue is a unimodal function of n, *i.e.*, it has a single maximum. That observation implies that one can search for the optimal number of servers to run by evaluating R for consecutive values of n, stopping either when R starts decreasing or, if that does not happen, when the increase becomes smaller than some value ϵ. This can be justified arguing that R is a concave function with respect to n. Intuitively, the economic benefits of powering more servers on become less and less significant as n increases, while the loss of potential revenues gets bigger and bigger as n decreases. Such a behavior is an indication of concavity. Hence, a fast algorithm of the binary search variety can be applied to find the 'best' n in $O(log_2(n))$ iterations. Both adaptive (*i.e.*, a policy which assumes that the load for the next configuration interval will be the same as that of the last one) and predictive algorithms (*e.g*, double exponential smoothing) can be used in conjunction to this algorithm. For example, in Figure 4 we report the energy savings produced by the 'Adaptive' policy compared to two versions of the 'Static' policy, a policy which runs always the same amount of servers. The size of the server farm is now set to 250 servers, configured as reported in [6], while the load ranges between 10% and 110% (*i.e.*, the system would be over-saturated without job abandonment) by varying the arrival rate. The figure clearly shows that the Adaptive heuristic runs servers only when necessary, thus reducing its carbon footprint as well as the provider electricity bill.

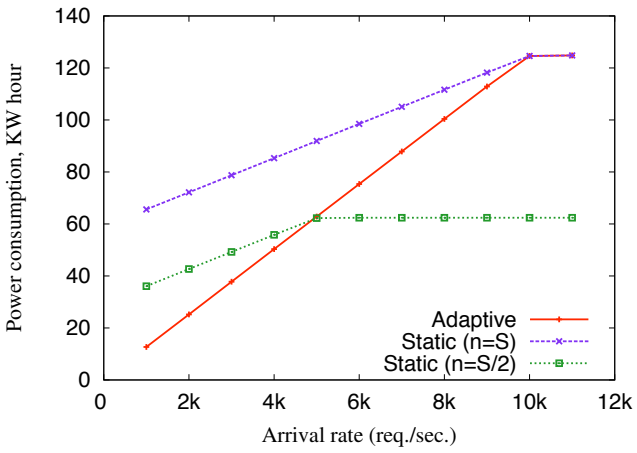


Fig. 4. Observed energy consumption for different policies.

V. Conclusions

This paper discussed an easily implementable policy for dynamically adaptable Internet services and Cloud provision. Decisions such as how many servers to run can have a significant effect on the revenue earned by the provider. Moreover, those decisions are affected by the contractual obligations between clients and provider.

The experiments we have carried out have shown that the proposed polices work well under different traffic conditions (because of space constraints, just a few of them are described into this paper), and that the Adaptive heuristic would be a good candidate for practical implementation. The policy we have described in Section IV adapts to the changes in user demand by assuming that the traffic during the next window will be the same as that of the previous window. If that is not the case, simple heuristics that try to predict what the future load will be can be easily embedded into the proposed framework [6].

Possible directions for future research include taking into account the trade offs between the number of running servers, the frequency of the CPUs and the maximum achievable performance, as well as fault tolerance issues.

Acknowledgements

The author would like to thank Dmytro Dyachuk, the SEARCHiN project (Marie Curie Action, contract number FP6-042467), and the EU Cost Action IC0804 (Energy Efficiency in Large Scale Distributed Systems).

References

[1] L. A. Barroso and U. Hölzle, "The case for energy-proportional computing," *Computer*, vol. 40, no. 12, pp. 33–37, 2007.

[2] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The Cost of a Cloud: Research Problems in Data Center Networks," *SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 68–73, January 2009.

[3] S. Harizopoulos, M. Shah, and P. Ranganathan, "Energy Efficiency: The New Holy Grail of Data Management Systems Research," in *4th Biennial Conference on Innovative Data Systems Research (CIDR)*, 2009.

[4] I. Mitrani, *Probabilistic Modelling*. Cambridge University Press, 1998.

[5] S. M. Ross, *Introduction to Probability Models*. Academic Press, 2000.

[6] M. Mazzucco, D. Dyachuk, and R. Deters, "Maximizing cloud providers revenues via energy aware allocation policies," in *Proceedings of the 3rd IEEE Internation Conference on Cloud Computing (IEEE Cloud 2010)*, July 2010.

[7] O. Hudousek, "Evaluation of the Erlang-B formula," in *Proceedings of RTT 2003*, 2003.

[8] W. Whitt, "Heavy Traffic Approximations for Service Systems with Blocking," *AT&T Bell Laboratories Technical Journal*, vol. 63, May-June 1984.

[9] J. McGovern, "Selfish, Mean, Impatient Customers," July 2008. [Online]. Available: http://www.cmswire.com/cms/web-content/selfish-mean-impatient-customers-002891.php

[10] C. Palm, "Research on Telephone Traffic Carried by Full Availability Groups," *Tele (English Edition)*, vol. 1, 1957.

[11] W. Whitt, "Efficiency-Driven Heavy-Traffic Approximations for Many-Server Queues with Abandonments," *Management Science*, October 2004.

Energy-Aware Resource Allocation

Damien Borgetto, Georges Da Costa, Jean-Marc Pierson, Amal Sayah

IRIT, Toulouse University

{borgetto,dacosta,pierson,sayah}@irit.fr

***Abstract*—This paper deals with the reduction of energy consumption in large scale systems, especially by taking into account the impact of energy consumption for server consolidation. Decreasing the number of physical hosts used while ensuring a certain level of quality of services is the goal of our approach. We introduce a metric called energetic yield which represents the quality of a task placement on a subset of machines, while taking into account quality of service and energy efficiency aspects. It measures the difference between resources required by a job and what the system allocates ultimately, while trying to save energy. Our work aims at minimizing this difference. We propose placement heuristics that are compared to the optimal solution and to a related system. In this paper, we present a set of experiments showing the relevance of this metric in order to reduce significantly energy consumption.**

I. Introduction and Motivation

For several years there as been an increase of power consumption for computational servers and data centers. Recent studies in the U.S [8], and in Europe [2] showed that power consumption becomes a matter of concern. Operators of such centers are investing substantial amounts in order to supply their infrastructure or to disperse emitted heat .

There are several solutions to reduce the energy bill in both financial and environmental aspects. The first one is based on the energy efficiency of the infrastructure itself, promoting air circulation or alternative cooling of machine rooms. The second one is to deploy less energy-hungry hardware, where the flops/watt ratio is better [6]. Major companies continuously improve their products in this way.

The third solution is algorithmic and tries to optimize processes in order to reduce their energy fingerprint. This solution aggregates numerous approaches, such as network protocols [11], modelling of the consumption [7], task scheduling predicting idle time [5]. Our work aims at providing an energy-efficient job placement, while guaranteeing a certain quality of service, that can lead to turning off hosts.

This approach has been studied before by Stillwell et al. [12] in a different context. While the approach used by Stillwell et al. relies on server consolidation to optimize the utilization of a consistent subset of machines (see section II for details), we go further by integrating in the approach an electric consumption consideration so that the global energy impact is reduced. The idea is here to create a compromise between cost in terms of energy consumption and quality of service. On the one hand the use of only one machine, chosen to have the lowest energy consumption, optimizes the energy consumption but does not guarantee the task's quality of service. On the other hand using the maximum amount of machines fits better the needs of the jobs, but consumes a large amount of energy. Wasting energy can be avoided by grouping jobs on the same host. Although, as shown in this paper, the naïve approach that allows one to turn off unused machines, even after sorting these by energy consumption, can be improved by defining more precisely a metric allowing one to place tasks tending to a compromise.

II. Server Consolidation

The works of Stillwell et al. in [12] consists of a heuristic addressing the job placement problem in a multi-constrained context of memory and computational capability. The assumption is made that hardware is fully homogeneous. The works of Stillwell et al. introduces a user-centric metric, the *yield*, defining the quality of a placement. In order to make a good placement, one will want to maximize this metric.

$Y_{ij} = \sum_{j=1}^{H} (\frac{\alpha_{ij}}{\alpha_i})$

α_i : fraction of computational capability used by the job *i* if alone on a physical host.

α_{ij} : fraction of computational capability of host *j* currently allocated to the job *i*.

H : Number of hosts

The Y_{ij} notation is used in order to be consistent with the formula in section III, and to show on which host the job is actually allocated.

The yield can be thought as a measure of the job's satisfaction rate, meaning the yield of a job represents the fraction of the maximum achievable computing rate that is achieved. For example, a job that requires 60% of the computational capability of an host, but is allocated at 30% of it, will achieve a yield of $\frac{30}{60} = \frac{1}{2}$. The works of Stillwell et al. are about resource allocation using *multi capacity bin-packing* (MCB) [9] over memory and CPU load. Stillwell et al. uses this metric to aggregate services over a small (hopefully minimal) subset of machines, while guaranteeing a good quality of service by maximizing the minimum yield.

The techinque used in [12] allows to group jobs over a small number of physical machines, thus turning off those that aren't used. However, considering energy consumption is only a by-product: when by chance a host is left unused, it can be turned

off in order to save energy.

The performance of this heuristic depends heavily on the functions used to sort the job lists. In [12], the most efficient sorting function is MCB8 which consists of sorting in descending order the maximum of the memory and CPU requirements (i.e.: *max(memory,CPU)* in descending order).

III. Taking Energy Into Account

The approach presented by Stillwell et al. focuses on a metric (the *yield*) measuring the quality of a placement. The more a job is allocated close to its requirements, the larger the yield. The assumption is made that there is a full homogeneity of the physical hosts. We chose to keep this assumption because in a cluster context, we often find machines with similar specifications in a same rack. Furthermore, studies [10] showed that physical location of the machines induces differences in their energy consumption.

In order to address the problem of energy efficiency without having too much of a complex model, we make the following simplifying assumptions:

(H1) Machines are heterogeneous in terms of energy, but have the same computational capability.

(H2) A machine constantly consumes C^{min} watts while empty and C^{max} watts fully loaded.

(H3) The extra energy consumed by the execution of a job on a host is a (linear) function f of its computational capability and the host's specifications. It means that $\forall i \in [1..J], \forall j \in [1..H] : \delta C_{ij} = \alpha_{ij} \times f(j)$ where δC_{ij} is the induced energy by the job i on the host j. For now, f is assumed to be $f(j) = (C^{max}_j - C^{min}_j)$.

(H4) As in [12], jobs are assumed to be infinite and being able to migrate.

A. Energetic yield

It is necessary to change the metric in order to take into account the energy efficiency with those new assumptions. We use the following formula to define the energetic yield.

For a job $i \in [1..J]$, allocated on a host $j \in [1..H]$, we have :

$$YE_{ij} = \frac{\left[\sum_{j=1}^{H}(\frac{\alpha_{ij}}{\alpha_i})\right]^{1-k}}{\left[\lambda\delta C_{ij}+(1-\lambda)(A_j(1-\sum_{i'=1,i'\neq i}^{J}(\alpha_{i'j})))\right]^{k}} = \frac{(Y_{ij})^{1-k}}{(E_{ij})^{k}}$$

With $0 \leq \lambda \leq 1$ and $0 \leq k \leq 1$.

This equation is used to conciliate 3 different goals that are:

- aggregating jobs on a reduced number of hosts in order to shut down unused ones;
- placing jobs on energy efficient hosts;
- taking into consideration the quality of service of the jobs.

The Y_{ij} part is the yield and evaluates the quality of the computational resources allocation to the jobs. The E_{ij} part takes into account the energy efficiency problematic. k allows us to make a tradeoff between computational performances and energy savings. The energy part can be separated in two parts : $E_{ij} = \lambda\delta C_{ij} + (1-\lambda) \times A_j(1 - \sum_{i'=1,i'\neq i}^{J}(\alpha_{i'j}))$

- Term δC_{ij} represents the job contribution to the energy consumption of the host j.
- Term $A_j(1 - \sum_{i'=1,i'\neq i}^{J}(\alpha_{i'j}))$ allows to group jobs on attractive hosts. A_j is the attractiveness of the host j and the smaller the value, the more attractive the machine is. In the following $A_j = C^{min}_j$. This part goes smaller (thus leading to a big energetic yield) as the host is both attractive and loaded.

The term A_j is used to weight hosts in order to be inclined to choose energy-efficient machines. λ allows us to weight between the 2 terms above, so that we can make more important one or the other. So if it is not possible to turn off machines, we will set λ to 1. Our approach allows to define the quality of the placement of a job, without caring of placements to come.

B. Properties

Our metric is bound by the three following properties.

Property 1: The metric's value will be larger with a host that consumes less than another when other host's specifications are the same. It means that if we have 2 machines with the same proposed yield and the same computational load, when a job arrives, the metric will be better for the host in which the increase of energy consumption is the lowest.

$$\begin{aligned}&\forall i \in [1,J]; \forall j,h \in [1,H]; j \neq h; k \neq 0; \lambda \neq 0 :\\ &(\textstyle\sum_{i'=1,i'\neq i}^{J}(\alpha_{i'j}) = \sum_{i'=1,i'\neq i}^{J}(\alpha_{i'h})\\ &\wedge \textstyle\sum_{j=1}^{H}(\frac{\alpha_{ij}}{\alpha_i}) = \sum_{h=1}^{H}(\frac{\alpha_{ih}}{\alpha_i})\\ &\wedge A_j = A_h\\ &\wedge \delta C_{ij} < \delta C_{ih})\\ &\Rightarrow YE_{ij} > YE_{ih}\end{aligned}$$

Property 2: With the same specifications, the metric will be inclined to have a larger value with an allocation of a job on a host that already executes jobs (thus that can't be turned off) than on an empty machine. Thereby, if we have 2 hosts and 2 jobs, we will prefer putting the 2 jobs on only one host rather than allocating one job to each host.

$$\begin{aligned}&\forall i \in [1,J]; \forall j,h \in [1,H]; j \neq h; k \neq 0; \lambda \neq 1 :\\ &(\textstyle\sum_{i'=1,i'\neq i}^{J}(\alpha_{i'j}) > \sum_{i'=1,i'\neq i}^{J}(\alpha_{i'h})\\ &\wedge \textstyle\sum_{j=1}^{H}(\frac{\alpha_{ij}}{\alpha_i}) = \sum_{h=1}^{H}(\frac{\alpha_{ih}}{\alpha_i})\\ &\wedge A_j = A_h\\ &\wedge \delta C_{ij} = \delta C_{ih})\\ &\Rightarrow YE_{ij} > YE_{ih}\end{aligned}$$

Property 3: It is possible to set the system sensitivity to energy. Here, the parameter k varies between 0 and 1, such as if $k = 0$ only the yield will be taken into account and if $k = 1$ only the energy will be taken into account. Thereby, when k increases, the properties above are step by step extended to accept a bigger loss of yield.

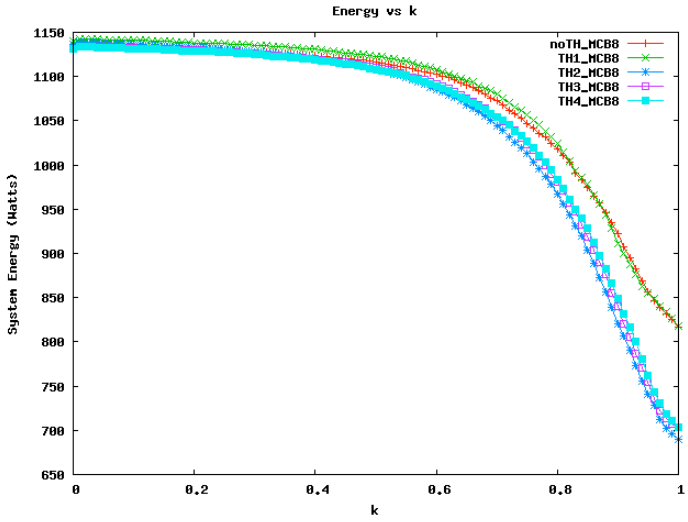


Figure 1. Different sorting functions for the hosts, small problems

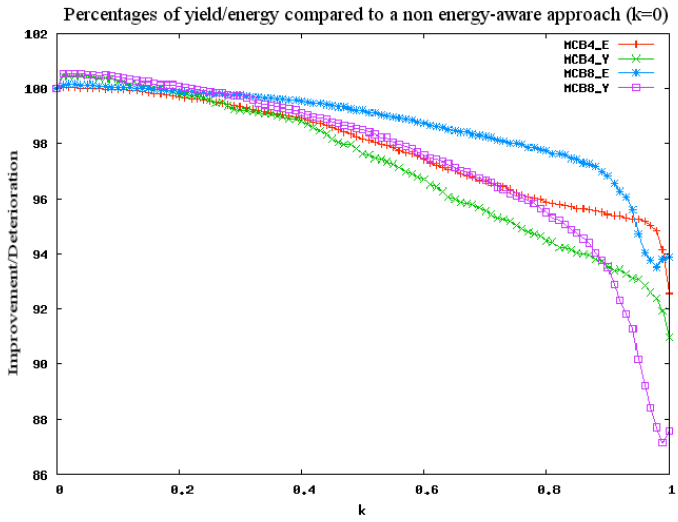


Figure 2. Comparison between energy and yield losses by varying k, small problems

C. Optimizations

The algorithm proposed in [12] relies on an algorithm of bin-packing using the yield to properly allocate jobs on hosts. The paper compares several techniques to sort jobs and thus evaluate the impact of the sorting function on the quality of the resulting placement.

One of our contributions is to take into account the energy heterogeneity of the machines. Indeed, the machines sort that the bin-packing takes into account has an impact over the algorithm's performance. In [3] we compared the following sorting functions for the hosts :

- TH1 : C^{min}, ascending order
- TH2 : C^{max}, ascending order
- TH3 : $C^{min} + C^{max}$, ascending order
- TH4 : $C^{max} - C^{min}$, ascending order

IV. Experiments

In order to evaluate our approach, we ran various simulations, watching the evolution of the yield deterioration with the reduction of the system energy consumption, while increasing the parameter k. The system energy is the sum of all energies consumed by each host with at least one job running on it.

A. Methodology

We have generated a set of problems based on the problem generation in [12], a problem being a set of hosts and a set of jobs to be allocated, using the following methodology. We added a medium sized problem generation, multiplying the number of hosts and jobs by 6, thus having 24 hosts with 36, 48, 60 and 72 jobs.

In order to highlight the influence of our metric on the system energy consumption, we search for a near optimal solution with the different MCB algorithms described in [12], while, for each generated problems, varying k (the weight of the energy part of our metric) from 0 to 1 by steps of 0.01. Moreover, these tests were made with values of λ varying from 0 to 1 by steps of 0.1.

