

Towards Green HPC Blueprints

Goran Martinovic, Zdravko Krpic

Faculty of Electrical Engineering
Josip Juraj Strossmayer University of Osijek
Osijek, Croatia

e-mail: goran.martinovic@etfos.hr, zdravko.krpic@etfos.hr

Abstract—Effectiveness and power consumption is becoming a major problem in high-performance computing. Numbers of researchers are working on methodologies in order to increase efficiency of these systems on hardware and software levels. Several “green” technologies are explained in this paper along with their pros and cons, with the aim of improving the power efficiency for high performance computers and cloud computing systems. Many of the aspects of green HPC are still in their initial stages, so this paper analyzes recent contributions in that respect, and proposes related work for every “green” improvement of HPC systems. It gives detailed blueprints for the Green HPC using state-of-the-art technologies from this field of research.

Keywords - green computing; high performance computing; cloud computing; energy efficiency.

I. INTRODUCTION

Throughout the history of High Performance Computing (HPC), raw processing power was a primary concern. Various companies have had tendencies to build bigger computer systems in order to solve demanding computing tasks which could not be done on mainstream computer machines, or at least, in a reasonable time. As computer power grew, so did its “pat on the back” - heat dissipation, power consumption, production costs and software costs. A mere “petaflop race” became too expensive to participate in, although some institutions ignored the phenomenon of power consumption increase, claiming that this is a normal evolution of HPC. The “Green Destiny” Project [1] proved them wrong, and caught a tremendous amount of interest in both the computing and business industries. This revelation lead to different theories, but in the end, most computer scientists agreed that the energy footprint from the computers must be reduced, trying thereby to preserve performance. This being the case, various “Green” standards, such as EPEAT; for details see [2], Energy Star 5.0, [3], and RoHS directive [4], that were established as a guide for HPC equipment manufacturers and Cloud Computing (CC) vendors.

The second important reason for reducing HPC energy consumption is of a financial nature. HPC centers and CC system holders tend to exploit their resources in the most economic way, thus increasing the profit. In the world of

ever-growing HPC systems, CC systems and service computing, the ability to offer more resources imply large expenses for maintenance, cooling and electric bills.

The purpose of this paper is to give a basic insight into available and proposed methods for the “Greening” of HPC and CC systems, as well as their positive and negative impact on performance and energy savings. Every method will be referred to related work.

The rest of the paper is organized as follows: Section 2 introduces possible “green” solutions for HPC and CC systems, while in Section 3, every solution is elaborated and supported by examples from available sources. Section 4 uses tiered HPC design to pinpoint objects for implementing proposed solutions. Finally, Section 5 concludes the paper and announces future research by authors.

II. SOLUTIONS AND RELATED WORK

Hardware manufacturers are constantly introducing lower power Integrated Circuits (ICs), which are the basis for reducing power consumption, and overall running costs of HPC/CC systems. Certain authors, as in [5] and [6], claim that advanced power management plays a key role in “greening” the computing systems. Some authors, e.g., [7] and [8] propose their vision of reducing the footprint of HPC energy footprint reduction through the use of advanced task management tools (high and low level schedulers and mappers) and frameworks. Other sources partially rely, amongst others, on “smarter applications”, efficient programming and reconfigurable compilers, such as in [9] and [10]. However, a true energy-efficient HPC is an ideal combination of all the aspects mentioned. Careless disabling of the compute nodes while they are not in use, switching power states too often, reducing CPU frequency too much, incorporating bulky resource monitoring systems, using over-complicated scheduling systems and algorithms, can do more damage than good, resulting in even minor performance systems and bigger power consumption. The more “Green” technologies are used, the greater care should be taken to successfully balance their impact on reducing power consumption, while trying to keep acceptably high performance. The existing research in this area is based on testing the green methods for computing systems, as in [11] and [12]. In [11], a bit more technical approach is given, and the authors are based mostly on greening the data centers and

servers. In [12], a survey and taxonomy for green computing systems is given, but not all the green aspects are covered. Other topics are discussed in detail.