Finally, in order to define the energy consumption of the hosts, we also assigned to each host a C^{min} value between 100 and 200 watts, and a value of C^{max} between 200 and 400 watts, both generated using a uniform distribution. For further details about the existing, it will be necessary to generate those values using a more realistic model.

Each of those 1440 problems will be the input of a simulator that we specifically developed for those experiments. This simulator evaluates accurately the metric using the problem in input.

B. Results

In order to analyze the results of our simulations, we focused on the energy consumption of the system and on the average yield. More detailed experiment results can be found in [3]. Indeed, the system energy will directly be used to evaluate the impact of our method over the energy reduction, whereas analyzing the average yield will allow us to see how much the performances are affected.

As in [12], once the bin-packing is done, we can switch off machines that have no jobs to execute, once the jobs are grouped on other machines. We can go further in energy reduction, going beyond the naïve approach in order to globally take into account the energy consumption. Thus saving more energy

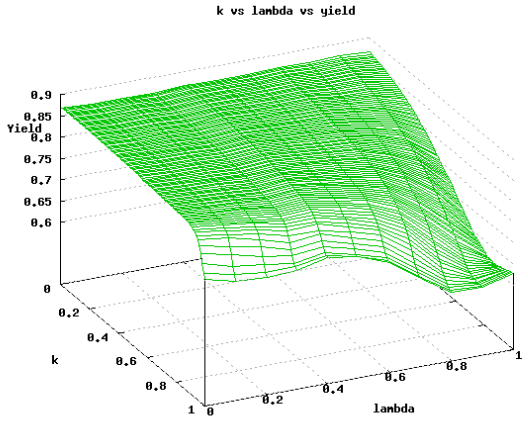


Figure 3. Evolution of the average yield while varying k and λ, small problems

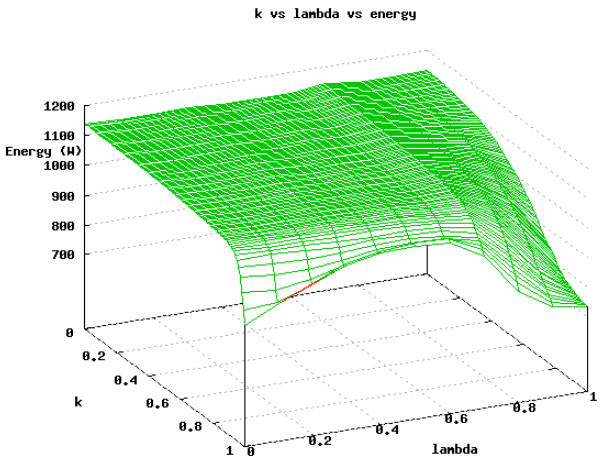


Figure 4. Evolution of the energy while varying k and λ, small problems

by placing jobs on hosts regarding their specifications, while using those specifications to make a good placement. In fact, in [12], a homogeneous context is considered. Obviously, it isn't the case in a realistic case, because even if the machines are identical, they won't necessarily consume the same power based on where they are physically placed [10]. Figure 1 shows the significance of taking into account the heterogeneity of the host's configuration. We can see that with some simple host sorting functions, on a small problem such as 4 hosts, we are already able to save a significant of energy. Without host sorting, we achieve a lower performance because the host to fill in first is the first on the host list. Thereby, we can imagine that on a number of hosts way above the number of jobs, which is a common use case in grids and clusters when not in burst periods, we can easily achieve a substantial increase of the energy efficiency of the system. Figure 2 shows the energy loss percentage and the average yield loss percentage, while varying the parameter k. We can see, for example, that with the MCB8 algorithm and with $k = 0.6$, a loss of yield of 2.5% enables about 1.5% of energy saving. We can imagine for example in the case of jobs that are not too much time constrained, this yield loss is acceptable. Moreover, a loss of 2.5% in average yield isn't necessarily lead to an increase of 2.5% of the computation time, or an increase of 2.5% in energy consumption, because the major part of the energy consumption is due to C^{min}. Of course, this is only for the small problems, thus only for 4 hosts, which means that the leeway is relatively restricted. We have to add that the percentages were generated using the respective values of the algorithms. In other words for example, losing 2% of average yield for the MCB8 is larger in value than 2% for the MCB4 algorithm, because the average yield of the MCB8 is higher than the average yield of the MCB4. For a value of $k = 0.99$, which means where the difference between loss of average yield and energy efficiency profit is maximal, there is about 6% of energy gain, for a loss of yield of less than 13%. This case corresponds to the situation where the yield efficiency is the worst, but it also corresponds to the case where the energy consumption is the lowest.

Figures 4 and 3 show the evolution of the average yield and energy when varying both parameters k and λ. As expected, the energy is the lowest around values $k = 1$ and $\lambda = 1$, and the average yield is also the lowest around those values. Those values are corresponding to the case where we take into account in the energy component only the placement of jobs on good hosts. Besides, it should be noted that the performances achieved with $\lambda = 0$ probably aren't what they seems to be, because there is too little room to regroup jobs on hosts. For such values of the parameters, we are saving around 40% of energy compared to the case where we do not take into account energy ($k = 0$). We also see deterioration of the average yield of around 45% . One should note though that simulations ran to achieve those results were made with the sorting function TH3 (see section III-C) for hosts, and sorting function MCB8 for jobs.

V. Conclusion and Future Works

In this paper, we presented a theoretical study of jobs placement over a set of homogeneous machines, while taking into account parameters *reduction of the energy consumption* and job satisfaction in terms of obtaining wanted resources on chosen sites. The impact of gathering jobs on a reduced number of machines were also handled, which allows to consider if machines can be switched off or not. The first simulations that

we ran in a *small problems* context have :

- validated the interest considering the energy consumption parameter as such and not only as a side-effect induced by a placement tending to load as much as possible a smaller number of machines.
- showed the impact of the energy saving parameter over job satisfaction: the more the energy consumption is considered, the more the yield deteriorates. Thereby an extreme placement achieves to save 40% of energy, while loosing 45% of the average yield.
- showed the importance of the sorting functions, therefore the choice of the hosting machine the most energetically beneficial in order to increase energy savings.
- confirmed that the context of *small problems* allows too few room to achieve both a significant energy saving and a low average yield deterioration.

We hope that from the different cases studied, we will be able to characterize placement strategies suitable to configurations in order to obtain the best energy saving achievable. The context of this study (homogeneous machines, infinite jobs that we manage only the initial placement, ...) has to be extended in order to be closer to real situations, and to consider aspects such as :

- the heterogeneity of the considered machines.
- the dynamic creation/extinction of jobs and the dynamic review of the jobs to the hosts, with the corollary migration of activities[4].
- the consideration of more constrained parameters of quality of service than obtaining required resources : response time, deadlines, planning, starvation, ...
- the overhead generated by the energy management.

At the same time, we are seeking to verify theoretical results obtained with these simulations by the implementation of an autonomous system managing the activity placement (virtual machines [1]) on a given hardware architecture and aiming to reduce energy consumption while using the triptych *Sensors* to observe (resources, activities, ...) - *Decision* (placement, switching on/off hardware, ...) - *Actuators* to implement the decisions.

REFERENCES

[1] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pages 164–177, New York, NY, USA, 2003. ACM.

[2] P. Bertoldi and B. Atanasiu. Electricity consumption and efficiency trends in the enlarged european union. Technical Report EUR 22753EN, Institute for Environment and Sustainability, 2007.

[3] Damien Borgetto, Georges Da Costa, Jean-Marc Pierson, and Amal Sayah. Energy-Aware Resource Allocation (regular paper). In *Energy Efficient Grids, Clouds and Clusters Workshop (co-located with Grid 2009) (E2GC2), Banff, 13/10/2009-15/10/2009*, page (electronic medium), http://www.ieee.org/, octobre 2009. IEEE.

[4] Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansen, Eric Jul, Christian Limpach, Ian Pratt, and Andrew Warfield. Live migration of virtual machines. In *NSDI'05: Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*, pages 273–286, Berkeley, CA, USA, 2005. USENIX Association.

[5] Georges Da-Costa, Jean-Patrick Gelas, Yiannis Georgiou, Laurent Lefèvre, Anne-Cécile Orgerie, Jean-Marc Pierson, Olivier Richard, and Kamal Sharma. The green-net framework: Energy efficiency in large scale distributed systems. In *HPPAC 2009 : High Performance Power Aware Computing Workshop in conjunction with IPDPS 2009*, Rome, Italy, may 2009.

[6] W. Feng, M. Warren, and E. Weigle. Honey, i shrunk the beowulf! In *International Conference on Parallel Processing, 2002. Proceedings.*, pages 141–148, 2002.

[7] Helmut Hlavacs, Georges Da Costa, and Jean-Marc Pierson. Energy consumption of residential and professional switches. Rapport de recherche IRIT//RR-2009-7-FR, IRIT, Université Paul Sabatier, Toulouse, mars 2009.

[8] J. G. Koomey. Estimating total power consumption by servers in the u.s. and the world. Technical report, Stanford University, 2007.

[9] William Leinberger, George Karypis, and Vipin Kumar. Multi-capacity bin packing algorithms with applications to job scheduling under multiple constraints. In *ICPP '99: Proceedings of the 1999 International Conference on Parallel Processing*, page 404, Washington, DC, USA, 1999. IEEE Computer Society.

[10] A.-C. Orgerie, L. Lefevre, and J.-P. Gelas. Chasing gaps between bursts: Towards energy efficient large scale experimental grids. In *Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies, 2008. PDCAT 2008.*, pages 381–389, Dec. 2008.

[11] Barry Rountree, David K. Lowenthal, Shelby Funk, Vincent W. Freeh, Bronis R. de Supinski, and Martin Schulz. Bounding energy consumption in large-scale mpi programs. In Becky Verastegui, editor, *Proceedings of the ACM/IEEE Conference on High Performance Networking and Computing, SC 2007, November 10-16, 2007, Reno, Nevada, USA*, page 49. ACM Press, 2007.

[12] M. Stillwell, D. Schanzenbach, F. Vivien, and H. Casanova. Resource allocation using virtual clusters. In *Proceedings of the 9th IEEESymposium on Cluster Computing and the Grid (CCGrid'09)*, may 2009.

BSC contributions in Energy-aware Resource Management for Large Scale Distributed Systems

Jordi Torres, Eduard Ayguadé, David Carrera, Jordi Guitart, Vicenç Beltran, Yolanda Becerra, Rosa M. Badia, Jesús Labarta and Mateo Valero

Barcelona Supercomputing Center (BSC)
Computer Architecture Department - Technical University of Catalonia (UPC)

{jordi.torres, eduard.ayguade, david.carrera, jordi.guitart, vbeltran, yolanda.becerra, rosa.m.badia, jesus.labarta, mateo.valero}@bsc.es

ABSTRACT

This paper introduces the work being carried out at Barcelona Supercomputing Center in the area of Green Computing. We have been working in resource management for a long time and recently we included the energy parameter in the decision process, considering that for a more sustainable science, the paradigm will shift from "time to solution" to "kWh to the solution". We will present our proposals organized in four points that follow the cloud computing stack. For each point we will enumerate the latest achievements that will be published during 2010 that are the basics for our future research. To conclude the paper we will review our ongoing and future research work and an overview of the projects where BSC is participating.

1. INTRODUCTION

Due to the escalating price of power, energy-related costs have become a major economic factor for ICT infrastructure and its host data centres. In addition to improving energy efficiency, companies are facing increasing pressure to reduce their carbon footprint due to EU regulations and campaigns demanding greener businesses. Our research community is therefore being challenged to rethink ICT strategies, adding energy efficiency to a list of critical operating parameters that already includes service performance or reliability.

In this paper we will present a brief overview of the current work that BSC is doing in the Green Computing field. Until now, our research was centered in performance management of distributed and parallel system [1]. Recently we included the energy parameter in the decision process, considering that for a more sustainable science, the paradigm will shift from "time to solution" to "kWh to the solution".

In order to be clearer, we decided to present the overview of our work, organizing the content of this paper following the well-known cloud computing stack based on three layers: Infrastructure-as-a-Service (IaaS), Platforms-as-a-Service (PaaS) and Software-as-a-Service (SaaS).

We are considering that the current workloads that we should deal with are heterogeneous, including different types of jobs, not only CPU-intensive jobs, but also streaming, transactional, data-intensive, etc.

Regarding the resources, the current hardware that we should consider includes heterogeneous clusters of hybrid hardware (with different types of chips, accelerators, GPUs, …).

The research goals that will direct BSC research proposals, in addition to the goals that we have previously dealt with, like performance, includes different aspects: fulfilling the Service Level Agreements (SLA), considering the energy consumption or taking into account the new wave of popular programming models like MapReduce, amoung others.

However, these cloud goals have made the resource management a burning issue in today

systems. For BSC, Self-management is considered the solution to this complexity and a way to increase the adaptability of the execution environments to the dynamic behavior of Cloud Computing.

This is the BSC approach lead by one of our departments, the "Autonomic Systems and eBusiness Platforms" that is trying to build a "Smart Cloud" that can address the present challenges of the Cloud. The aim of this department is to research on autonomic resource allocation and heterogeneous workload management with performance and energy-efficiency goals for Internet-scale virtualized data centers comprising heterogeneous clusters of hybrid hardware.

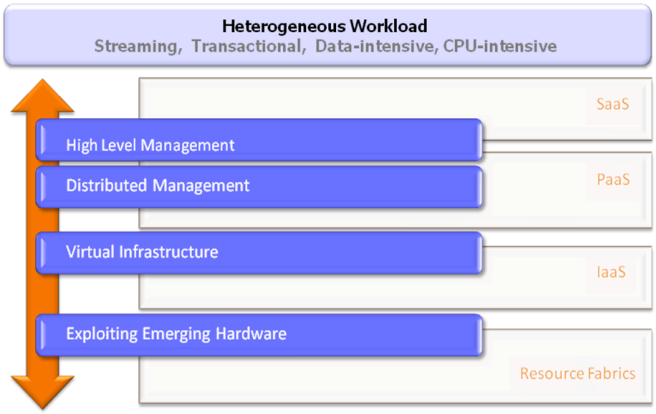


Fig 1. Cloud computing stack organization and summary of points.

As we mentioned previously, we will present our proposals under the cloud computing stack organization, where we can consider 4 main points. One giving solutions at the IaaS level, proposing a virtualized infrastructure. Another, offering a new proposal at the PaaS level. We also consider how the PaaS level functions can provide better support to the SaaS layer. Finally, we consider how emerging hardware can be exploited in an efficient way to reduce the whole energy consumption.

We are considering a whole control cycle with an holistic approach. By that, we mean that each level cooperates with the other levels through a vertical dialogue. Figure 1 shows a diagram that summarizes our proposals.

The rest of the paper revisits the different research points being conducted at the BSC, which emphasizes the main lines of work and the results obtained so far in each focus that will be published during 2010. We conclude the paper with the review of the ongoing work and we briefly list the projects in which the BSC is participating and are related to in this area.

2. VIRTUAL INFRASTRUCTURE

BSC is contributing the research community with the EMOTIVE framework, which allows to simplify the development of new middleware services for the Cloud. EMOTIVE framework is an open-source software infrastructure for implementing Cloud computing solutions that provides elastic and fully customized virtual environments in which to execute Cloud services. EMOTIVE abstracts a Cloud architecture using different layers and provides users with basic primitives for supporting the execution of services (features for resource allocation and monitoring, data management, live migration, and checkpointing, etc.). The core layer wraps each virtualized node and monitors its state, granting full control to the application of its execution environment without any risks to the underlying system or the other applications. One of the main distinguishing features of EMOTIVE framework is their functionalities that ease the development of new resource management proposals, thus contributing to the innovation in this research area. At the moment there are some scheduler implementations that take into account power-aware parameters. The EMOTIVE framework (www.EMOTIVEcloud.net) is build in collaboration with the Grid Computing and Clusters research group at BSC.

The latest achievements in this area are summarized in 2 contributions [2,3] that will be published during 2010.

3. DISTRIBUTED MANAGEMENT

At PaaS level we are working on application placement to decide where applications run and the allocated resources required. For this objective, applications must be modeled to make proper placement decisions in order to obtain a solution not only considering performance parameters, but also energy constrains. We are specially paying attention to MapReduce workloads (currently the most prominent emerging model for cloud scenario), working on the runtimes that allows control and dynamically adjusts the execution of these types of applications with energy awareness. Finally the energy awareness will be addressed at two levels: compute infrastructure (data placement and resource allocation) and network infrastructures (improving data locality and placement to reduce network utilization.

The latest research results in this point are summarized in 2 contributions [4,5] that will be published during 2010.

4. HIGH LEVEL MANAGEMENT

As we previously introduced, we are considering extending the Platform-as-a-Service layer functions to provide better support to Software-as-a-Service layer, according to high level parameters for resource allocation process. The main goal is to propose a new resource management aimed to fulfil the Business Level Objectives (BLO) of both the provider and its customers in a large-scale distributed system. We have preliminary results in the way that the decision-making is done in relation to several factors in a synergistic way depending on provider's interests, including business-level parameters such as risk, trust, and energy.

The most recent developments in this area are summarized in 2 contributions [6,7] that will be published during 2010.

5. EXPLOITING EMERGING HARDWARE

The fourth point deals with the study and development of both new hardware architectures that deliver best performance/energy ratios, and new approaches to exploit such architectures. Both lines of research are complementary and will aim to improve the efficiency of hardware platforms at a low level.

Preliminary results demonstrate that the energy modeling in real time (based on processor characterization) will be leveraged to make decisions.

The latest achievements in this point are summarized in 2 contributions [8,9] that will be published during 2010.

6. ON GOING WORK AND PROJECTS

Although we are considering a whole control cycle with a holistic approach, we will summarize our ongoing work by points in order to be more clear.

The major goal of the first focus was to be a testbed platform for our research, which is almost accomplished. We are currently working in extending the framework with plugins for third-party providers and the federation support for simultaneous access to several clouds that can take into consideration energy-aware parameters.

Our ongoing work in the second focus is devoted to Energy saving in MapReduce workloads. The key element to achieve energy efficiency at this level is the cooperation with the underlying platform and the dynamic modeling of application performance on real time, so that dynamic performance adaptation can be leveraged to control energy-consumption. The energy saving will be achieved by reducing network usage, placing DataNodes in low-consumption nodes or using hybrid data centers with suitable consolidation tasks strategy.

Our ongoing work in the third point, high level management, is continuing in the way of adding these new parameters (as Risk, Trust, Revenue,

Power efficiency, …) in the resource allocation process that a PaaS layer offers to SaaS layer.