III. THE “GREENING” LEVELS OF AN HPC SYSTEM

A. Hardware General

Driven by large expenses of running HPC systems, vendors prompted computer manufacturers to build their systems in a more eco-friendly manner. Creating power-efficient hardware is undoubtedly the basis for building Green HPC/CC systems. A single PC consumes on average 200W, which is the power of two light bulbs, but HPC systems such as computer grids and clouds consume much more. Computer grids punched the MW limit in power consumption, and every upgrade demands a new approach for cooling systems and power sources. As the world’s top supercomputing organizations found that mere increase in size and number of their systems has a negative impact on running costs, green supercomputing ideas were born. The process of greening computing resources started at hardware level. There were several key technologies that lead to significant energy savings. First, there was denser Very-large-scale Integration (VLSI) of chips, which ultimately led to multi-core chips and an increasing number of integrated components in the CPU die. Intel, for example, has recently announced a 22nm 3-D Tri-Gate technology, [13], which offers lower operating voltages and leaking currents in order to gain more performance per Watt (increasingly popular metrics used for measuring power-efficiency of a computer system). The same manufacturer has announced a 15nm technology by 2013 and 10nm by 2015, continuing to follow Moore’s Law. In this way, unnecessary buses are removed from the system, cache and memory bandwidth are increased, providing at the same time solutions to a major problem of HPCs – memory bandwidth limitation. Since these modern HPCs have the memory bandwidth bottleneck, increased energy efficiency is granted. Also, a new era of low power chips emerged, providing less horse power, but with greatly increased power savings and reliability. Hardware technology advances also promise greater control over energy consumption of other components in the computer system. Leading CPU manufacturers, including AMD, Intel, Sun and IBM, offered their representative low power CPUs, which clearly states the importance of power savings, especially for HPCs. The importance of the given fact applies also to Graphics processing units (GPUs), which have recently evolved from a special purpose to an efficient high performance processing units. The Chinese Tianhe-1A is a good example of harnessing GPU power in order to achieve true supercomputing performance, but at lower energy footprint.

With respect to gains, manufacturers claim that the 22nm technology could bring up to a 37% performance increase at low voltage compared to actual 32nm, and a 50% power reduction at constant performance. An increase in VLSI density resulted in multicore processors, CPU-GPU integration and high performance GPUs. Even the other components such as chipsets, PSUs, disk drives and network

systems, tend to have reduced energy footprint by using more advanced microcontrollers and ICs.

In relation with cons, as any other new technology, new production facilities should be built, which again questions the “greener process” of technology advance. But, considering many platforms based on the new technology, power saving benefits should be able to overcome production costs.

B. Power Management

The drive towards sustainable IT, which encompasses HPC and CC systems, has encouraged the creation of metrics claiming to quantify energy usage and apply objective math to the measurement of data center efficiency. Even different benchmarks for new age supercomputing systems are proposed, e.g., [14]. There are several metrics proposed for measuring data center power efficiency. The Green Grid, a consortium of IT industry experts has presented a series of proposals for IT facilities power measurement. The Green Grid proposes two key metrics for data center efficiency, and these are: Power Usage Effectiveness (PUE) and Data Center Efficiency (DCE), as in [15]. For example, the metrics of the former is based on the ratio between Total Facility Power (TFP) and IT Equipment Power (IEP):

$$PUE = \frac{TFP}{IEP}. \quad (1)$$

The Green Grid consortium, along with some others, including the work from [5], offer proposals for a complete power assessment of IT installations, which consist of analyzing present states, pinpointing weak points of systems which cause power inefficiency and giving propositions for improvements. Establishing a good Green metrics gives organizations valuable guidelines to reduce their costs by utilizing power management.

In addition, advanced power states have been incorporated in systems for years now. In green HPC, the reason for using multiple power states is to adapt HPC power consumption to real needs. The Advanced Configuration and Power Interface (ACPI) replaced old Advanced Power Management (APM), and introduced new techniques for more thorough power consumption suppression. The ACPI has the ability not only to reduce the processor speed, but also to monitor other components, thus providing greater versatility in a disabling system which is not used. The biggest power consumers in the computer system can be seen in Fig. 1, based on the survey in [11]. The low power states (also called the S-states) are used at the node level. The S-states which have the best power saving/wake up time ratio as suggested by authors, are S3 (“Suspend to RAM”), S4 (“Suspend to disk”) and S5 (“soft off”). By using these states, nodes are deactivated when they are not needed, and woken up or turned on when the running HPC/CC system demands more performance. This approach can greatly reduce power consumption, if the time and energy needed to power down or wake up the node do not affect the overall revenue. That is, changing power states can be performance and energy consuming, as highlighted in [16].