Our work in the fourth focus is to leveraging hybrid systems to improve energy-saving. This work will address the problem of low-level programmability of hybrid systems that can result in poor resource utilization and, in turn, poor energy efficiency.

Finally, mentioning that BSC is participating in three projects in the area of Green Computing: the Spanish NUBA project (2009-2011), the EU OPTIMIS project (2010-2013), and the EU COST IC804 action (2009-2012).

7. ACKNOWLEDGMENTS

Many thanks to BSC researchers for their research effort and contributions to this field.

This work is partially supported by the Ministry of Science and Technology of Spain (contract TIN2007-60625), the NUBA project under contract MITyC TSI-020301-2009-30 , the Generalitat of Catalunya (2009-SGR-980), the European Commission in the context of the HiPEAC2 network of excellence (contract no. ICT-217068), the COST (European Cooperation in Science and Technology) framework, under Action IC0804, the MareIncognito and SoW Adaptive Systems project under the BSC-IBM collaboration agreement.

8. REFERENCES

[1] J. Guitart, J. Torres, and E. Ayguadé. *A Survey on Performance Management for Internet Applications*. Concurrency and Computation: Practice and Experience, Vol. 22 (1),pp.68-106, Jan 2010. ISSN: 1532-0634

[2] J. Ejarque, M. de Palol, I. Goiri, F. Julià, J. Guitart, R. Badia, and J. Torres. *Exploiting Semantics and Virtualization for SLA-driven Resource Allocation in Service Providers*. Concurrency and Computation: Practice and Experience, Vol. 22 (5), pp. 541-572, April 2010. ISSN: 1532-0634

[3] I. Goiri, F. Julià, J. Guitart, and J. Torres. *Checkpoint-based Fault-tolerant Infrastructure for Virtualized Service Providers*. 12th IEEE/IFIP Network Operations and Management Symposium (NOMS'10) Osaka, Japan, April 19-23, 2010, pp. 455-462 ISBN: 978-1-4244-5367-2, ISSN: 1542-1201

[4] J. Polo, D. Carrera, Y. Becerra, J. Torres, E. Ayguadé, M. Steinder, I. Whalley. *Performance-Driven Task Co-Scheduling for MapReduce Environments*. 12th IEEE/IFIP Network Operations and Management Symposium (NOMS'10). Japan, April, 2010.

[5] J. Polo, Y. Becerra, D. Carrera, V. Beltran, J. Torres and E. Ayguadé. *Towards Energy-Efficient Management of MapReduce Workloads.* Poster session. 1st Int'l Conf. on Energy-Efficient Computing and Networking (e-Energy 2010), 2010.

[6] J. Ll Berral, I. Goiri, R. Nou, F. Julia, J. Guitart, R. Gavaldà and J. Torres. *Towards energy-aware scheduling in data center using machine learning*. 1st Int'l Conf. on Energy-Efficient Computing and Networking (e-Energy), 2010.

[7] I. Goiri, J. Guitart, J. Torres. *Characterizing Cloud Federation for Enhancing Providers' Profit*. 2010 IEEE 3rd International Conference on Cloud Computing (CLOUD 2010).

[8] R. Bertran, M. Gonzalez, Y. Becerra, D. Carrera, J. Torres and E. Ayguade. *Towards Accurate Accounting of Energy Consumption in Shared Virtualized Environments*. Poster session. 1st Int'l Conf. on Energy-Efficient Computing and Networking (e-Energy'10). April, 2010.

[9] J. Polo, D. Carrera, Y. Becerra, V. Beltran, J. Torres, E. Ayguadé. *Performance Management of Accelerated MapReduce Workloads in Heterogeneous Clusters.* 39th International Conference on Parallel Processing (ICPP2010). San Diego, CA, USA. September 2010.

Energy-Efficient Management of Physical and Virtual Resources - A Holistic Approach

Gergö Lovász, Florian Niedermeier, Hermann de Meer
University of Passau
Innstrasse 43, 94032 Passau, Germany
{gergoe.lovasz,florian.niedermeier,demeer}@uni-passau.de

***Abstract*—The spreading of information and communication technology has contributed much to the reduction of energy consumption in many areas of everyday life. Nevertheless the energy consumption of information and communication technology itself is rapidly growing and has to be dealt with. Currently used approaches focus mostly on the reduction of hardware energy consumption. This paper presents a vision of a holistic approach for reducing energy consumption in future communication infrastructures. Beside energy-efficient hardware as well as protocols that support the energy-efficient operation of communicating devices, the main focus of this paper is energy-efficient resource management. According to the Principle of Economic Efficiency, a limitation of resource provision is suggested by encapsulating applications in virtual machines with fixed resource requirements, together with the determination of an energy-minimal subset of resources on which applications are consolidated and which is able to fulfill the application requirements without over-provisioning.**

I. Introduction

The energy consumption of IT and communication infrastructures is dramatically increasing. A doubling of energy consumption from 2000 to 2005 of volume, mid-range, and high-end servers in the U.S. and worldwide is reported by Koomey [1]. In the year 2000 the total energy consumption of servers, routers, and PCs in Germany was about 5 billion kWh. For the year 2010 an energy consumption of more than 55 billion kWh per year is expected for the information and communication technology (ICT) in Germany [2].

Nowadays there are several approaches that address energy-efficient computing and communication. The most common approach is to develop more efficient hardware that consumes less energy and offers special energy efficiency features, e.g. energy saving modes. This effort is fostered by labels like the US Energy Star[1] or the European TCO Certification[2] who rate IT products (mostly monitors) for their environmental properties. But the development of energy-efficient hardware alone could not slow down or even stop the trend of increasing ICT energy consumption. Whereas it is important to develop hardware with low energy consumption and energy efficiency features, a holistic approach is needed to make ICT ”green”. A new energy-aware resource management has to be applied which considers the quality of service (QoS) requirements of applications on the one hand but also takes into account the energy consumption that is needed to perform a certain task on the other hand. Such a resource management aims at a resource allocation that is able to provide the QoS that is required by the applications and at the same time minimizes the energy that is needed to provide the requested services. Also energy-efficient application layer and communication layer protocols are needed, that support the energy-efficient operation of the hardware and an energy-efficient resource management by taking into account the energy-saving features of communicating devices, like power-saving modes. Section 2 gives a brief overview on energy-efficiency features of hardware, section 3 describes the problems with the currently used resource management paradigm and presents an alternative, energy-efficient way of managing resources. Section 4 describes the vision of an autonomous and energy-efficient resource management. Section 5 points out the weaknesses of currently used communication protocols with respect to energy efficiency and suggests methods to enable an energy-efficient communication. A conclusion is given in section 6.

II. Energy-Efficient Hardware and its Features

Computer power can be saved by means of various techniques. First, the processor can be powered down by mechanisms like SpeedStep [3], PowerNow[3], Cool’n’Quiet[4], or Demand-Based Switching[5]. These measures enable slowing down the clock speeds (Clock Gating), or powering off parts of the chips (Power Gating), if they are idle [4],[5]. By sensing user-machine interaction, different hardware parts can incrementally be turned off or put in hibernating mode (display, disk, etc.). The ACPI specification defines four different power states which an ACPI-compliant computer system can be in. These states reach from G0-Working to G3-Mechanical-Off. The states G1 and G2 are subdivided into further states that describe which components are switched off in the particular state. For devices and the CPU separate power states (D0-D3 for devices and C0-C3 for CPUs) are defined which are similar to the global power states[6]. Some of the mentioned techniques are usually applied to mobile devices but can be

[1]http://www.energystar.gov
[2]http://www.tcodevelopment.com

[3]http://www.amd.com/us-en/Processors/ProductInformation/0,,30_118_10220_10221%5E964,00.html
[4]http://www.amd.com/us-en/Processors/ProductInformation/0,,30_118_9485_9487%5E10272,00.html
[5]http://softwarecommunity.intel.com/articles/eng/1611.htm
[6]http://www.acpi.info/DOWNLOADS/ACPIspec30.pdf

used for desktop PCs as well. Energy saving techniques are also adopted by data centers to reduce power consumption. The overall power consumption of a data center consists of two major components: the power consumption of routers, servers, storage, switches and other devices on the one hand and the power consumption of the air-conditioning system (for hardware cooling) on the other hand. The energy efficiency standards of the hardware which is used in data centers exceeds the energy efficiency of ordinary PCs and hardware for personal usage in most cases.

III. The Principle of Economic Efficiency - A Resource Management Paradigm

The Principle of Economic Efficiency states two paradigms: The maximization and the minimization principle. The first aims at maximizing profit by optimizing the amount that is being produced with a given set of resources. The latter works in the opposite direction, minimizing the resources needed to provide a predefined output [6]. By applying the minimization principle on IT infrastructure, a massive gain in energy-efficiency can be expected.

A. Resource management using the maximization principle

Current resource management relies on the paradigm of maximization. This means that at any time, managed resources will provide a maximum of performance whereas energy (resource) consumption is in most cases only a minor concern. The hardware will always deliver the maximum available performance in order to provide the best service possible and scales its input power accordingly. Traditionally, application execution speed is only limited by the hardware capabilities. Application developers therefore allocate resources in a conservative way, meaning over-provisioning of resources in most cases and not taking into account the real requirements of the applications.

B. Resource management using the minimization principle

By performing the transition to the minimization principle, a fundamentally new paradigm is created. Managed resources should not provide maximum performance at any given time, but instead only deliver the amount of performance that is really needed by using as few resources as possible. Two key steps are necessary to implement the minimization principle:

1) Predefinition of the output
   Often, especially in times with no or little load, the performance of components can be reduced without affecting user experience (e.g. lower connection speed, lower CPU frequency). In load-balanced environments, even complete machines may be shut down. By deploying services inside a VM, it is possible to effectively and predictably predefine the resource impact of a service. This predictability enables a resource management system to achive high utilizations in data center machines by deploying VMs in a "'best fit"' way. By carefully examining the required level of QoS, VM parameters can be set according to real resource needs.
2) Minimization of the input
   Finally, knowledge about the resource requirements of the applications and capabilities of the underlying hardware are used to find a near energy-minimal subset of resources. This subset is defined as a set of hardware resources that provides all requested services (within the predefined QoS requirements) at a given time, while consuming the least possible amount of energy in a defined environment.
   Let R be the set of available hardware resources in a certain environment (e.g., a federation of data centers, or a cloud computing environment) and let A be the set of applications (e.g., a mail or web service) with predefined QoS requirements that have to be provided. Let $energy(A, X)$, with $X \subseteq R$ be the energy needed to run A on a subset of hardware resources X. $E \subseteq R$ is an energy-minimal subset of hardware resources for A if
   a) E can provide the required QoS to A,
   b) $energy(A, E) \leq energy(A, F)$, where F, $(F \subseteq R)$ is any other subset of hardware resources that can provide the required QoS to A.

   If two subsets of hardware provide the requested services in an energy-minimal way, the preferred subset is the set that provides the best QoS. Finding $E \subseteq R$ is an instance of the multidimensional bin-packing problem and therefore an NP-hard problem. However, even an approximation of $E \subseteq R$ could lead to a considerable reduction of energy consumption since resources that are not needed can be turned off or set into an energy-saving mode. Recently, a feature called live migration is becoming available in an increasing number of VMMs. This feature enables a nearly seamless migration of a running service inside a virtual machine from one physical host to another. Therefore it is possible to dynamically determine the energy-minimal subset of resources for a given set of applications and reallocate resources by consolidation of applications at runtime.

IV. Energy-efficient Resource Management

In this section a vision of an autonomic and energy-efficient monitoring is presented which realizes a resource management according to the minimization principle and is based on energy-related monitoring and energy consumption models. Figure 1 shows the main components of such a resource management system.

Main components are energy consumption models of the managed devices (e.g. servers, routers) that are needed to compute estimations of the energy consumption of different resource allocations. A model describes the energy consumption of a resource depending on the load of its sub-components. Therefore the model is devided into static properties and dynamic properties. Whereas the static properties describe the characteristics of hardware and usually do not change (e.g.

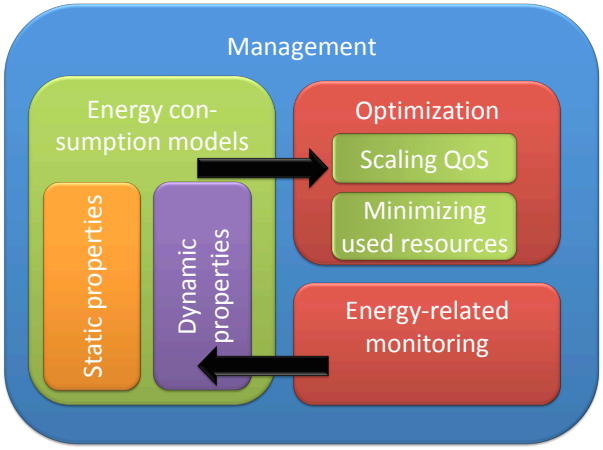


Fig. 1. Energy-efficient management of resources based on energy consumption models and energy-related monitoring

number of CPU cores, size of RAM), the dynamic properties strongly depend on the load on the resource and its components (e.g. the load on a server's CPU, used RAM). To identify the energy-relevant components and properties of a hardware resource, detailed measurements have to be performed. In a first stage, the single components are stress tested and the influence on the total energy consumption is measured. In a second stage, the cross-correlation of different components is analyzed. By combining the results of both stages, a generic energy consumption model can be derived for the resource that will have static and dynamic input parameters.

Another main component of the resource management system is the energy-related monitoring of resources. Realtime monitoring has to be applied to provide the dynamic, energy-relevant monitoring data as input to the energy consumption models. This includes especially the load of certain resource components (e.g. CPU, HDD). But also environmental properties, like the temperature, have to be monitored. This information can then be used for the distribution of server load to avoid hot spots in the equipment and to minimize the energy consumption of the air conditioning.

A key component of the management system is the optimizer. Through the encapsulation of applications in virtual machines it is able to exactly determine the application requirements (predefined output). The energy consumption models and the energy-related monitoring allows the optimizer to compute different potential resource allocations, estimate their power consumption, and approximate the energy-minimal subset of resources (minimal input).

V. Energy-Efficient Communication Protocols

Another critical area that has to be addressed in order to reduce ICT's energy consumption are communication protocols. Whereas communicating devices often offer mechanisms to save energy, communication protocols prevent the application of certain energy-saving features. Communication protocols have to be analyzed, concerning the energy-efficient support of network elements and their features. The protocols have to be designed in a way that the availability of certain network services does not assume the permanent accessibility of all virtual or physical entities. This enables devices to change into an energy-efficient mode for certain time periods. Protocols have to support the synchronization of communication, the delegation of services, on-demand mechanisms, and an energy-efficient signaling. Not all services do necessarily require permanent and direct accessibility, for instance.

- **Synchronization of communication**
  Communication between two entities can be synchronized in such a case, to allow the devices to hibernate between two communication phases without interrupting the availability of the service. Protocol mechanisms have to be developed which allow the synchronization of active communication phases. Devices should hibernate not only when no communication is taking place but also if they are lowly utilized (e.g. persistent TPC connections).
- **Delegation of services and functions**
  A network device should be enabled to delegate services to other devices, if possible (clients should not be aware of this delegation). Such a delegation would allow the transfer of a service from an energy inefficient to a rather energy efficient device or to a device which has to be always on anyway (e.g. a router). The delegating device can change to dormant mode or can be turned off. The delegated device can either provide the services itself or wake up the delegating device in case of a request. In order to keep the response time low for the requesting device in the latter case, the delegated device can take over first communication steps (e.g. connection establishment) while waiting for the delegating device to change to active mode. Alternatively, nodes can be activated on demand, instead of being permanently available.
- **Get rid of heart-beats, power on network devices on demand**
  Current protocols, like TCP or BGP, rely on keep-alive messages to determine the reachability of a host. These keep-alive messages prevent hardware from entering sleep states while at the same time being available in the network. The protocol stack has to be modified to not assume constant reachability of each device. Instead, device need to be able to enter a new state that allows them to be marked as available, but on standby. This state should allow machines to enter a sleep mode and only be waked up if actual communication with this machine is attempted. Mechanisms have to be developed that allow entities in the network to buffer context information in order to minimize signaling when a sleeping device changes to active mode.
- **Signaling**
  Currently, signaling and data traffic share the same physical links, although they have significantly different key features. While data traffic is likely to need high bandwidth and occur in bursts, signaling packets are very

small in size and more evenly spread over time. Using the same link for both means that in idle periods the capacity of the link is much larger than needed for signaling. Using a seperate, low bandwidth link exclusively for signaling would allow the high performance link to be shut down in idle periods to save energy. Alernatively, variable speed connections could be established between the network devices, allowing a low speed to be negotiated in idle periods. This lower speed of the link could in turn enable processors used inside switches and computer NICs to scale down their frequency.

VI. Conclusion

To counter the rapid increase in ICT energy consumption, a new paradigm has to be introduced. The transition from maximization to minimization principle is driven by the restriction of output to the needed levels and minimization of the power consumption. We implement this principle by using virtual machines to encapsulate services and consolidate them onto a near energy-minimal subset of the given physical resources. To achive an autonomous management of the infrastructure, energy consumption has to be modeled and predicted using real-time monitoring data. To achieve a further reduction of energy consumption, current communication protocols have to be replaced by energy-aware protocols to allow devices to use sleep states more effectively.
Future research will include a prototype implementation of the envisioned management system on the G-Lab testbed.

Acknowledgment

The authors would like to thank The German Ministry for Education and Research (BMBF) who is funding the project G-Lab_Ener-G that deals with energy efficiency in future networks.

References

[1] Koomey. Estimating total power consumption by servers in the us and the world, Technical report, 2007, http://enterprise.amd.com/Downloads/svrpwrusecompletefinal.pdf

[2] Fraunhofer ISI. Der Einuss moderner Gertegenerationen der IuK-Technik auf den Energieverbrauch in Deutschland bis zum Jahr 2010, Studie. http://www.isi.fhg.de/e/publikation/iuk/Fraunhofer-IuK-Kurzfassung.pdf

[3] Intel white paper 30057701. Wireless Intel SpeedStep Power Manager: Optimizing Power Consumption for the Intel PXA27x Processor Family, sunsite.rediris.es/pub/mirror/intel/pca/applicationsprocessors/whitepapers/30057701.pdf, 2004

[4] ENERGY STAR* System Implementation, Published by Intel with technical collaboration from the U.S. Environmental Protection Agency, Whitepaper, 2007, Revision-001, http://www.energystar.gov/ia/partners/prod_development/revisions/downloads/316478-001.pdf

[5] Windeck. Energy Star 4.0, C't German Magazine for Computer Techniques, Vol. 14, 52-53, 2007

[6] E. Grochla and E. Gaugler, Eds., Handbook of German Business Management. Poeschel, Stuttgart, Springer, Berlin/Heidelberg, 1990, vol. 1 (A-E).