Gains can vary depending on the level of power plan adaptation, power state changing frequency, power changing cost of a system, etc. Researchers in [6] predict about 10 – 13% of power saved, up to 32% with energy proportional devices with deviance less than 5%.

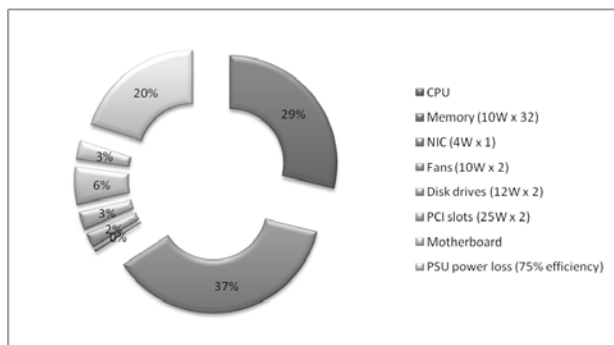


Figure 1. Power consumption in a typical HPC node, based on data from [14]

Power saving policies can significantly improve system scalability.

Cons are mainly based on the power changing costs (wake up time and energy), reduced reliability (especially for hard drives, because of their limited power on/off cycles - 40,000 on-off cycles claimed by HP for their machines), financial burden of adaptation to power requirements, etc.

C. Dynamic Voltage and Frequency Scaling, CPU and Memory Throttling

The primary power saving technology at CPU level is DVFS (Dynamic Voltage and Frequency Scaling), which enables current, voltage and frequency reduction of CPU when its utilization is below some threshold. Since energy consumption is proportional to frequency squared, DVFS offers a promising approach to reduce energy usage. Another benefit of lowering the frequency is the reliability. In the time of the “GHz race” (in 1990s and early 2000s), processors became more and more subject to failure, which from the perspective of a multi-processor system, such as Grid or Cloud, is an important issue to be reckoned with. However, larger voltage ranges do not improve power efficiency, as shown in [17]. They demonstrated that for sub-threshold supply voltages, leakage energy becomes dominant, making “just in time completion” energy inefficient. They also showed that extending voltage range below half V_{dd} will improve energy efficiency for most processor designs while extending this range to sub-threshold operations is beneficial only for specific applications. Supply voltage can be reduced if frequency of operation is reduced. If reduction in supply voltage is quadratic, then an approximately cubic reduction of power consumption can be achieved. However, it should be noted that frequency reduction slows the operation.

Memory throttling is similar to CPU throttling, but instead of lowering the frequency, it is basically the limitation of memory bandwidth based on the current memory bandwidth request. There are several memory

throttling technologies already in use, e.g., Intel’s Closed Loop Thermal Throttling (CLTT), and Open Loop Throughput Throttling (OLT).

The benefits of memory throttling are studied in [18], where authors managed to achieve up to 35% reduction of the total memory power consumption. There were some limitations, however, because the results have shown that 40% of memory power consumption is not controllable, also, a memory bandwidth was limited to 75%. In the system with the memory throughput up to 100%, as much as 60% of memory power consumption can be saved if memory is not needed.

Authors in [18] offer an insight into the types of applications where memory throttling does not improve power savings. These are the applications with low memory requirements, or highly optimized applications with heavy cache usage.

D. Power Aware OS

Low power operating systems are mainly researched for mobile platforms, and embedded systems. But the advances in these operating systems can be applied to the world of HPC. HPC/CC operating systems play the key role in resource efficiency. First, the OS should be aware of the current load of the node, and when the node is loaded, other non-essential tasks should be treated as low priority tasks and be given only a portion of resources. The OS should be aware of the task priority and adjust the resources accordingly. Other features of Green computing such as DVFS, advanced power states, throttling, task scheduling, load balancing and a dynamic adaptation of the HPC environment to the current need should be all issued by the HPC/CC operating system.

Greening level: as much power saving as a combination of installed power saving technologies (if the OS contains routines which successfully exploit them).

Cons: The OS with many processes (which manipulate the power saving technologies) can degrade performance of the HPC.

E. Virtualization

Virtualization is one of the fundamental software technologies that leads to a development of CC systems. Even though virtualization mostly applies to CC systems, migration of this technology to HPC is almost inevitable, so its energy savings can be taken into account when sketching “greener” HPC. Virtualization enables more thorough use of CC systems’ resources, because it provides an abstraction of real resources and resource transparency. If a node is to be used, a Virtual Machine(s) (VM) takes possession of the nodes’ resources up to VMs maximum. If not all of nodes’ resources are taken, another VM or VMs can occupy the rest of the same physical node, which enables efficient use of the resources and application scaling. There are several virtualization technology vendors active, such as Xen, VMWare, KVM and Virtualbox, and a lot of work is based on virtualization in the HPC world (e.g., [12], [19], [20]), thus the term can be regarded as one of the postulates of Green HPC/CC. VMs are the media which can hold other

“green” technologies, such as dynamic power policies, managing Source Level Agreements (SLAs) in cloud computing systems, DVFS, and workload balancing.

Gains are the combination of gains from other “green” technologies contained in VM, especially excellent scalability and resource provisioning.

Cons: VMs can be inert, so resource scaling can be slow. Every VM contains resources for its internal functionality (by OS), so if there are several VMs on one physical node, a great deal of resources is dedicated to it (mostly memory and disk space).

F. Task Scheduling, Application Granulation

A lot of work was conducted in this area even before green computing ideas were born, cf. [21]. The high application granularity level is a key for economical resource provisioning and application scalability, especially on heterogeneous computer systems. That includes optimal parallel programming in the way of making independent tasks that can be executed concurrently. After the application is submitted to HPC resources a high level scheduler takes care of running the application instances, and a low level scheduler/mapper assigns tasks to resources (nodes, processors, cores) in HPC environment, more in [7]. The tasks should be kept as small as possible, so that the applications could be more scalable, and efficiently use up computing resources. There are newer approaches, which already implement energy aware scheduling, for details see [8].

Gains: An intelligent scheduling/mapping system can lead to significant utilization efficiency, good resource provisioning and increased scalability. The exact energy savings can vary.

Problems in this area are caused by different task sizes, which refer to highly heterogeneous jobs for running on computer systems, data dependencies among tasks, scheduling duration and Amdahl’s law.

G. Power Aware Compilers

Not only that power aware compilers should be present in embedded systems to reduce power consumption while compiling an application, they could also improve power efficiency of applications running on HPC nodes. These compilers would have direct control over DVFS, DCD (Dynamic Component Deactivation) and power management routines for resources in the HPC system. At the moment of compiling a code, a compiler should adapt the application for efficient execution on the proposed systems, and be aware of the past executions of similar applications and adjust resource usage accordingly. Automatic parallelization compilers could serve as a potent platform for additional power, voltage and frequency aware compilation, and could, in addition to efficient parallelization, improve usage of HPC system resources.

An interesting example, see [19, p. 72], shows that by using the ifort 9.1 compiler for compiling a simple matrix times a matrix (MxM) operation on Pentium 4 running at 2.8GHz, the two loops deliver 0.97GF/s, and by using ifort 10.0 performance rises to 2.6GF/s. From the aforementioned

example it can be seen that a certain code can be several times faster on a more efficient compiler. Some of the compiler optimization methods are automatic loops exchange, automatic loop enrolling, replacing the subroutine calls with direct kernels, ignoring the “if” statements in the loops, etc.

Cons: sometimes a more advanced compiler has a trouble understanding the programmer’s intention, which can output wrong results.

H. Resource Utilization Monitors

Resource Utilization Monitors (RUMs) should contain up-to-date information about resource utilization in the systems. The information about system load would then be transferred to other vital components of HPC, such as OS, Virtual Machine Managers (VMMs), power aware compilers, prediction policies and task schedulers. Such monitor is the basis for a highly promising green technology – the prediction. Various logs about application runs, errors, power state changes, voltage and frequency changes, VM migrations, CPU and memory load, and other components usage over time can enhance statistical analysis of resource use. An example of such proposed system can be found in [22], describing the OVIS project – an attempt to exploit HPC resources over cloud services. The approach addresses a scalable collection and analysis of resource metrics from both component-health and resource utilization perspectives, and hence it can contribute to the application-tailored resource allocation of hardware and the subsequent allocation and/or migration of virtual resources on the hardware.

Pros: works concurrently with prediction technologies and task scheduling. RUMs can enhance statistics for investing in a HPC upgrade.