Including Energy Efficiency into Self-adaptable Cloud Services

Ivona Brandic, Vincent C. Emeakaroha, Michael Maurer, Schahram Dustdar
Distributed Systems Group, Vienna University of Technology, Vienna, Austria
{ivona,vincent,maurer,dustdar}@infosys.tuwien.ac.at

***Abstract*—Nowadays, novel computing paradigms as for example Cloud Computing are gaining more and more on importance. In case of Cloud Computing users pay for the usage of the computing power provided as a service. Beforehand they can negotiate specific functional and non-functional requirements relevant for the application execution. However, providing computing power as a service bears different research challenges. On one hand dynamic, versatile, and adaptable services are required, which can cope with system failures and environmental changes. On the other hand, energy consumption should be minimized. In this paper we present the first results in establishing adaptable, versatile, and dynamic services considering negotiation bootstrapping and service mediation achieved in context of the Foundations of Self-Governing ICT Infrastructures (FoSII) project. We discuss novel meta-negotiation and SLA mapping solutions for Cloud services bridging the gap between current QoS models and Cloud middleware and representing important prerequisites for the establishment of autonomic Cloud services.**

***Index Terms*—Cloud Computing; SLA management; autonomic computing;**

I. Introduction

Service-oriented Architectures (SOA) represent a promising approach for implementing ICT systems [1]. Thereby, software is packaged to services and can be accessed independently of the used programming languages, protocols, and platforms. Despite remarkable adoption of SOA as the key concept for the implementation of ICT systems, the full potential of SOA (e.g., dynamism, adaptivity) is still not exploited [3]. SOA approach and Web service technologies represent large scale abstractions and a candidate concept for the implementation novel computing paradigms where sophisticated scientific applications can be accessed as services over Internet [2] or where massively scalable computing is made available to end users as a service as in case of *Cloud Computing* [4]. In all those approaches the access to computing power is provided as a service.

The key benefits of providing computing power as a service are (a) avoidance of expensive computer systems configured to cope with peak performance, (b) pay-per-use solutions for computing cycles requested on-demand, and (c) avoidance of idle computing resources. The development of novel concepts for dynamic, versatile, and adaptive services represents an open and challenging research issue [5]. Major goal of this paper is to facilitate service negotiation in heterogeneous Clouds. In order to enable service users to find services which best fit to their needs (considering costs, execution time and other functional and non-functional properties), service users should negotiate and communicate with numerous publicly available services.

Non-functional requirements of a service execution are termed as *Quality of Service (QoS)*, and are expressed and negotiated by means of *Service Level Agreements (SLAs)*. *SLA templates* represent empty SLA documents with all required elements like parties, SLA parameters, metrics and objectives, but without QoS values. However, most existing Cloud frameworks assume that the communication partner knows about the *negotiation protocols* before entering the negotiation and that they have matching *SLA templates*. In commercially used Clouds this is an unrealistic assumption since services are discovered dynamically and on demand. Thus, so-called *meta-negotiations* are required to allow two parties to reach an agreement on what specific negotiation protocols, security standards, and documents to use before starting the actual negotiation. The necessity for SLA mappings can be motivated by differences in terminology for a common attribute such as *price*, which may be defined as *usage price* on one side and *service price* on the other, leading to inconsistencies during the negotiation process.

Thus, we approach the gap between existing QoS methods and Cloud services by proposing an architecture for Cloud service management with components for *meta-negotiations* and *SLA mappings* [9]. Meta-negotiations are defined by means of a *meta-negotiation document* where participating parties may express: the pre-requisites to be satisfied for a negotiation, for example, requirement for a specific authentication method; the supported negotiation protocols and document languages for the specification of SLAs; and conditions for the establishment of an agreement, for example, a required third-party arbitrator. SLA mappings are defined by XSLT[1] documents where inconsistent parts of one document are mapped to another document e.g., from consumer's template to provider's template. Moreover, based on SLA mappings and deployed taxonomies, we eliminate semantic inconsistencies between consumer's and providers SLA template.

II. Overview

To facilitate dynamic, versatile, and adaptive IT infrastructures, SOA systems should react to environmental changes,

[1]XSL Transformations (XSLT) Version 1.0, http://www.w3.org/TR/xslt.html

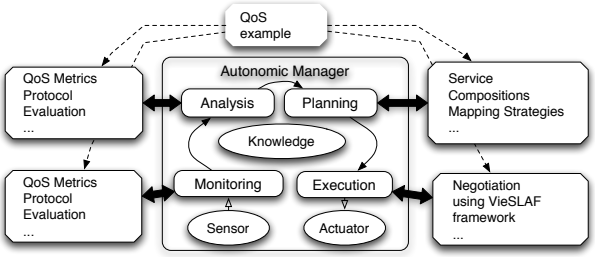


Fig. 1. General Architecture of an Autonomic System Explained on a QoS Example

software failures, and other events which may influence the systems' behavior. Therefore, adaptive systems exploiting self-* properties (self-healing, self-controlling, self-managing, etc.) are needed, where human intervention with the system is minimized. We propose models and concepts for adaptive services building on the approach defined by means of autonomic computing [6], [7].

We identified the following objectives:

- **Negotiation bootstrapping and service mediation.** The first objective is to facilitate communication between publicly available services. Usually, before service usage, service consumer and service provider have to establish an electronic contract defining terms of use [8]. Thus, they have to negotiate the exact terms of contract (e.g., exact execution time of the service). However, each service provides a unique negotiation protocol often expressed using different languages, representing an obstacle within the SOA architecture. We propose novel concepts for automatic bootstrapping between different protocols and contract formats increasing the number of services a consumer may negotiate with. Consequently, the full potential of public services could be exploited.
- **Service Enforcement** Services may fail, established contracts between services may be violated. The second objective is to develop methods for service enforcement, where failures and malfunctions are repaired on demand and where services are adapted to changing environmental and system conditions. We propose development of knowledge bases where the directives, policies, and rules for failure adjustment and repair may be specified and stored. Furthermore, adequate methods for the condition specification and condition evaluation are emerging research issues.
- **Service adaptivity** Service failures or violations of electronic agreements must be detected in an efficient manner. Moreover, the reaction to failures should be done in an adequate way. Thus, the third objective is the development of novel methods for modeling of intelligent logging capabilities at the level of a single service as well as composite services. Sophisticated concepts for the measurement of service execution parameters and Quality of Service (QoS) are needed as well as generic monitoring capabilities which can be customized on-demand for different services.

In order to achieve aforementioned goals we utilize the principles of *autonomic computing*. Autonomic computing research methodology can be exemplified using Quality of Service (QoS) as shown in Figure 1. The management is done through the following steps: (i) *Monitoring*: QoS managed element is monitored using adequate software sensors; (ii) *Analysis*: The monitored and measured metrics (e.g., execution time, reliability, availability, etc.) are analyzed using knowledge base (condition definition, condition evaluation, etc.); (iii) *Planning*: Based on the evaluated rules and the results of the analysis, the planning component delivers necessary changes on the current setup e.g., renegotiation of services which do not satisfy the established QoS guarantees; (iv) *Execution*: Finally, the planned changes are executed using software actuators and other tools (e.g., *VieSLAF* framework [9]), which query for new services.

A. Negotiation Bootstrapping and Service Mediation

Autonomic computing can be applied for other managed elements e.g., service negotiation. In the following we explain the first steps in achieving aforementioned architecture: *meta-negotiations* and *SLA mappings*.

Figure 2 depicts how the principles of autonomic computing can be applied to negotiation bootstrapping and service mediation. As a prerequisite of the negotiation bootstrapping users have to specify a meta-negotiation document describing the requirements of a negotiation, as for example required negotiation protocols, required security infrastructure, provided document specification languages, etc. During the *monitorig phase* all candidate services are selected where negotiation bootstrapping is required. During the *analysis phase* existing knowledge base is queried and potential bootstrapping strategies are found. In case of missing bootstrapping strategies users can define in a semi-automatic way new strategies (*planning phase*). Finally, during the *execution phase* the negotiation is started by utilizing appropriate bootstrapping strategies.

The same procedure can be applied to service mediation. During the service negotiation, inconsistencies in SLA templates may be discovered (*monitoring phase*). During the *analysis phase* existing SLA mappings are analyzed. During the *planning phase* new SLA mappings can be defined, if existing mappings cannot be applied. Finally, during the *execution phase* the newly defined SLA mappings can be applied.

III. Meta-Negotiations

In this section, we present an example scenario for the meta-negotiation architecture, and describe the document structure for publishing negotiation details into the meta-negotiation registry.

A. Meta-Negotiation Scenario

The meta-negotiation infrastructure can be employed in the following manner: (i) *Publishing*: A service provider publishes descriptions and conditions of supported negotiation protocols

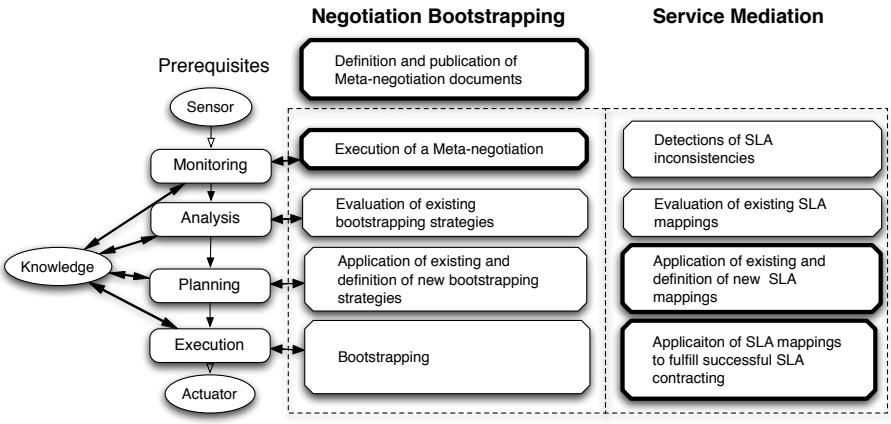


Fig. 2. Negotiation Bootstrapping and Service Mediation as Part of the Autonomic Process

into the registry; (ii) *Lookup*: Service consumers perform lookup on the registry database by submitting their own documents describing the negotiations that they are looking for. (iii) *Matching*: The registry discovers service providers who support the negotiation processes that a consumer is interested in and returns the documents published by the service providers; (iv) *Negotiation*: Finally, after an appropriate service provider and a negotiation protocol is selected by a consumer using his/her private selection strategy, negotiations between them may start according to the conditions specified in the provider's document.

In the following we explain the sample meta-negotiation document.

B. Meta-Negotiation Document (MND)

The participants publishing into the registry follow a common document structure that makes it easy to discover matching documents. This document structure is presented in Figure 3 and consists of the following main sections.

Each document is enclosed within the `<meta-negotiation>` ... `</meta-negotiation>` tags. Each meta-negotiation (MN) comprises three distinguishing parts, namely *pre-requisites*, *negotiation* and *agreement* as described in the following paragraphs.

a) Pre-requisites: The conditions to be satisfied before a negotiation starts are defined within the `<pre-requisite>` element (see Figure 3, lines 3–10). Pre-requisites define the *role* a participating party takes in a negotiation, the *security credentials* and the *negotiation terms*. The `<security>` element specifies the authentication and authorization mechanisms that the party wants to apply before starting the negotiation process. The negotiation terms specify QoS attributes that a party is willing to negotiate and are specified in the `<negotiation-term>` element. For example, in Figure 3, the negotiation terms of the consumer are *beginTime* and *endTime*, and *price* (line 6).

b) Negotiation: Details about the negotiation process are defined within the `<negotiation>` element. Each document language is specified within the `<document>` element. In Figure 3, *WSLA* is specified as the supported document language. Additional attributes specify the URI to the API or WSDL for the documents and their versions supported by the consumer. In Figure 3, *AlternateOffers* is specified as the supported negotiation protocol. In addition to the *name*, *version*, and *schema* attributes, the URI to the WSDL or API of the negotiation protocols is specified by the *location* attribute (line 12).

c) Agreement: Once the negotiation has concluded and if both parties agree to the terms, then they have to sign an agreement. This agreement may be verified by a third party organization or may be logged with another institution who will also arbitrate in case of a dispute. These modalities are specified within the `<agreement>` clause of the meta-negotiation document as shown in line 14.

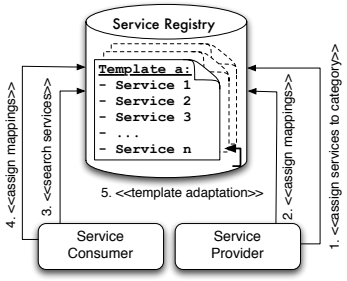


Fig. 4. Management of SLA-Mappings

IV. SLA MAPPINGS

In the presented approach each SLA template has to be published into a registry where negotiation partners i.e., provider and consumer, can find each other.

A. Management of SLA mappings

Figure IV-A depicts the architecture for the management of SLA mappings and participating parties. The registry comprises different *SLA templates* whereby each of them represent a specific application domain, e.g., SLA templates for medical, telco or life science domain. Thus, each service provider may assign his/her service to a particular template (see step 1 in Figure IV-A) and afterwards assign SLA mappings if

```
1. <meta-negotiation ...>
2.  <pre-requisite>
3.   <role name="consumer"/>
4.   <security> <authentication value="GSI" location="uri"/> </security>
5.   <negotiation-terms>
6.    <negotiation-term name="beginTime"/> <negotiation-term name="endTime"/>
8.   </negotiation-terms>
9.  </pre-requisite>
10. <negotiation>
11.  <document name="WSLA" value="uri" version="1.0"/>
12.  <protocol name="alternateOffers" schema="uri" version="1.0" location="uri"/>
13. </negotiation>
14. <agreement> <confirmation name="arbitrationService" value="uri"/> </agreement>
15.</meta-negotiation>
```

Fig. 3. Example Meta-negotiation Document

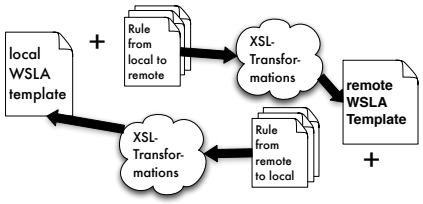


Fig. 5. Scenario for XSL Transformations

necessary (see step 2). Each template *a* may have *n* services assigned.

Service consumer may search for the services using meta-data and search terms (step 3). After finding appropriate services each service consumer may define mappings to the appropriate template the selected service is assigned to (step 4). Thereafter, the negotiation between service consumer and service provider may start as described in the next section. As already mentioned templates are not defined in a static way. Based on the assigned SLA mappings and the predefined rules for the adaptation, SLA templates are updated frequently trying to reflect the actual SLAs used by service provides and consumers (step 5).

Currently, SLA mappings are defined on an XML level, where users define XSL transformations. However, a UML based GUI for the management of SLA mappings is subject of ongoing work.

B. Scenario for SLA mappings

Figure 5 depicts a scenario for defining XSL transformations. For the definition of SLA agreements we use Web Service Level Agreement (WSLA). WSLA templates are publicly available and published in a searchable registry. Each participant may download previously published WSLA templates and compare them with the local template. This can be done in an automatic way by using appropriate tools. We are currently developing a GUI that can help consumers to find suitable service categories. If there are any inconsistencies discovered, service consumer may write rules (XSL transformation) from his/her local template to the remote template. The rules can also be written by using appropriate visualization tools. Thereafter, the rules are stored in the database and can be applied during the runtime to the remote template. During the negotiation process, the transformations are performed from the remote WSLA template to the local template and vice versa.

Figure 5 depicts a service consumer generating a WSLA. The locally generated WSLA plus the rules defining transformation from local WSLA to remote WSLA, deliver a WSLA which is compliant to the remote WSLA. In the second case, the remote template has to be translated into the local one. In that case, the remote template plus the rules defining transformations from the remote to local WSLA deliver a WSLA which is compliant to the local WSLA. Thus, in this manner, the negotiation may be done using non-matching templates.

Even the service provider can define rules for XSL transformations from the publicly published WSLA templates to the local WSLA templates. Thus, both parties, provider and consumer, may match on a publicly available WSLA template.

ACKNOWLEDGMENT

The work described in this paper was partially supported by the Vienna Science and Technology Fund (WWTF) under grant agreement ICT08-018 Foundations of Self-governing ICT Infrastructures (FoSII).

REFERENCES

[1] A. P. Barros, M. Dumas. The Rise of Web Service Ecosystems. IT Professional 8(5): 31 – 37 , Sept.-Oct. 2006.

[2] J. Blythe, E. Deelman, Y. Gil. *Automatically Composed Workflows for Grid Environments*. IEEE Intelligent Systems 19(4): 16–23 2004.

[3] M.P. Papazoglou, P. Traverso, S. Dustdar, F. Leymann. Service-Oriented Computing: State of the Art and Research Challenges, IEEE Computer, 40(11): 64-71, November 2007

[4] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and Ivona Brandic. Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems, ISSN: 0167-739X, Elsevier Science, Amsterdam, The Netherlands, 2009, in press, accepted on Dec. 3, 2008.

[5] Foundations of Self-Governing ICT Infrastructures (FoSII) Project, http://www.infosys.tuwien.ac.at/linksites/FOSII

[6] J.O. Kephart, D.M. Chess, *The vision of autonomic computing*. Computer, 36:(1) pp. 41-50, Jan 2003.

[7] D. Ardagna, G. Giunta, N. Ingraffia, R. Mirandola and B. Pernici. QoS-Driven Web Services Selection in Autonomic Grid Environments. GADA 2006, International Conference, Montpellier, France, 2006.

[8] K. Czajkowski, I. Foster, C. Kesselman, V. Sander and S. Tuecke, *SNAP: A Protocol for Negotiating Service Level Agreements and Coordinating Resource Management in Distributed Systems*. 8th Workshop on Job Scheduling Strategies for Parallel Processing, Edinburgh Scotland, July 2002.