Cons: a resource monitor and load balancer uses resources as well, and in the case of a highly heterogeneous environment it can lead to periodic slowdowns.

I. Prediction, Analysis of Past Executions

By monitoring the HPC system load, efficient tuning and adjustments can be made in order to improve future application executions. That involves smart application mapping, removing unnecessary resources, disabling error-prone nodes, analysis of power wastage sources, etc. Significant improvements can be made to reduce power consumption in HPC systems by applying knowledge gained by resource monitoring systems. It is essential that the latest information collected has the greatest priority in future run-time decisions because this technique ensures acceptance of every system change, hardware and/or software-wise without the need to reset the logs.

Pros: most benefits of prediction can be felt in large systems, where statistics is less prone to errors.

Cons: The “bed in” time of such system can make it inoperable in the first period of use. Also, in smaller systems there is more influence of special cases in the statistics of the load, which can lead to false analysis.

J. Component-based Software Engineering

Based on [10], Component-Based Software Engineering (CBSE) is an approach that deals with making different software by combining reusable software components. Making reusable/recyclable software lowers the software production costs significantly, by reducing the equipment and programmer utilization, what indirectly leads to power savings. However, an impact of CBSE to Green HPC is not entirely known, and it is part of future research that will be done by authors.

Pros: Lower software production costs.

Cons: Components can often be very large, leading to unnecessary overhead and diminished gains.

K. Programmer's Contribution

All the research in making "greener" HPCs in the world would not suffice if the programmer's role in efficient use of resources is ignored. This is the final frontier for a battle against the energy spill in HPC/CC systems, and thus the most important one. Programmer's role gives utter importance to conducting research in performance optimizations in programming, and making the right tools for parallel programming, debugging and compiling. After all, if all of the technologies mentioned in this work are inaccessible by the programmer, they cannot be utilized.

Pros: A high level of power adaptation for every aspect of HPC/CC systems.

Cons: Programmer's role is becoming more automated, and error-prone.

L. Cyber-physical Approaches

Several researches regarding usage of physical characteristics of HPC nodes in order to reconfigure the system or reschedule tasks are conducted, but they are still in their infant phase. In [20], thermal-aware scheduling was proposed with the purpose of reducing hot air recirculation among nodes in the HPC system. The load is balanced by means of scheduling tasks according to the node temperature. Other propositions involve various real-time power meters attached to nodes with the purpose of scaling the load corresponding to current power consumption.

Pros: Many physical characteristics can complement the information needed to increase energy efficiency. These characteristics often carry true information about the node state, and bring valuable information on how to upgrade energy efficiency.

Cons: Cyber-physical approaches demand additional systems, which inject supplementary problems into HPC environment. There is also limited physical information which is relevant to power efficiency improvement.

IV. GREEN HPC BLUEPRINT

If the taxonomy from [9] is adopted, the HPC or CC systems can follow the multi-tiered hardware architecture shown in Fig. 2. Every method/solution from Section 3 can be applied to (a) certain HPC components tier(s).