[9] I. Brandic, D. Music, Ph. Leitner, S. Dustdar. VieSLAF Framework: Enabling Adaptive and Versatile SLA-Management. Gecon09, In conjunction with Euro-Par 2009, 25- 28 August 2009, Delft, The Netherlands

FIT4Green – Energy aware ICT Optimization Policies

Robert Basmadjian, Christian Bunse, Vasiliki Georgiadou, Giovanni Giuliani, Sonja Klingert, Gergö Lovasz and Mikko Majanen

Abstract— Protecting the environment by saving energy and thus reducing carbon dioxide **emissions is one of today's hottest** and most challenging topics and is of a rapidly growing importance in the computing domain. The motivation and reasons for optimizing energy consumption from ecological and business perspectives are clear. However, the technical realization still is way behind expectations. One reason might be that technical problems range from pure hardware issues (e.g., low-power devices, energy harvesting, etc.) to software to cooling issues. This paper discusses recent findings and first ideas regarding policies and strategies for energy optimization and the development of a generic plug-in for managing data centers, accompanied by the **introduction of the concept of "Green Service Level Agreements (GSLA"). We discuss the general** structure (generic architecture) of the plug-in and sketch some of the embedded policies. It is also to be noted that all results are part of the recently started FIT4Green project, funded by the European Union.

Index Terms—FIT4Green, Energy, Optimization, Policies, SLA, Data Centre

I. Introduction

Over 500 million host computers, three billion PCs and mobile devices consume over a billion kilowatts of electricity per year. Following predictions of Greenpeace or EUROSTATS ICT consumes an increasing amount of energy, and is estimated to consume up to 20% of the global energy consumption by 2020. Traditionally, systems and network design seeks to minimize network cost and maximize quality of service (QoS). Electrical energy is needed for ICT both to operate and cool the equipment. This insight led to the Green-IT paradigm referring to environmentally sustainable computing or IT. Thus ICT can help reducing energy expenditure by substituting energy-intensive activities (e.g., E-Work, E-Commerce, or E-Learning) to optimizations of the software itself.

Manuscript received May 14, 2010. This work was partially funded by the Commission of the European Union under the 7th Framework Programme, ICT Call 4 – ICT for Energy Efficiency.

Robert Basmadjian and Gergö Lovasz are with the University of Passau, Innstraße 41, D-94032 Passau, Germany (e-mail: {Gergoe.Lovasz, Robert.Basmadjian}@Uni-Passau.de.

Christian Bunse and Sonja Klingert are with the University of Mannheim, A 5, 6, D-68131 Mannheim, Germany (e-mail: {Christian.Bunse, klingert}@uni-mannheim.de).

Vasiliki Georgiadou is with Almende B.V., Westerstraat 50, 3016 DJ Rotterdam, The Netherlands (e-mail: Vicky@almende.org).

Giovanni Giuliani is with HP Italiana srl, via Grandi 4, 20063 Cernusco S/N, Milan, Italy (e-mail: Giuliani@hp.com).

Mikko Majanen is with VTT Technical Research Centre of Finland, Kaitoväylä 1, FI-90571 Oulu, Finland (e-mail: mikko.majanen@vtt.fi).

The price of such substitution processes to the benefit of the environment is partly responsible for the sharp increase in the volume of data centre services. A study of the company Telecomspricing of November 2009 for instance predicts that the data centre revenue across 19 of the EU25 countries will increase with an annual growth rate of 25% per annum between 2010 and 2015 [1] – accompanied of course by a growing impact of data centers on the carbon footprint of mankind

In summary, ICT offers a way forward for reducing the consumption of energy and carbon emission by reducing land and air transport. However, this potential reduction is partially offset by the power used by data centers and computer networks [2]. Network and service providers have electrical costs reaching billions of EUR. Even a fraction of energy savings in networks could lead to reduced financial costs and carbon emissions. The importance of packet networks on energy consumption increases with the "convergence to the Internet Protocol (IP)" paradigm whereby most modes of communication, including mobile telephony, are increasingly supported by underlying packet networks. Since IP networks rely on network nodes and links, the electrical energy used for operating and cooling these equipment, creates a crucial need for research on energy saving strategies for networks. In order to solve these mentioned lacks, recently a great deal of research effort has been dedicated, especially to the following topics:

- Energy-efficient hardware
- Energy-efficient multiprocessor and Grid systems and data centers
- Energy-efficient clusters of servers
- Energy-efficient wireless and wired networks
- Energy-efficient cooling.

In FIT4Green, we aim at the development of a comprehensive view for energy efficiency, involving all layers ranging from technological to business aspects. The focus thereby is on single and federated data-centers supporting different computing styles (traditional-, super- and cloud computing). On a more technical level the FIT4Green strategies and tools includes physical nodes, cooling of nodes, networking hardware, communication protocols, the services themselves that are running on the nodes, up to business plans and service-level agreements (SLAs).

In detail within the FIT4Green project, a set of energy aware scheduling mechanisms and policies will be developed. More specifically, in the case of a single site data center, the idea is to provide algorithms to multiplex, de-multiplex

workload in order to save energy. This also includes findings regarding the trade-off between performance, quality-of-service (QoS), and energy consumption, In addition to scheduling issues the improvements of FIT4Green also benefits federated (cloud) environments.

Beneath this optimization the energy consumption might also benefit from software optimizations. Research has shown that especially communication is one of the largest cost factors with respect to energy. This makes communication an ideal candidate for optimizing a system's uptime. Energy optimization has to be aware of the tradeoffs between performance, energy consumption, and QoS. In contrast to hardware optimizations, software systems are usually optimized at development time by specifying their energy characteristics and by adapting the implementation. However, this requires individual adaptations of each system variant, and often implies a negative impact on the performance or QoS of such systems. The FIT4Green challenge is to explore the relations among the various components and to understand the tradeoffs. This enables the development of systems that achieve an optimal balance between performance, QoS, and energy consumption by adapting themselves at runtime (i.e., dynamic optimization).In conclusion, FIT4Green introduces a new paradigm of energy reducing efforts by creating an energy-aware plug-in placed on top of existing data centre automation frameworks. This paradigm will be realized with appropriate models and technology and tools that FIT4Green will develop, implement and test. The remainder of this paper is structured as follows. Section II provides a brief overview on the goals of the FIT4Green project and also gives a short summary of related work. Section III discusses the technical approach for reaching the previously defined goals. Based on this, Section IV introduces the generic plug-in architecture, and finally Section V provides a short summary and conclusions.

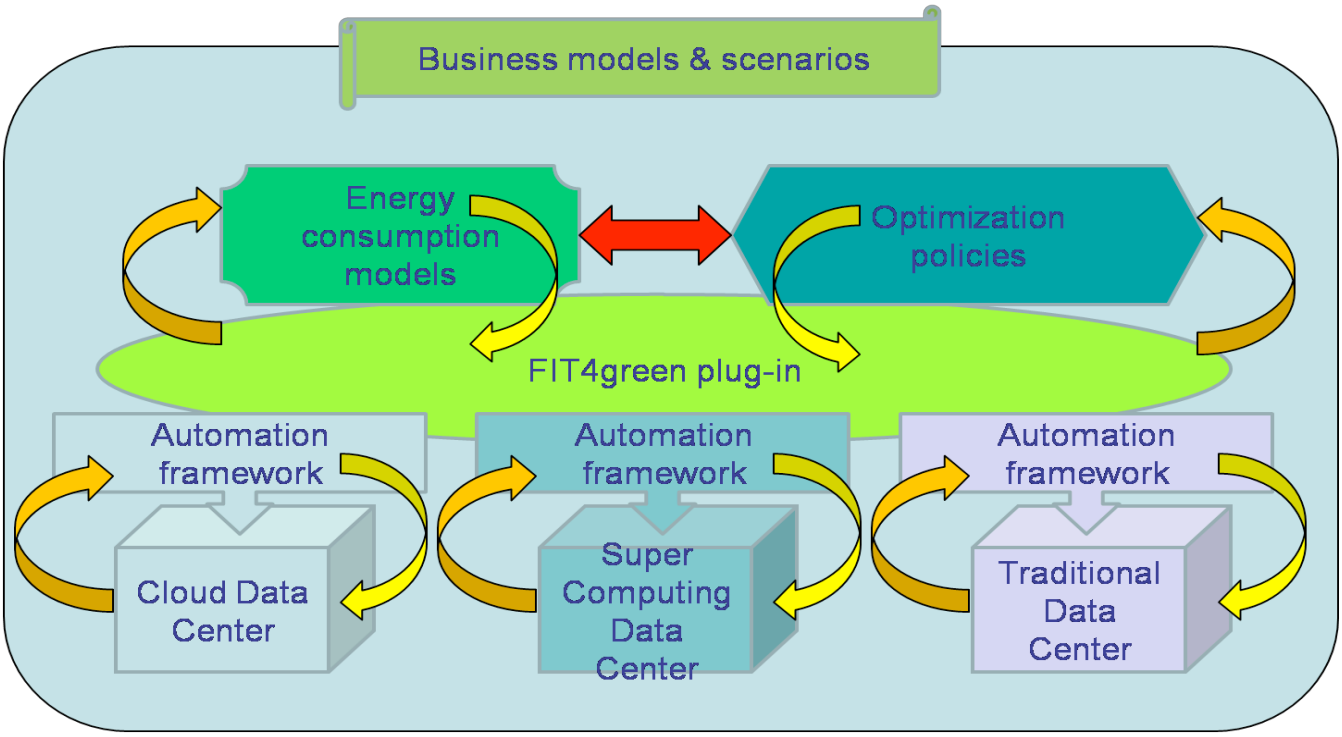


Figure 1. Overview of the FIT4Green technical approach

II. Project Goals

One goal of the FIT4Green project is to develop energy aware optimization policies for data centers, which - once applied - would reduce the energy consumption of their ICT infrastructure, without compromising compliance with Service Level Agreements (SLA) and Quality of Service (QoS) metrics. The FIT4Green approach will be potentially applicable to any type of data centre with any automation framework. Based on the specifics of the data centre and the scenario at hand the percentage of the energy reduction induced by applying these policies can vary; we envision that for a data centre with no previous steps with regard to energy optimization, FIT4Green policies and models can provide on average 20% saving in direct server and network devices' energy consumption and induce an additional 30% saving due to reduced cooling needs. The first estimate is based on recent "Data Center Energy Forecast Report" in which Accenture [3] illustrated a "State of the Art" exploitation of consolidation and optimization strategy of IT computing resources inside a data centre in Silicon Valley. Additional savings are possible via the reduction of cooling energy. This can be concluded from a study by HP and the Uptime Institute [4] that shows that "most data centre power is spent on cooling IT equipment (between 60 and 70%)".

Additionally, the federation of data centres will allow

lowering the overall Green House Gas (GHG) emissions (e.g., CO_2 emissions) by relocating applications and services to sites where lower environmental-impact power-generation may be available.

In addition to the technical goal to prove the energy saving potential of FIT4Green policies, the consortium aims at creating a lasting impact by supporting the data centre industry in the adoption of the FIT4Green policies instantiated in the plug-in. One issue in this context is the design of Green-SLAs that account for the modifications of the service delivery induced by FIT4Green. Another issue is to acknowledge the fact that a positive evaluation of the economic ROI is the number "1" criterion also for environmentally targeted investment decisions. Both issues will have a bearing on the design and especially on the exploitation phase of the project.

III. Technical Approach

Lately, many ICT players have been proposing focused solutions to optimize single component's energy consumption (low consumption servers, CPU speed scaling, power save modes, efficient cooling devices for data centres, etc.). These solutions enable the energy footprint reduction of single devices, treating them as isolated components and therefore lacking any savings obtainable through a holistic approach. Consolidation and virtualization techniques lead to energy savings through the reduction of the number of active servers; however, the current deployment strategies are fairly static and not guided by energy saving principles. Also, current SLAs do not include any metrics related to the ecological footprint.

FIT4Green goes beyond this state of the art with the global analysis of IT solutions deployment needs and the optimized deployment scheme inside a single site data centre as well as a federation of data centres with different energy related characteristics, considered as a global resource pool. By (re-) distributing computation and resources among several data centres, FIT4Green is able to capitalize on additional degrees of freedom with potential high impacts on the optimization strategies at federation level using, for example:

- Data centres at different geographical latitudes with the respective implications on cooling requirements based on external temperature, possibility to recycle heat through co-generation devices (different latitudes in the same hemisphere – capitalizing on temperature range variations – or different hemisphere – for seasonal changes)
- Data centres in regions where different sources of power generation are available, at different costs and GHG emissions
- Data centres inside different time zones with respect to the clients, allowing a different balance or mix of computing tasks.

Taking a global view on IT solutions (rather than a focused view on single components) and applying global optimization throughout the whole ICT-based process, FIT4Green technical approach (see Figure 1) includes the following topics:

- An optimization layer on top of existing data centre automation frameworks – integrated as a modular "plug-in" – to guide the allocation decisions based on the optimized energy model-based policies.
- Energy consumption models will be developed, and validated with real cases, for all ICT components in an IT solution chain, including the effects due to hosting data centres in sites with particular energy related characteristics, like alternative power availability and energy waste/recycle options, etc.
- Optimizations, based on policy modelling descriptions able to capture the variety of deployment that are possible for a given application or a set of applications, integrated with specific attributes supporting the evaluation of energy consumption models, will guide the deployments decisions on which/when equipments need to stay on, where/when applications should be deployed/relocated, also capitalizing on the intrinsic non linear behaviours of energy consumption growth with respect to the load of ICT components.

SLAs and business models will be analysed with respect to their potential impact on the carbon footprint of ICT. Consolidating these findings with implications of a deployment of the FIT4Green policies, new Green-SLAs and Green business model components will be developed taking into account the carbon footprint of data centre services. These will be integrated into the FIT4Green exploitation strategy.

Obviously, by moving applications and services to alternate data centres force the network traffic between client and servers to follow different paths, which have different impacts on the global energy consumption of the full process: both networks operated by telecommunication operators and local area networks will be considered in the global optimization schemes.

In this context, the typical distribution of clients is very significant as well: services with a global scope, e.g. search engines or Web 2.0 applications, receive requests from all over the world (following some distribution patterns bound to local times); on the other hand a local public administration service provider will most likely receive requests from a much more limited geographical area (and therefore time of day interval). The effect of the deployment of such services inside a federation of data centres implies a completely different parameterization of the model for the computation of the energy consumption: long term relocation of the public administration services to a data centre in the opposite hemisphere will force all users to have a much longer network path, while such effects do not show up so strongly if the services has an almost homogeneous distribution of clients around the globe. The picture changes again if it is possible to quickly relocate services, for instance with a "follow the sun" pattern; the energy impact of the service relocation (transition) needs to be considered in the overall computation for the optimal solution.

In the first phase of the project a set of FIT4Green scenarios

was developed: The computing styles traditional, cloud and supercomputing are dealt within the context of single site and federated site data centres. Each scenario highlights the special feature of the particular setting: The traditional computing

the results collected from the real test beds.

Finally, FIT4Green will investigate on the lessons learnt in the development of energy models, optimization policies and plug-ins for the various computing styles, and rationalize them

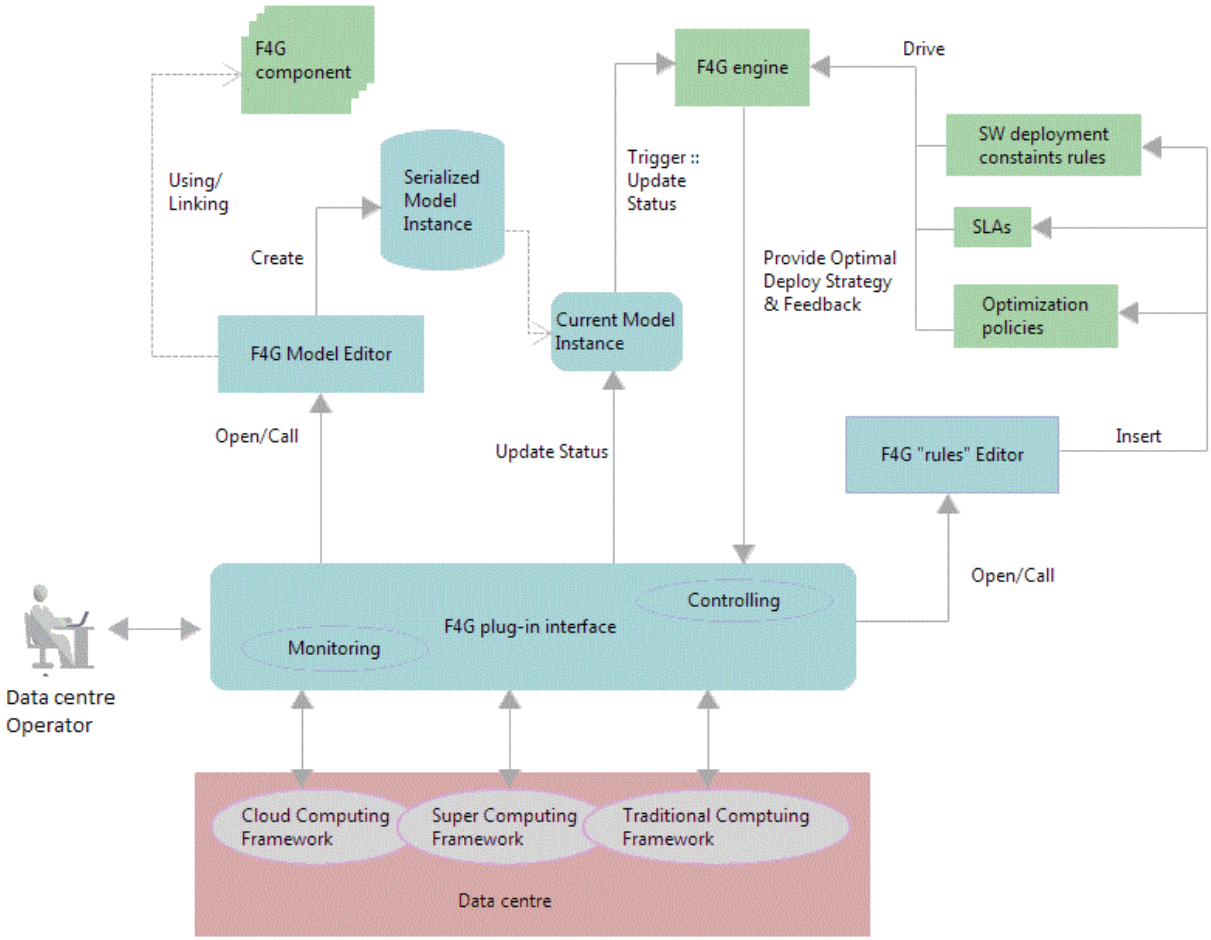


Figure 2. Schematic overview of the FIT4Green general architecture

scenarios, for instance, mainly deal with the challenge of deploying the FIT4Green energy saving strategies under the constraints of the data centre automation framework reacting rather slowly to the FIT4Green policies suggested by the plug-in. The automation framework in the cloud computing scenarios is much more flexible; however the plug-in has to cope with an unknown variety of applications and unforeseeable spikes in demand. There will be one pilot site for each computing style. Service/Enterprise Portal, Grid and Cloud pilots will support both single site and federated sites scenarios: multiple collaborating data centres inside the same organization for the Portal pilot; federation of supercomputer systems for Grid and open cloud federation of multiple labs for the Cloud.