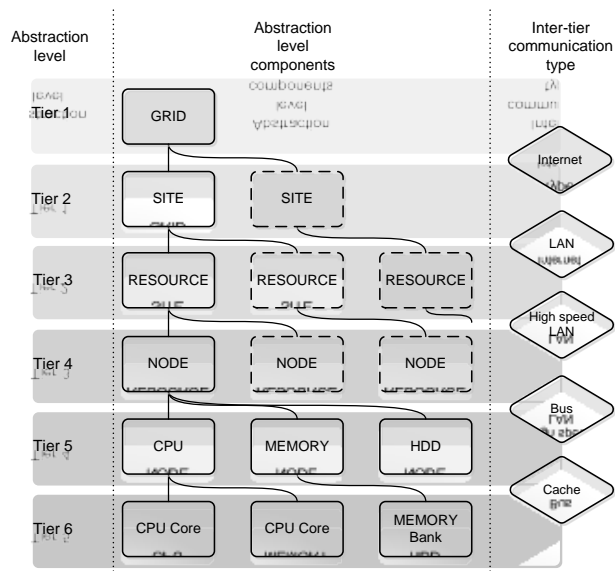


Figure 2. Tiered architecture of a typical HPC system

Fig. 2 in conjunction with Table 1 forms a picture for green HPC blueprints. Table 1 represents corresponding tiers for "greening" method implementations. Tiers are deliberately arranged from the highest to the lowest, because the purpose is to show the highest tier of implementation first, and also because without power efficiency of high tier components, power gains in the lower ones cannot be efficiently exploited. The first level of improvement is undoubtedly the manufacturing technology of hardware. The improvements at these levels stretch through the whole HPC systems, and present the basis for power production. Power management can be applied to tiers 6, 5 and 4, as well as to the network. Changing power states at tier 3 level is not convenient. DVFS and throttling can achieve power savings at tiers 6 and 5. Lower tiers can gain much and improve greening technologies at these levels. A power-aware OS is implemented in tier 4; these are nodes and virtual machines. Virtualization integrates at tier 5 or tier 4, depending on the SLA in the CC systems. Scheduling can be applied to multiple tiers of the HPC system. Scheduling abstraction is based on the current needs and application properties. Power-aware compilers are implemented at tier 4, but they can directly benefit from all the power-saving technologies of higher tiers. Resource monitoring systems and prediction based methods share the same tier of implementation. These are often tiers below 4, depending on the amount of control over HPC. CBSE, as the majority of software power saving techniques, is implemented at tier 4. Programmer's contribution can be indirectly installed at every tier, but most frequently controllable tiers today are tier 4 and 3. With the advances of future systems, high tier components are becoming more tightly coupled, so the hybrid green methods are going to be introduced. These will comprise several green technologies combined in power efficient fashion, so either of which does not diminish gains of other ones.

TABLE I. POWER EFFICIENT TECHNOLOGIES – TIERS OF APPLICATIONS

“Greening” technology	The tier of implementation
Hardware production technology	tier 6, tier 5
Power management	tier 6, tier 5, tier 4
DVFS, Throttling	tier 6, tier 5
Power-aware OS	tier 4
Virtualization	tier 5, tier 4
Scheduling	tier 6, tier 5, tier 4, tier 3
Power-aware compilers	tier 4
Resource monitoring systems	tier 4, tier 3, tier 2
Prediction	tier 4, tier 3, tier 2
CBSE	tier 4
Programmer contribution	tier 4, tier 3

V. CONCLUSION AND FUTURE WORK

Future HPC and CC systems will have to deal not only with the performance upgrades, but also with the increasingly present green standards. Energy costs of running these systems can be overwhelming even for large institutions. Numerous projects try to address various issues regarding the reduction of power consumption in these systems. These are primarily hardware advances, which cannot be exploited without appropriate software. Prior to applying green techniques for HPC and CC systems their overall revenue should be investigated, because not all of them bring power savings and high performance. This is the reason which implies that such attempts should be thoroughly modeled first. We presented different technologies for “greening” the HPC/CC systems, their pros and cons, and a level of HPC/CC system where these can be applied. Every “green” technology was referred to related work and an example was given. In future work, we plan to extend modeling of “green” technologies in HPC and CC environments, and to investigate new means of enhancing power savings in these systems. The possible combinations of green technologies and hybrid green technologies are going to be researched, as well as their gains compared to contemporary ones. Also, the impact of reusable software in terms of software cost reduction, and thereby also in terms of green HPC systems, is going to be evaluated.

ACKNOWLEDGMENT

This work was supported by research project grant No. 165-0362980-2002 from the Ministry of Science, Education and Sports of the Republic of Croatia.

REFERENCES

[1] M. Warren, E. Weigle and W-C Feng, “High-Density Computing: A 240-Node Beowulf in One Cubic Meter”, Proc. ACM/IEEE High-Performance Networking and Computing Conf., Baltimore, MD, USA, Nov. 16-22. 2002, pp. 1-11.

[2] IEEE Standard 1680.1, Section 4, “Environmental Performance Criteria for Desktop Personal Computers, Notebook Personal Computers and Personal Computer Displays”, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1633760, accessed: April 2011.

[3] ENERGY STAR Program Requirements for Computer Servers, www.energystar.gov/ia/partners/product_specs/programreqs/Computer_Servers_Program_Requirements.pdf, 2010, accessed: March 2011.

[4] Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Directives, www.rohs.gov.uk, accessed: March 2011.