Each pilot will implement the respective scenarios, measure the overall energy consumption and related cost reduction, apply the optimizations, evaluate the results and assess the expected impacts; this process will be iterated three times to allow energy models and optimizations to be refined based on

in a set of guidelines for the development of future IT solutions, which will intrinsically consider energy consumption and environmental footprint as essential design principles.

IV. General Architecture

Traditionally, the development of plug-ins for management & control software of data centers has followed the semantics defined by control design approach. In this context maintainability, fitness-for-change, coupling and cohesion are considered to be important structuring criterions. It is easier to keep consistency and completeness of information with a structured design approach. In the context of the FIT4Green project coupling and cohesion criteria as well as a model-based development process (e.g., using UML) have been considered for mapping functional requirements into components.

Moreover, for domain systems (i.e., the data center domain), a reference-architecture is needed that represents a domain specific way of structuring the control software plug-in

through decomposing the problems into parts and their relationships, and mapping them to software units and their interactions. To systematically achieve this structure of the plug-in, its functional and non-functional requirements as well as architecture styles and patterns are needed. Architecture styles denote well-known ways of structuring. Thus, the FIT4Green project follows an architecture-based development process comprised by the following steps:

1. Developing subsystems for the requirements: A set of subsystems is generated from functional and non-functional requirements, based on architectural styles and patterns.
2. Determining an actual architecture: These subsystems can be seen as components in a larger subsystem. Thus, functional view is described and transformed into a process view based on the considerations of parallelism.
3. Validating the solution: The architecture solution is validated using the quality scenarios, e.g., change scenario for modifiability, use scenario for performance, etc.

In the context of this process the systematic, concise and precise description of the plug-in structure(s) is of uttermost importance. It is the basis for all design activities including comprehending, communicating, analyzing, trading-off, as well as for modifications, maintenance, and reuse.

In the context of FIT4Green the architecture specification is based on mathematical, textual, and graphical notations. In order to manage the plug-in's inherent complexity the overall, general architecture specification is divided into multiple views.

Figure 2 presents the schematic overview of the FIT4Green general architecture. The FIT4Green plug-in is placed on the top of the existing data centre automation and management frameworks. The plug-in is divided into two parts: monitoring and controlling. The monitoring part updates the dynamic parameters of the current meta-model instance that describes the status of the data centre under optimization. The meta-model is initially built up by the data centre operator through the FIT4Green model editor. The latter offers to the operator a variety of FIT4Green components which may be used and linked in order to provide a model of both the structure and the features of the data centre. The FIT4Green components model various ICT components and their energy consumption.

The controlling part drives the FIT4Green optimization engine that finds the optimal deployment actions to reduce the energy consumption of the data centre with regard to the current status, SLAs and rules set up by the FIT4Green plug-in user. The rules consist of SW constraints and FIT4Green policies. Both constraints and policies can be edited with editors. The optimal deployment actions are reported back to the FIT4Green plug-in which invokes the appropriate data centre framework to execute them.

V. Summary and Conclusion

Recognizing that human-made greenhouse gas emissions are the major reason for global warming (i.e., green-house effect) created the urgent need to tackle environmental issues by adopting environmentally sound practices. This leads quite naturally to environmentally sustainable computing or IT, especially in the (large) data-center domain.

Within this paper the FIT4Green approach energy-aware computing for single or federated data-centers following different computing paradigms was introduced. FIT4Green provides energy saving strategies and policies and package these into a plug-in that can be used in the context of data-center control frameworks. Within the paper a generic plug-in architecture was introduced that outlines the general structure and approach. It is important to note that the architecture is kept as generic as possible in order to allow for "easy" instantiation and porting.

Based on individual assessments of the FIT4Green application partners it is currently expected to reach direct savings of 10-30% of the energy costs of a site. In addition, one can expect to save additional energy by reduced cooling needs. As soon as the FIT4Green policies are formally specified and implemented these hypotheses will be evaluated by several industrial-scale case studies.

Acknowledgment

The authors would like to thank all the partners and colleagues working in the FIT4Green project (http://www.fit4green.eu/).

References

[1] http://www.telecomspricing.com/news_detail.cfm?item=2592

[2] X. Fan, W.-D. Weber, and L. A. Barroso. "Power provisioning for a warehouse-sized computer", ISCA '07: Proc. 34th Annual International Symp. on Computer Architecture, pp. 13-23, 2007.

[3] "Data Center Energy Forecast Report" by Accenture: Final Report, July 29, 2008.

[4] C. Malone and C. Belady. "Metrics to Characterise Data Centre & IT Equipment Energy Use", Digital Power Forum, Richardson, Sep 2006 & C. Belady, "How to Minimise Data Centre Utility Bills", Sep 2006.

An Approach to Reduce the Energy Cost of the Arbitrary Tree Replication Protocol

Robert Basmadjian and Hermann de Meer

***Abstract*—Until recently, there have been no efforts of devising energy-efficient replication protocols for large-scale distributed systems. In this paper, we introduce an approach that reduces the energy cost of a particular tree-structured replication protocol. We show that, by shutting down some replicas and by a simple logical structural transformation (rearrangement), our approach achieves comparable characteristics as the original protocol, yet with much reduced energy cost as well as overall energy consumption. The logical transformation does not necessitate the reconfiguration of the protocol whenever energy efficiency requirements change.**

***Index Terms*—Energy efficiency, quorum systems, replica control protocols, load, availability, energy cost.**

I. Introduction

REPLICATION involves creating and maintaining duplicates of data in order to provide *fault tolerance* and to improve the *system performance*. However, when replication is used, data becomes susceptible to inconsistency problems. Therefore, *replica control protocols* (RCP), are required in order to maintain *data consistency* among the replicas.

Basically, such replica control protocols implement two operations; *read* (query) and *write* (update). To ensure *one-copy equivalence*, a read and write operations to two different copies of the data should not be allowed to execute concurrently. *Quorum systems*[1] are used by these protocols which serve as a basic tool of achieving one-copy equivalence.

Given the importance of the topic, several replica control protocols that enforce a specified semantics of accessing data have been described in the literature [18]. In general, these protocols can be classified into two categories: *structured* and *non-structured* RCPs. Only the former assumes that replicas of the system are arranged logically into a particular structure.

On the other hand, the concept of building *energy-efficient* distributed systems has attracted a great deal of attention lately due to the increase of *energy costs* (fuel) and the world wide desire to reduce CO_2 *emissions*. For instance, it was stated in [10] that storage centers can consume as much energy as a whole city if the number of servers reaches a certain level. Furthermore, in order to address to the problem of overheating, these storage centers are forced to be equipped with cooling systems, which as a matter of fact, increase the overall energy consumption. Therefore, minimizing and balancing the system's energy consumption becomes as important as the more traditional quality-of-service concerns, such as reliability, security, fault tolerance, and so forth.

One of the techniques to save energy costs, is to force some of the machines to enter into a sleep mode or even to be turned off. However, it is always desirable to keep the same overall performance level while minimizing the energy consumption of the system.

In this paper, we propose an approach to diminish the overall *energy consumption* of the *Arbitrary Tree* [3] protocol, where we compute the *energy cost* of its read and write operations. For this purpose, we adapt the energy cost model of [17] to the general approach of quorum systems. We show that, by shutting down significant amount of replicas of the system and by logically reorganizing the others into a new logical structure, it is possible to reduce the energy cost as well as consumption of [3] while preserving most of its characteristics. The tradeoff is that, in the worst case, the new approach has a worse load of 14% for its write operations than that of the original approach, however with much reduced energy cost as well as overall energy consumption.

The rest of this paper is organized as follows. Section 2 discusses related work. After representing in section 3 the energy cost model, we present in section IV the Arbitrary Tree protocol. Then, we introduce in section V our energy efficient approach. A comparison is given in section VI and the paper is concluded in section VII.

II. Related Work

As mentioned above, several replication protocols have been proposed in literature which make use of *quorum systems* to achieve data consistency among the replicas. They differ according to various parameters of their read and write operations such as the *quorum size*, the *availability*, as well as the *load* induced on the system.

The general fault-tolerance (availability) properties of quorum systems were examined in [15]. Also, the load of these systems was studied in [14] and it was shown that the *optimal load* of any quorum system of n replicas is $\frac{1}{\sqrt{n}}$ (highest is 1) if the smallest quorum is of size $\sqrt{n}$.

The *ReadOneWriteAll* (*ROWA*) [5] and *Majority Quorum Consensus* (MQC) [19] protocols have quorum sizes of $\mathcal{O}(n)$. By imposing a *logical structure* on the replicas, it is possible to reduce the quorum sizes further. Several protocols have been introduced which also make use of quorum systems and assume that the replicas are organized logically into a specific structure: *finite projective plane* [13], a *grid structure* [7] and [14], or a *tree structure* [1], [2], [12], [11], [8] and [3].

R. Basmadjian and H. de Meer are with Chair of Computer Networks and Communications, University of Passau, Innstrasse 43, 94034, Passau, Germany.

The full version of the paper appeared in the Proceedings of the ACM SIGCOMM 1st Int'l Conf. On Energy-Efficient Computing and Networking.

[1] A quorum system is defined as a set of subsets of replicas called quorums having pair-wise non-empty intersections.

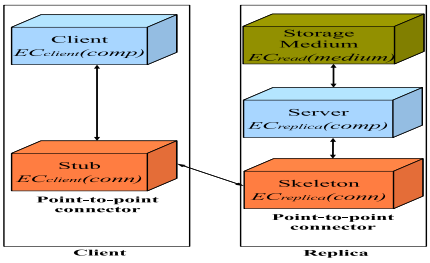


Fig. 1. Interactions of performing a read operation on a single replica.

In general, the *tree-structured* RCPs have a tight trade-off between the quorum size and the system load: a small size results in inducing a high load and vice versa. In [3], the *Arbitrary Tree* protocol was proposed and it was shown that its write operations induce an optimal system load of $\frac{1}{\sqrt{n}}$ with a quorum size of $\sqrt{n}$, which are lower than the state-of-the-art *tree-structured* RCPs, while preserving comparable write availability. On the other hand, it was proven that its read operations induce a quorum size of $\sqrt{n}$ which is smaller than previously proposed tree-structured RCPs with comparable system load and availability.

Recently, the need for saving the energy consumption of distributed systems composed of a large set of machines has attracted a great deal of research due to various ecological and economical concerns. Several techniques such as [6], [16], [9] have been proposed that deal with turning off some machines and shifting their load to other machines which have low thermic metrics. However, as it was stated in [20], these approaches are not appropriate for replication, because they assume that any request can be achieved by a number of currently active machines (replicas) which might have a stale copy of data. Our approach is also based on the idea of turning off replicas, however unlike [6], [16], [9], the overall data consistency is preserved.

In [17], a generic energy cost model was proposed which gives for each different architectural style (e.g., *client-server*, *Peer2Peer, Publish-Subscribe*) its corresponding energy cost model. Inspired by [17], we propose a model to compute the energy cost of read and write operations of replica control protocols based on quorum systems.

III. Energy Cost Model

In order to compute the *energy cost* of read and write operations of the replica control protocols, we propose a general *quorum-based* model. In this section, we give only the relevant equations, whereas the interested readers can refer to [4] for further details.

A. Client-side Energy Cost

In this section, we give the energy costs of the client-side *component* as well as *connector* induced when performing read and write operations of a replica control protocol.

1) Energy Cost of the Component: Is modeled by summing the energy costs due to (1) executing some algorithm (E_{comp_logic}), (2) sending a read or write operation request to the connector (E_{toConn}) and (3) receiving its response from the connector ($E_{fromConn}$):

$$EC_{client}(comp) = E_{comp\_logic} + E_{toConn} + E_{fromConn} \quad (1)$$

2) Energy Cost of the Connector: Is modeled by summing the energy costs due to (1) receiving an operation request from the component ($E_{fromComp}$), (2) sending this request to all the members of a given quorum, (3) receiving the responses from all the members of the quorum, and (4) sending a response to the component (E_{toComp}):

$$EC_{client}(conn) = E_{conn\_logic} + E_{comm} \quad (2)$$

where E_{conn_logic} represents the energy cost of services that a connector can provide, whereas E_{comm} represents the energy cost of exchanging data both locally and remotely such that:

$$\begin{aligned} E_{conn\_logic} &= E_{conversion} + E_{facilitation} \\ E_{comm} &= E_{commWithComp} + E_{remoteComm} \end{aligned}$$

Furthermore, the parameters of the above two equations are given by:

$$\begin{aligned} E_{conversion} &= (qSize) \times (E_{marshal} + E_{unmarshal}) \\ E_{facilitation} &= qSize \times E_{remoteConnect} \\ E_{commWithComp} &= E_{fromComp} + E_{toComp} \\ E_{remoteComm} &= (qSize) \times [\ (tSize \times tEC + tS) \\ &\quad + (rSize \times rEC + rS)\ ] \end{aligned}$$

such that $qSize$ denotes the size of the read or write quorum, $tSize$ ($rSize$) represents the size of the transmitted request (response), tEC (rEC) denotes the energy cost of transmitting a request (response), and tS (rS) represents constant energy overhead associated with channel acquisition. On the other hand, $E_{fromComp}$ and E_{toComp} represent the energy costs due to receiving a read or write operation request and sending its response to the component. Finally, $E_{remoteConnect}$ denotes the energy cost of creating and managing remote connections.

It is worthwhile to note that, if the component and connector are implemented as a single process, then E_{toConn}, $E_{fromConn}$, $E_{fromComp}$ and E_{toComp} have a value of zero.

B. Replica-side Energy Cost

In this section, we give respectively the energy costs of the replica-side *connector*, *component* as well as *storage medium*.

1) Energy Cost of the Connector: Is modeled by summing the energy costs due to (1) receiving an operation request from the client-side connector, (2) sending this request to the component (E_{toComp}), (3) receiving the response from the component ($E_{fromComp}$), and (4) sending a response to the client-side connector:

$$EC_{replica}(conn) = E_{conn\_logic} + E_{comm} \quad (3)$$

where E_{conn_logic} and E_{comm} have the same definitions as in (2) such that:

$$\begin{aligned} E_{conn\_logic} &= E_{conversion} + E_{facilitation} \\ E_{comm} &= E_{commWithComp} + E_{remoteComm} \end{aligned}$$

Furthermore, the parameters of the above two equations are given by:

$$\begin{aligned} E_{conversion} &= E_{marshal} + E_{unmarshal} \\ E_{facilitation} &= E_{remoteConnect} \\ E_{commWithComp} &= E_{fromComp} + E_{toComp} \\ E_{remoteComm} &= [\ (tSize \times tEC + tS) \\ &\quad +(rSize \times rEC + rS)\ ] + E_{buffer} \end{aligned}$$

where E_{buffer} represents the energy cost of buffering the read or write request whereas the remaining other parameters have the same definitions as above.

2) Energy Cost of the Component: Is modeled by summing the energy costs due to (1) receiving an operation request from the connector ($E_{fromConn}$), (2) executing some algorithm (E_{comp_logic}), (3) sending a read or write operation request to the storage medium (E_{toDisk}), (4) receiving its response from the storage medium ($E_{fromDisk}$) and (5) sending a response to the connector (E_{toConn}):

$$\begin{aligned} EC_{replica}(comp) &= E_{fromConn} + E_{comp\_logic} + E_{toDisc} \\ &\quad +E_{fromDisc} + E_{toConn} \end{aligned} \quad (4)$$

Finally, we use $EC_{replica}$ to denote:

$$EC_{replica} = EC_{replica}(comp) + EC_{replica}(conn) \quad (5)$$

3) Energy Cost of the Storage Medium: Is modeled by summing the energy costs due to (1) receiving an operation request from the component ($E_{fromComp}$), (2) executing the desired operation (reading from the disk or writing to the disk), and (3) sending a response to the component (E_{toComp}). Since writing to the disk might have different energy cost than reading from the disk, then the above energy costs are represented by the following two equations:

$$EC_{read}(medium) = E_{fromComp} + Size_{rd} \times EC_{rd} + E_{toComp} \quad (6)$$

$$EC_{write}(medium) = E_{fromComp} + Size_{wt} \times EC_{wt} + E_{toComp} \quad (7)$$

where $Size_{rd}$ ($Size_{wt}$) and EC_{rd} (EC_{wt}) denote respectively the size of read (write) operation and the energy cost of accessing the storage medium to perform a read (write) operation.

C. General Quorum-based Energy Costs

The energy costs are modelled by summing (1) the energy costs of the client-side component and connector, and (2) the energy costs of the replica-side component, connector and storage medium.

1) Read Operation: The overall energy cost of executing a read operation on a read quorum of size $rqSize$ is given by:

$$\begin{aligned} EC_{Read} &= rqSize \times [EC_{replica} + EC_{read}(medium) + \\ &\quad E_{marshal} + E_{unmarshal} + E_{remoteConnect} + \\ &\quad (tSize \times tEC + tS) + (rSize \times rEC + rS)] \\ &\quad +EC_{client}(comp) + E_{commWithComp} \end{aligned} \quad (8)$$

We rewrite the above equation in the following form:

$$EC_{Read} = rqSize \times k_r + c \quad (9)$$

where k_r and c represent energy related constants.

2) Write Operation: The overall energy cost of executing a write operation on a write quorum of size $wqSize$ is:

$$\begin{aligned} EC_{Write} &= wqSize \times [EC_{replica} + EC_{write}(medium) \\ &\quad +E_{marshal} + E_{unmarshal} + E_{remoteConnect} \\ &\quad +(tSize \times tEC + tS) + (rSize \times rEC \\ &\quad +rS)] + EC_{client}(comp) + \\ &\quad E_{commWithComp} \end{aligned} \quad (10)$$

We rewrite the above equation in the following form:

$$EC_{Write} = wqSize \times k_w + c \quad (11)$$

where k_w and c represent energy related constants.

D. Overview

As we can notice from equations (9) and (11) that the *quorum sizes* ($rqSize$ or $wqSize$) play a major role in the overall energy cost computation of replication-based systems.

IV. The Arbitrary Tree protocol

The *Arbitrary Tree* protocol was proposed by [3], which assumes that replicas of the system are organized logically into *any tree* structure where its nodes can be either *logical* or *physical*. Unlike a physical node which corresponds to a replica of the system, a logical node is used only to preserve the tree structure. Two new notions of *physical and logical levels* were introduced such that a physical level consists of at least one physical node whereas a logical level has all of its nodes logical. Basically, a write quorum is composed of all physical nodes of a single physical level of the tree whereas a read quorum consists of any single physical node of every physical level of the tree.