[5] J.R. Stanley, K.G. Brill and J. Koomey, “Four Metrics Define Data Center Greenness”, [http://uptimeinstitute.org/wp_pdf/\(TUI3009F\)FourMetricsDefineDataCenter.pdf](http://uptimeinstitute.org/wp_pdf/(TUI3009F)FourMetricsDefineDataCenter.pdf) 2007, accessed: Feb 2011.

[6] T. Minartz, J. M. Kunkel and T. Ludwig, “Simulation of Power Consumption of Energy Efficient Cluster Hardware”, Computer Science - Research and Development, Vol. 25, Numbers 3-4, 2010, pp. 165-175.

[7] Ripal Nathuji, Canturk Isci, and Eugene Gorbatov, “Exploiting Platform Heterogeneity for Power Efficient Data Centers”, Proc. IEEE Int. Conf. on Autonomic Computing, Jacksonville, FL, USA, 11-15 June 2007, p. 5.

[8] L. Wang, G. von Laszewski, J. Dayal and F. Wang, Towards Energy Aware Scheduling for Precedence Constrained Parallel Tasks in a Cluster with DVFS, Proc. IEEE/ACM Int. Conf. on Cluster, Cloud and Grid Computing, Melbourne, Victoria, Australia, 17-20 May, 2010, pp. 368-377.

[9] R. Gruber and V. Keller, HPC@Green IT, Springer, 2010.

[10] J. Sametinger, Software Engineering with Reusable Components, Springer-Verlag, 1997.

[11] L. Minas and B. Ellison, “Energy Efficiency for Information Technology: How to Reduce Power Consumption in Servers and Data Centers”, Intel Press, 2009.

[12] A. Beloglazov, R. Buyya, Y.C. Lee and A. Zomaya, “A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems”, Advances in Computers, Vol. 82, 2011, pp. 47-111.

[13] J. Bruner, “Intel 22nm 3-D Tri-Gate Transistor Technology”, Intel Newsroom, <http://newsroom.intel.com/docs/DOC-2032>, May 2011, accessed: May 2011.

[14] Graph 500, <http://www.graph500.org>, accessed: Jan. 2011.

[15] Recommendations for Measuring and Reporting Overall Data Center Efficiency, www.thegreengrid.org/~media/WhitePapers/RecommendationsforMeasuringandReportingOverallDataCenterEfficiency2010-07-15.ashx?lang=en, 2010, accessed: May 2011.

[16] N. Vasić, M. Barisits, V. Salzgeber and D. Kostić, “Making Cluster Applications Energy-Aware”, Proc. 1st Workshop on Automated Control for Datacenters and Clouds, Barcelona, Spain, June 19, 2009, pp. 37-42.

[17] B. Zhai, D. Blaauw, D. Sylvester and K. Flautner, “Theoretical and Practical Limits of Dynamic Voltage Scaling”, Proc. 41st Ann. Design Automation Conf., San Diego, CA, USA, June 7-11, 2004, pp. 868-873.

[18] H. Hanson and K. Rajamani, “What Computer Architects Need to Know About Memory Throttling”, Proc. IBM Workshop on Energy Efficient Design, Saint Malo, France, May 14, 2010, pp. 44-49.

[19] D.A. Menasce and M.N. Bennani, “Autonomic virtualized environments”, Proc. Int. Conf. Autonomic and Autonomous Systems, Silicon Valley, CA, USA, July 19-21, 2006, p. 28.

[20] T. Mukherjee, A. Banerjee, G. Varsamopoulos, S.K.S. Gupta and S. Rungta, “Spatio-Temporal Thermal-Aware Job Scheduling to Minimize Energy Consumption in Virtualized Heterogeneous Data Centers”, Computer Networks, Vol. 53, Issue 17, 2009, pp. 2888-2904.

[21] H.J. Siegel and S. Ali, “Techniques for Mapping Tasks to Machines in Heterogeneous Computing Systems”, Journal of Systems Architecture, Vol. 46, Issue 8, 2000, pp. 627-639.

[22] J. Brandt, A. Gentile, J. Mayo, P. Pébay, D. Roe, D. Thompson and M. Wong, “Resource Monitoring and Management with OVIS to Enable HPC in Cloud Computing Environments”, Proc. IEEE Int. Symp. Parallel & Distributed Processing, Anchorage, AK, USA, 16-20 May 2009, pp. 1-8.