A. The Proposed Algorithm

In order to obtain satisfactory results both for read and write operations in terms of the quorum size, availability and load induced on the system, the proposed Algorithm 1 of [3] constructs the tree structure in the following manner:

- Sets the root of the tree to be logical.
- Sets the number of physical levels as well as the height of the tree to be $\sqrt{n}$.
- Arranges 4 physical nodes (replicas) at the 1st seven physical levels of the tree.
- Arranges $\frac{n-28}{\sqrt{n}-7}$ physical nodes (replicas) at every remaining physical level of the tree by obeying the assumption 3.1 of [3].

V. Energy-efficient Approach

A. Motivation

In contrast to previous replication protocols, in this paper, we suggest to compute the *energy cost* of the read and write operations of replica control protocols using the model of section III. Moreover, we make use of the protocol defined in [3] (see section IV) in order to study the case of the proposed *Algorithm 1* (see section IV-A) by taking into account the system's *energy consumption*.

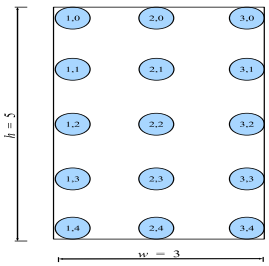
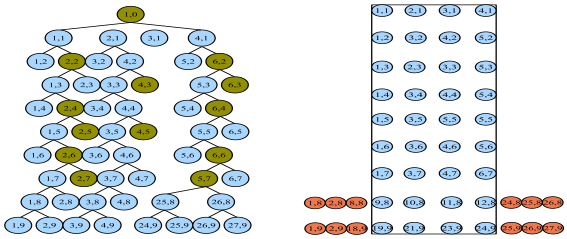


Fig. 2. An example illustrated by the pair (replica number,level) of the rectangle structure for $n = 15$ replicas.



(a) Tree (b) Rectangle

Fig. 3. Transformation of the tree structure into a rectangle structure.

In order to *reduce* the energy consumption of Algorithm 1 of [3], we propose to *turn off* a significant amount of replicas of the system and to *logically reorganize* the other replicas into a new logical rectangle structure. Note that, the logical transformation from tree structure into rectangle does not necessitate the reconfiguration of the protocol of [3] and ensures overall *data consistency* among replicas of the system.

B. Rectangle Structure

We propose to *rearrange* the replicas, which are organized logically into an arbitrary tree structure in [3], into a rectangle structure of height $h > 1$ and width $w > 1$.

Given a set of replicas organized logically into a rectangle structure of height $h > 1$ and width $w > 1$, the read operation of this structure using protocol of [3] has:

$$\begin{aligned}
\text{A quorum size of } RD_{cost} &= h \\
\text{An availability of } RD_{av}\ (p) &= \prod_{i=0}^{h-1}(1-(1-p)^w) \\
\text{An optimal system load of } \mathcal{L}_{RD} &= \frac{1}{w} \\
\text{An energy cost of } EC_{Read} &= h \times k_r + c
\end{aligned}$$

On the other hand, the write operation of such a structure using protocol of [3] has:

$$\begin{aligned}
\text{A quorum size of } WR_{cost} &= w \\
\text{An availability of } WR_{av}\ (p) &= 1 - \prod_{i=0}^{h-1}(1-p^w) \\
\text{An optimal system load of } \mathcal{L}_{WR} &= \frac{1}{h} \\
\text{An energy cost of } EC_{Write} &= w \times k_w + c
\end{aligned}$$

It is important to note that the availability computations are carried out by assuming that every replica is independently available with a probability $p > 0.5$. Also, for the optimal load computation of the operations, interested readers can refer to the *Appendix* of [3].

Figure 2 illustrates an example of a rectangle structure composed of 15 replicas where each replica is denoted by the pair (replica number,level).

C. Transformation

Since the original algorithm proposes to set 4 replicas at the first seven (physical) levels of the tree (see section IV-A), then we set the width of the rectangle to have a value of $w = 4$. Moreover, the height of the rectangle is set to have a value of $h = \sqrt{n}$, where n denotes the number of replicas of the initial tree structure. By setting the values of w and h as mentioned above to construct the rectangle structure, we propose to switch off any $(\frac{n-28}{\sqrt{n}-7} - 4)$ replicas at every physical level (other than the first seven ones) of the tree structure constructed using Algorithm 1 of [3].

Figure 5 illustrates an example of transforming a system composed of $n = 81$ replicas, from a tree structure of Algorithm 1 of [3] into the rectangle one. Note that not all replicas are demonstrated in both subfigures and that only the blue circles represent replicas of the system which are denoted by the pair (replica number,level). In Figure 3(b), the red circles represent the replicas that have been switched off from the original tree structure of Figure 3(a).

We can notice from Tables 1 and 2 that we preserved comparable characteristics in terms of quorum size, availability and system load of read and write operations, while *reducing the energy cost* of write operations as well as *saving the energy consumption of* 45 machines.

TABLE I
READ OPERATION OF 81 REPLICAS

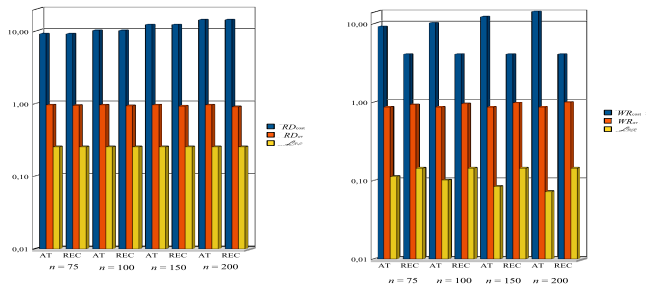
	Tree	Rectangle
RD_{cost}	9	9
RD_{av} (0.7)	0.94	0.92
$\mathcal{L}_{RD}$	0.25	0.25
$EC_{Read}(min)$	$9k_r + c$	$9k_r + c$
$EC_{Read}(avg)$	$9k_r + c$	$9k_r + c$
$EC_{Read}(max)$	$9k_r + c$	$9k_r + c$

VI. COMPARISON

In this section, we compare the *tree structure* constructed using Algorithm 1 of [3] with the proposed *rectangle structure* by taking into account each one's quorum size, availability, system load as well as the energy cost of read and write operations. For this purpose, we set up several configurations

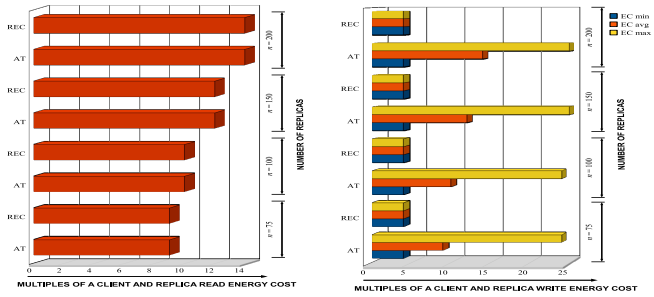
TABLE II
WRITE OPERATION OF 81 REPLICAS

	Tree	Rectangle
WR_{cost}	9	4
WR_{av} (0.7)	0.85	0.915
$\mathcal{L}_{WR}$	0.111	0.142
$EC_{Write}(min)$	$4k_w + c$	$4k_w + c$
$EC_{Write}(avg)$	$9k_w + c$	$4k_w + c$
$EC_{Write}(max)$	$27k_w + c$	$4k_w + c$



(a) Read (b) Write

Fig. 4. Quorum size (RD_{cost}), availability (RD_{av}) and system load ($\mathcal{L}_{RD}$) of Tree (AT) and Rectangle (REC) structures.



(a) Read (b) Write

Fig. 5. Energy cost of Tree (AT) and Rectangle (REC) structures

composed of $n = 75$, 100, 150 and 200 replicas respectively. For the availability computations, we suppose that every replica is independently available with a probability $p = 0.7$.

A. Read Operation

Figure 4(a) illustrates the quorum size (RD_{cost}), availability (RD_{av}) and system load ($\mathcal{L}_{RD}$) of read operations of both *tree* (AT) and *rectangle* (REC) structures. Figure 5(a) gives the energy cost of the read operations for both structures in terms of multiples of a single client and replica read energy cost. We can notice in these two figures that both structures have quite identical characteristics in terms of the quorum size, availability, system load and energy cost. As we can see in Table III that, with the rectangle structure, we achieve the same characteristics of the tree structure while *using fewer number of replicas* (reduced overall energy consumption).

B. Write Operation

Figure 4(b) illustrates the quorum size (WR_{cost}), availability (WR_{av}) and system load ($\mathcal{L}_{WR}$) of write operations of both *tree* (AT) and *rectangle* (REC) structures. We can notice that the rectangle (REC) structure has smaller quorum sizes than the tree (AT) one due to the fact that at every level there are always 4 replicas. On the other hand, both structures have quite comparable availability for their write operations. Finally, we can notice that, the tree structure has a smaller system load ($\frac{1}{\sqrt{n}}$) than rectangle (always $\frac{1}{7}$) and that such a load diminishes as the number of replicas of the system becomes larger. Note that, we can sacrifice load which is at the worst case around 14% higher than the tree structure, however

TABLE III
NUMBER OF MACHINES SAVED FROM ENERGY CONSUMPTION.

Number of replicas	Tree	Rectangle
$n = 75$	0	39
$n = 100$	0	60
$n = 150$	0	102
$n = 200$	0	144

as we will see in Figure 5(b), with much *less energy cost* and *reduced overall energy consumption*.

Figure 5(b) gives the energy cost of the write operations for both structures in terms of multiples of a single client and replica write energy cost. The write operation of the rectangle structure has a much fewer *energy cost* than that of the tree structure. Furthermore, we can observe that, in the worst case, the write operation of the tree structure has an energy cost more than 5 times greater than that of the rectangle structure.

C. Overview

Table III indicates the number of turned off machines for the corresponding number of replicas. By a simple logical structural transformation from tree to rectangle, we are capable of achieving the same characteristics of the tree structure of Algorithm 1 of [3], yet with much reduced *energy cost* as well as reduced overall *energy consumption*.

VII. CONCLUSION

In this paper, we suggested to compute the energy cost of operations of replica control protocols by proposing a new *quorum-based energy cost* model. Also, we introduced an approach to reduce the energy cost as well as consumption of the tree-structured protocol of [3]. We showed that by shutting down a significant amount of replicas and by performing a logical structural transformation, our approach has fewer energy cost for its write operations than that of [3] while conserving its major characteristics. Also, the read operations of our approach has the same energy cost as that of [3], however with much reduced overall energy consumption. It is important to note that our proposal does not require the reconfiguration of the protocol whenever energy efficiency requirements change in the system.

REFERENCES

[1] D. Agrawal and A. E. Abbadi. The tree quorum protocol: An efficient approach for managing replicated data. In *Proceedings of the sixteenth international conference on Very Large Databases*, pages 243–254, 1990.

[2] D. Agrawal and A. E. Abbadi. An efficient and fault-tolerant solution for distributed mutual exclusion. *ACM Transactions on Computer Systems*, 9(1):1–20, 1991.

[3] J. P. Bahsoun, R. Basmadjian, and R. Guerraoui. An arbitrary tree structured replica control protocol. *The 28th International Conference on Distributed Computing Systems*, pages 502–511, 2008.

[4] R. Basmadjian and H. D. Meer. An approach to reduce the energy cost of the arbitrary tree replication protocol. *Proceedings of the ACM SIGCOMM 1st Int'l Conf. On Energy-Efficient Computing and Networking*, 2010.

[5] P. Bernstein and N. Goodman. An algorithm for concurrency control and recovery in replicated distributed databases. *ACM Transactions on Database Systems*, 9(4):596–615, 1984.

[6] J. S. Chase, D. C. Anderson, P. N. Thakar, A. M. Vahdat, and R. P. Doyle. Managing energy and server resources in hosting centers. *ACM SIGOPS Operating Systems Review*, 35(5):103–116, 2001.

[7] S. Y. Cheung, M. H. Ammar, and M. Ahamad. The grid protocol: A high performance scheme for maintaining replicated data. *IEEE Transactions on Knowledge and Data Engineering*, 4(6):438–445, 1990.

[8] S. C. Choi, H. Y. Youn, and J. S. Choi. Symmetric tree replication protocol for efficient distributed storage system. *International Conference on Computational Science*, pages 474–484, 2003.

[9] T. Heath, A. P. Centeno, P. George, L. E. S. Ramos, Y. Jaluria, and R. G. Bianchini. Mercury and freon: temperature emulation and management for server systems. In *Proceedings of the 12th international conference on Architectural support for programming languages and operating systems*, pages 106–116, 2006.

[10] K. H. Kim, R. Buyya, and J. Kim. Power aware scheduling of bag-of-tasks applications with deadline constraints on dvs-enabled clusters. In *Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid*, pages 541–548, 2007.

[11] H. Koch. An efficient replication protocol exploiting logical tree structures. *The 23rd Annual International Symposium on Fault-Tolerant Computing*, pages 382–391, 1993.

[12] A. Kumar. Hierarchical quorum consensus: A new algorithm for managing replicated data. *IEEE Transactions on Computers*, 40(9):996–1004, 1991.

[13] M. Maekawa. A $\sqrt{n}$ algorithm for mutual exclusion in decentralized systems. *ACM Transactions on Computer Systems*, 3(2):145–159, May 1985.

[14] M. Naor and A. Wool. The load, capacity, and availability of quorum systems. *SIAM Journal on Computing*, 27:214–225, 1998.

[15] D. Peleg and A. Wool. The availability of quorum systems. *Information and Computation*, 123(2):210–223, 1995.

[16] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath. Dynamic cluster reconfiguration for power and performance. In *Compilers and operating systems for low power*, pages 75–93, 2003.

[17] C. Seo, G. Edwards, D. Popescu, S. Malek, and N. Medvidovic. A framework for estimating the energy consumption induced by a distributed system's architectural style. In *Proceedings of the 8th international workshop on Specification and verification of component-based systems*, 2009.

[18] A. Silberschartz, H. F. Korth, and S. Sudarshan. *Database System Concepts*. McGraw-Hill, fourth edition, 2002.

[19] R. H. Thomas. A majority consensus approach to concurrency control for multiple copy databases. *ACM Transactions on Database Systems*, 4(2):180–209, June 1979.

[20] N. Vasic, M. Barisits, V. Salzgeber, and D. M. Kostic. Making cluster applications energy-aware. In *Proceedings of the 1st workshop on Automated control for data centers and clouds*, pages 37–42, 2009.

Robert Basmadjian received his Ph.D. degree from University of Toulouse, France, in 2008. Since then he has been a research associate at the chair of Computer Networks and Computer Communications in University of Passau. His main research interests include data replication and energy efficiency. He is currently involved in the EU FP7 project FIT4Green and is a member of the EuroNF Network of Excellence.

Hermann De Meer Prof. Dr. Hermann de Meer received his Ph.D. in 1992 on the topic "Transiente Leistungsbewertung und Optimierung rekonfigurierbarer fehlertoleranter Rechensysteme". He had been an Assistant Professor at Hamburg University, Germany, a Visiting Professor at Columbia University in New York City, USA, and a Reader at University College London, UK. He is currently appointed as Full Professor at the University of Passau, Germany, and as Honorary Professor at University College London, UK. He is director of the Institute of IT Security and Security Law (ISL) at the University of Passau. His main research interests include IT security and resilience, virtualization and energy efficiency, complex and self-organizing systems, peer-to-peer systems, quality of service and performance modeling, Internet protocols, home networking, and mobile computing. Hermann de Meer has led several nationally and internationally funded projects on Performance Modeling and Computer Networking. He currently holds several research grants funded by the Deutsche Forschungsgemeinschaft (DFG) and by the EU (FP6 and FP7). Prof. H. de Meer is co-authoring a textbook on Queueing Networks and Markov Chains - - Modeling and Performance Evaluation with Computer Science Applications, published by John Wiley in 1998 and 2006.

Energy-Efficient Office Environments

Andreas Berl
University of Passau
Passau, Germany
Email: berl@uni-passau.de

Hermann de Meer
University of Passau
Passau, Germany
Email: demeer@uni-passau.de

***Abstract*—The rising costs of energy and the world-wide desire to reduce CO_2 emissions has led to an increased concern over the energy efficiency of information and communication technology. Whilst much of this concern has focused on data centres, also office hosts that are located outside of data centres (e.g., in public administration or companies) have been identified as significant consumers of energy. Office environments offer great potential for energy savings, given that computing equipment often remains powered for 24 hours per day, and for a large part of this period is underutilised or even idle. This paper investigates the energy consumption of hosts in office environments, discusses the potential of energy savings and proposes an energy-efficient office management approach based on resource virtualization, power management, and service consolidation. Different virtualization techniques are used to enable management and consolidation of office resources. Idle services are stopped from consuming resources on the one hand and (underutilized) services are consolidated on a smaller number of hosts on the other hand.**

I. Introduction

Energy efficiency of information and communication technology has become an important topic in companies and public administration – the bottleneck of costs has changed. While hardware costs are decreasing on the one hand, costs of energy are increasing on the other hand. In addition, there are world wide efforts to turn IT green, (e.g., CO_2 emissions need to be reduced). Data centres are well known and often discussed consumers of energy. Koomey [1] reports a doubling of energy consumption from 2000 to 2005 of volume, mid-range, and high-end servers in the U.S. and worldwide. The power used by data centres and computer networks [2] runs in the billions of euros. Although this is mostly related to data centers, a similar tendency can be expected for computers outside of data centres. End devices are contributing to a large portion of the electricity consumption growth according to a 2006 survey commissioned by the EU [3].

Office hosts that are located outside of data centres contribute significantly to the overall IT energy consumption, simply because of the high number of such devices – in offices usually each employee has his own host. Office hosts, however, are often underutilized (in terms of CPU load) or not used at all (while being switched on). There are short term periods where hosts remain turned on without being used, e.g, if users are in meetings, do telephone calls, have lunch or coffee breaks. Additionally, office hosts often remain turned on on a 24/7 basis. Such hosts are running due to several reasons: Jobs might be scheduled over the night (e.g., security updates, or backups). Hosts are also often left switched on, because users require access to them remotely. Remote access typically happens from the users home or when users are working externally (e.g., at a customers office). Remote access is needed in such cases to access applications and data within the office environment. The user finds his working environment exactly in the same state in which he has left it, even the cursor in an opened text document is in the same spot as before. The user may need access to email accounts, personal data, or applications (e.g., special software with access to databases that is not available outside of the office). Apart from such reasons, some users simply forget turn off their hosts, when they leave the office.

Webber et al. [4] have analyzed sixteen sites in the USA and reported that 64% of all investigated office hosts were running during nights. Furthermore, even when office hosts are in use, they are often underutilized by typical office applications, e.g., mail clients, browsers or word processing. It is important to see that idle hosts (CPU usage of 0%) and underutilized hosts consume a considerable amount of energy, compared to computers that are turned off, without providing any added benefit. Measurements that have been performed at the University of Sheffield on hosts that are typically used as personal computers [5] show that idle office hosts still consume 49% to 78% of the energy that they need when they are intensely used.

Several approaches have been suggested that deal with high energy consumptions of hosts in office environments (see Section V). Such solutions range from the enforcement of office-wide power-management policies to thin-client approaches, where users share resources on terminal servers. As an extension to power-management solutions and opposed to data-centre based terminal-server approaches, this paper suggests a combination of office-wide power management with the consolidation of services in office environments. The key technology for this approach is the virtualization of services. The office environment is virtualized, based on system virtualization peer-to-peer approaches to enable resource sharing. The number of simultaneously running hosts in the office environment is reduced, while the utilization of hosts is raised. This enables a major reduction of the overall energy consumption within the office, without significantly decreasing quality or quantity of provided services.

II. A Managed Office Environment

When a user powers on his host in a common office, he finds his usual working environment. Within this paper this working environment is referred to as a *personal desktop environment (PDE)*. This typically consists of an operating system, applications, and the user's personal data and configurations. Although, in common offices often roaming profiles are available (see Section III), the PDE as a whole is fixed, i.e., it is bound to a certain host in the office. When the PDE is turned on/off, also the host is turned on/off and vice versa. Users are able to access their PDE locally within the office or they may also be able to access it remotely from outside the office.

In the managed office environment, PDEs are additionally used as *mobile services*. Mobile services are freely movable within the office environment (between physical hosts) and are used to achieve service consolidation. When the user is not physically using his office host, his PDE can be decoupled from the host and be migrated to another host for energy reasons. Several PDEs can be provided by a single host. Therefore, a user's host is not necessarily turned on when a user remotely utilizes his PDE – the PDE may be provided by a different host.

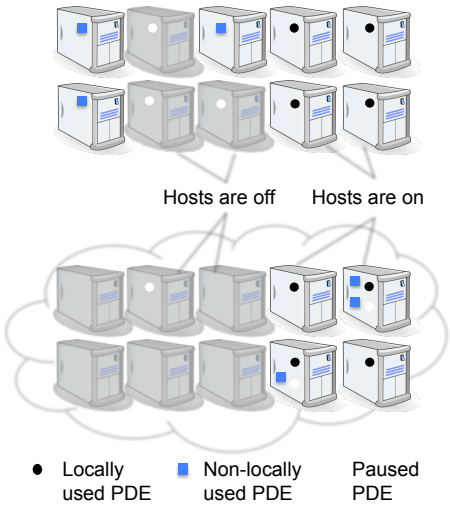


Fig. 1. Common and managed office environment

In Figure 1 the transition from a common to a managed office environment (based on PDEs) is illustrated. It can be observed in the upper part of the figure that in the common office environment the PDEs and the hosts are interdependent. Seven hosts are turned on together with seven PDEs and three hosts (with PDEs) are turned off. The situation is very different in the managed office environment shown in the lower part of the figure. Although the number of currently running PDEs is the same as before, only four hosts are actually turned on. It can be observed, e.g., that the upper right host is providing three PDEs to users simultaneously.

Based on the availability of mobile PDEs, energy efficiency is achieved in three steps:

1) Unloaded PDEs in the office environment are stopped from consuming resources. If a PDE is idle (no job is performed on behalf of its user) it will be suspended.
2) Loaded PDEs are consolidated on a small number of hosts. If a PDE is not accessed locally (the user does not physically access his office host), the PDE becomes a mobile service and may be migrated to other hosts to achieve consolidation.
3) Hosts that do not provide running PDEs are shut down to save energy.

The managed office environment has to dynamically determine an energy-efficient mapping of PDEs to hosts in the office and initiates necessary migrations of PDEs. This mapping has to fulfill contradicting goals and needs to solve a twodimensional optimization problem:

- The mapping needs to constantly maintain a **valid** configuration in the office environment to provide PDEs to users as needed. A mapping is called valid, if 1) all PDEs are located at their dedicated hosts, and 2) no host is overloaded with PDEs. Valid mappings allow all users to access their PDEs as desired, but are not necessarily optimized considering energy efficiency.
- The mapping needs to achieve energy-efficiency through consolidation, by approaching a **host optimal** configuration. A mapping is called host optimal, when it utilizes the minimum possible number of hosts to provide all required PDEs (locally or remotely) in the office.
- The mapping needs to minimize the number of migrations within the office environment because migrations are costly themselves (in terms of network traffic and interference with the users work). Unnecessary migrations need to be avoided and hosts should not be overloaded by performing several migrations simultaneously.

The architecture of the managed office environment is further described and evaluated in [6].

III. Virtualization Approach

An important virtualization approach that is used in the managed office environment is system virtualization. It enables service consolidation and is successfully applied to data centres today. It can be adapted to office environments in order to achieve a similar utilization and energy efficiency of office resources. In system virtualization *virtual machines (VMs)* are created from idle resources. Full hosts are virtualized, consisting of virtual CPUs, virtual memory, virtual hard disk, virtual network interface card, etc. A VM is an imitation of a real machine in such a way that an operating system can be installed on it without being aware of the resource virtualization. The software that provides VMs is usually called *Virtual Machine Monitor (VMM)* (e.g., VMWare Server[1], QEMU [7], or Xen [8]) and is able to process several VMs simultaneously on a single host. There are several basic primitives of management functions available for VMs: create, destroy, start, stop, migrate, copy, pause, and resume VM. It

[1] http://www.vmware.com/de/products/server

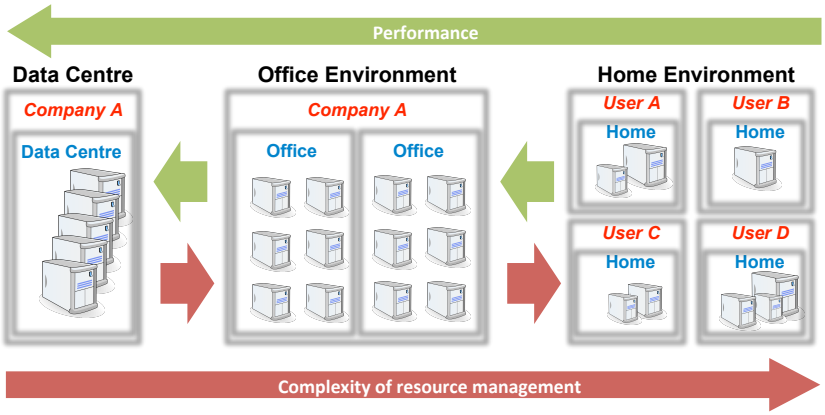


Fig. 2. Performance and complexity of resource management

is even possible to have a *live migration* [9]. This means that a service in a VM can be migrated to another host without being interrupted. A PDE, as it is described in Section II, can be encapsulated within a VM and inherits all of the VM-related features. This enables the operation of PDEs in separated runtime environments (VMs). The VMM can trigger the shut down of a host if required. Hosts can be powered up again, e.g., by using wake on LAN mechanisms[2], to boot into the VMM again. PDEs can be suspended by the VMM if they are idle and be resumed again if necessary. Additionally, when PDEs are enclosed in VMs they can be migrated from host to host, without a durable interruption of running services.

A second important virtualization approach that is needed to realize the managed office environment is based on P2P technology. Independent of the logical network that is used to interconnect hosts, the resource sharing in the managed office environment is done in a P2P manner. There is no central element that provides resources to run PDEs on, as it is available in the thin-client/terminal-server approach. Instead all of the office hosts are sharing their resources. Therefore, methods and principles from P2P overlays can be used to realize a management environment that interconnects hosts and provides mediation for hosts and PDEs. P2P content distribution networks (e.g., eDonkey[3] or BitTorrent[4]) are often used to share files among users. Such protocols provide several functions, the behaviour of which can be adapted to office environments. First, these kind of networks create and maintain an overlay network among participants that enables a logical addressing of hosts, users, and content. Second, they enable the mediation of resources and are able to bring providers and consumers of content together. Third, such networks additionally manage the access to resources, in order to achieve an optimal and fair distribution of resources among all users of the network. Concerning managed office environments, P2P overlays enable interconnection, addressing, and mediation of PDEs and hosts within the office environment. They also enable a management of PDEs and hosts based on their current states (e.g, powering off/on hosts or PDEs).

[2]http://www.energystar.gov/index.cfm?c=power _mgt.pr_power_mgt_wol
[3]http://www.overnet.org
[4]http://www.bittorrent.com

IV. Service Consolidation Obstacles

Energy-efficient consolidation of services is only achieved in data centres, today. The main reason for this is that data centres differ significantly from the other environments in terms of provided performance and the complexity of resource management. The more performant, centralized, homogeneous, and controllable an environment is, the easier service consolidation can be applied. Figure 2 illustrates the three different environments. It can be observed, that in data centres server hosts are located very close to each other, usually within a single room, and are interconnected with a high performance network. In office environments hosts are loosely coupled, distributed over several rooms and typically connected via Fast Ethernet. The home environment consists of heterogeneous and rather small networks, interconnected via asynchronous DSL connections.

Service consolidation (as it is done in data centres) can not easily be adapted to office environments. Server hosts tend to be more performant than office hosts, in terms of CPU cycles, memory capacity, and networking. This enables servers in data centres to run several virtualized services (up to hundreds) simultaneously, depending on the number of users that are using the services. Office hosts, in contrast, may run only a few virtualized services simultaneously. The high performance network in the data centre allows a fast migration of virtualized services from one host to another. Migration, however, is a problem in the office environment. Whereas in data centres usually only processes are migrated (operating system and data are typically stored on network storage), PDEs have to be migrated entirely. This leads to considerable overhead because operating system and user data and applications might sum up to several GBs of data. This issue is further discussed in [10].

Additionally resource management is less complex in data centres, compared to office environments. Whereas the data centre is a controlled environment, where only administrators have physical access to hosts, the office environment is rather uncontrolled. Users are able to power hosts on and off, unplug

cables, or move hosts to other locations. Furthermore, in data centres users use their services remotely via network access, which eases up the consolidation of services. Local access to hosts, as it is typical in office environments adds hard constraints to the management of resources: Services that are used locally can not be migrated or consolidated. The physical access of users to host in the office environment additionally raises security issues. When services are migrated to achieve consolidation, employees are potentially able to copy or modify contents of other persons.

It is even more difficult to approach consolidation of services in home environments [13], as they provide less performance and more resource management complexity. In office environments typically similar kinds of hosts, operating systems, and applications are used, whereas hardware and software is heterogeneous in home networks, applications and equipments depend on the flavour of the different home users. Office environment hosts are typically interconnected via Fast Ethernet, providing a symmetric up and download behaviour, whereas in home networks usually DSL-connections are applied (with different performance properties for different homes), often providing smaller upload than download performance. Data storage is realized in a completely decentralized way, there is usually no shared storage resource available between different home users. In office environments that belong to the same company there will at least be a minimum of trust among employees. In the home environment instead, services are migrated between completely unrelated users. A major obstacle that complicates service consolidation in home environments is the cost of energy. In data centres and office environments the energy is payed by a single company (probably on different expenditures). However, when resources are shared for consolidation among home users, some users will receive a higher energy bill than others – without having consumed more resources. This has to be balanced by the service management.

V. Related Work

There are several projects that provide power-management solutions for office environments. Examples are eiPowerSaver[5], Adaptiva Companion[6], FaronicsCore[7], KBOX[8], or LANrev [9]. In such approaches, office-wide power management policies are applied to office environments. Office hosts change to low-power modes, independent of user-specific power management configurations. Additionally, mechanisms are provided to wake up hosts if necessary. This way, hibernated hosts can be used for overnight jobs (e.g., backup processes) and for remote usage. Such solutions, however, rely on the capability of the host to switch to low-power modes which depends on the complex interaction of a host's hard and software. The approach presented in this paper is independent of such interaction. PDEs are suspended together with their VM without being aware of the suspension. What is more, the mentioned power-management solutions focus on idle hosts only. The solution suggested in this paper, additionally deals with the energy consumption of underutilized hosts in office environments.

Thin-client/terminal-server approaches use data-centre technology to provide energy-efficient services in office environments. User environments (similar to PDEs) are provided by terminal servers and users can access these environments via energy-efficient thin clients. Common terminal-server software products are Citrix XenApp[10], Microsoft Windows Server 2008[11], or the Linux Terminal Server Project[12]. Similar to the approach suggested in this paper, such approaches foster a resource sharing among users in the office environment. However, this approach is based of the usage of additional hardware in the office (energy-efficient thin clients and terminal servers) and PDEs are provided in a centralized way by the terminal server. Instead, the approach suggested in this paper utilizes available hosts in office environments and shares resources among them.

In [12], [13], [14] a virtualized future home environment is introduced that uses virtualization to aggregate and consolidate distributed hardware resources of home users in order to save energy. Similar to offices, also in home environments some machines are running on a 24/7 basis (e.g., media servers or P2P clients). These services can be consolidated by using different virtualization techniques in order to turn unused hosts off. In contrast to the future home environment approach, this work focuses on resource sharing in office environments as they can be found today in companies or public administration. Whereas in the future home environment separate services are virtualized (e.g., video-encoding or P2P file-sharing services) and are distributed among homes, this work suggests to virtualize user environments (PDEs) as a whole. As an important consequence, the approach in this paper envisions a seamless and transparent provision of user services within the PDE (e.g., when a PDE has been moved, the user still finds his text document open, with the cursor at the same position as before the migration). The future home environment approach, in contrast, is not transparent to the user. The user has to utilize special software that enables the envisioned migrations of services, and seamless access to migrated services is not possible. Instead the result of a service is transferred back to the user.

VI. Conclusions and Future Work

This paper has presented an architecture that manages resources in office environments in energy efficient ways. A shift from current decentralized resource management approaches (per user) is suggested to a centralized resource management approach (per office). The proposed solution extends available power-management approaches and is opposed to data-

[5] http://entisp.com/pages/eiPowerSaver.php
[6] http://www.adaptiva.com/products_companion.html
[7] http://faronics.com/html/CoreConsole.asp
[8] http://www.kace.com/solutions/power-management.php
[9] http://www.lanrev.com/solutions/power-management.html
[10] http://www.citrix.com/XenApp
[11] http://www.microsoft.com/windowsserver2008
[12] http://www.ltsp.org

center based thin-client/terminal-server solutions. It exploits available potentials of energy savings in office environments by managing office resources based on the behaviour of users. Resource virtualization technologies (system virtualization and peer-to-peer overlays) are used to suspend idle services and to consolidate underutilized services on a small number of hosts. The suggested architecture is evaluated in [10] and [6] and it is shown that that more than 70% of energy savings can be achieved in office environments, without significantly interrupting the day to day work of users.

In future work, the suggested architecture will be refined, together with an energy consumption model for office environments, based on discrete event simulation.

ACKNOWLEDGMENT

The research leading to these results has received funding from the German Federal Government BMBF in the context of the G-Lab_Ener-G project and from the European Community's FP7 in the context of the EuroNF Network of Excellence (grant agreement no. 216366).

REFERENCES

[1] J. Koomey, "Estimating total power consumption by servers in the US and the world, Technical report," Lawrence Berkeley National Laboratory Stanford University, Tech. Rep., February 2007.

[2] X. Fan, W. Weber, and L. Barroso, "Power provisioning for a warehouse-sized computer," in *Proceedings of the 34th annual international symposium on Computer architecture*. ACM New York, NY, USA, 2007, pp. 13–23.

[3] P. Bertoldi and B. Atanasiu, "Electricity consumption and efficiency trends in the enlarged European Union," *IES–JRC. European Union*, 2007.

[4] C. Webber, J. Roberson, M. McWhinney, R. Brown, M. Pinckard, and J. Busch, "After-hours power status of office equipment in the usa," *Energy-the International Journal*, vol. 31, no. 14, pp. 2487–2502, 2006.

[5] C. Cartledge, "Sheffield ICT Footprint Commentary." *Report for SusteIT. Available at: http://www.susteit.org.uk/files/files/26-Sheffield_ICT_Footprint_Commentary_Final_8.doc*, 2008.

[6] A. Berl and H. de Meer, "A Virtualized Energy-Efficient Office Environment," in *Proceedings of International Conference On Energy-Efficient Computing and Networking (e-Energy 2010), Passau, Germany, April 13-15, 2010*. ACM digital library and IEEE Computer Society Press, April 2010.

[7] F. Bellard, "QEMU, a fast and portable dynamic translator," in *Proceedings of the USENIX Annual Technical Conference, FREENIX Track*, 2005, pp. 41–46.

[8] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," *SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, pp. 164–177, 2003.

[9] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live Migration of Virtual Machines," in *2nd conference on Symposium on Networked Systems Design & Implementation (NSDI'05)*. Berkeley, CA, USA: USENIX Association, 2005, pp. 273–286.

[10] A. Berl and H. de Meer, "An Energy-Efficient Distributed Office Environment," in *Proceedings of European Conference on Universal Multiservice Networks (ECUMN 2009), Sliema, Malta, October 11-16, 2009*. IEEE Computer Society Press, October 2009.

[11] M. Satyanarayanan, B. Gilbert, M. Toups, N. Tolia, D. O'Hallaron, A. Surie, A. Wolbach, J. Harkes, A. Perrig, D. Farber *et al.*, "Pervasive personal computing in an internet suspend/resume system," *IEEE Internet Computing*, pp. 16–25, 2007.

[12] H. Hlavacs, K. A. Hummel, R. Weidlich, A. Houyou, A. Berl, and H. de Meer, "Distributed Energy Efficiency in Future Home Environments," *Annals of Telecommunication: Next Generation Network and Service Management*, vol. 63, no. 9, pp. 473–485, October 2008. [Online]. Available: http://dx.doi.org/10.1007/s12243-008-0045-2

[13] A. Berl, H. de Meer, H. Hlavacs, and T. Treutner, "Virtualization Methods in Future Home Environments," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 62–67, December 2009.

[14] A. Berl, H. Hlavacs, R. Weidlich, M. Schrank, and H. de Meer, "Network Virtualization in Future Home Environments," in *LNCS 5841: IFIP, Proceedings of Int. Workshop on Distributed Systems: Operations and Management (DSOM09), Venice, Italy, October 27-28, 2009*, C. Bartolini and L. Gaspary, Eds. Springer, Berlin (Germany), October 2009, pp. 177–190.