The Impact of Improving Software Functionality on Environmental Sustainability

Sedef Akınlı Koçak[1], Andriy Miranskyy[2], Gülfem Işıklar Alptekin[3], Ayşe Başar Bener[1], Enzo Cialini[2]

[1]Ryerson University, Ted Rogers School of Information Technology Management, 55 Dundas St. West, Toronto, M5B 2K3, Canada
sedef.akinlikocak@ryerson.ca, ayse.bener@ryerson.ca

[2]IBM Toronto Lab., 8200 Warden Ave, Markham, Toronto, L6G 1C7, Canada
andriy@ca.ibm.com, ecialini@ca.ibm.com

[3]Computer Engineering, Galatasaray University, Çırağan Cad. No: 36, 34357, Istanbul, Turkey
gisiklar@gsu.edu.tr

ABSTRACT

Green IT aims to achieve environmentally sustainable computing. Applying this concept to the existing systems to meet business demand or designing sustainable software is therefore a complex task. In this paper, we have discussed that improving existing software functionality or leveraging legacy systems according to end-users' requirements has a significant impact on environment. The researches show that many companies will have no other option than complying with the new green IT field. Hence, we have investigated the trade-off between improving software functionality and reducing energy consumption of a software product. Data compression is one of the software features which reduce the number of I/O operations while increasing CPU utilization. In this research, we have focused on the impact of this feature on software's energy consumption.

Keywords

Green IT, Legacy Systems, Green Software, Energy Consumption, Energy Efficiency, Environmental Sustainability

1. INTRODUCTION

Information technologies (IT) that require vast amount of energy and other sources are used in almost every field and processes. Recently, the global carbon dioxide emission has had its highest level in human history with 9.1 billion tones and 49% higher than in 1990 (the Kyoto reference year) [7]. At least 2% of global carbon emissions are directly due to IT systems [17] and further increase is expected since new IT systems are applied every day. Therefore, reducing the energy consumption and related carbon emission of the IT systems is becoming a crucial requirement. This global issue promotes the competition and the companies are being forced to implement energy efficient products and energy efficient technology services around the world. The competition along with regulation and standards for measuring energy efficiency will continue to rapidly drive energy efficiency [21]. In this context, green IT is an ideal way for most companies to address the environmental issues in order to achieve environmental sustainability. Since green IT has many different aspects inside and outside the data center, it is important to manage it by using the resources efficiently to reduce environment impacts [21]. Although a software system does not directly consume energy, it intensely affects the hardware functioning, hence its energy consumption. All the infrastructural layers in a data center amplify the energy consumption induced by software [6].

Most IT companies need to transform their applications to meet new business demands. On the other hand, transforming these systems may also cause a rebound effect. In that case, efficiency is a necessary but not a sufficient condition for saving resources. Hilty and Lohmann [19] emphasized that recent studies almost completely ignore the rebound effect. They also mentioned that software development plays a specific role in creating rebound effects.

The ideal solution is to transform legacy systems to newer, more productive platforms so that companies can exploit faster and cheaper development technologies. However, legacy systems tend to be unwieldy, monolithic, and inflexible and many firms regard modernization as somewhere between improbable and impossible [15]. Furthermore, in the acquisition of large-scale software systems, the effective and efficient management of user requirements is one of the most crucial issues [37]. Software systems that lack appropriate non functional requirements (NFR), i.e., how the system behaves with respect to some observable attributes, such as performance and reliability [36], carry the risk of failing to meet customer needs [38]. Zhu et al. [38] stated that there is a lack of research on NFRs trade-offs in the systems.

As software continues to affect all aspects of our lives under ever-renewed forms, we realize that leveraging existing systems is a challenging concern for the companies. Keeping the software on demand and with high quality levels in respect to end-users' requirements create a conflict in terms of software energy consumption. Moreover, each integrated quality feature is accompanied by increasing levels of energy consumption. Therefore, it is hard to maintain software as environmental friendly.

Our contribution presents the rising awareness of environmental sustainability among IT studies/ practices and green software. We

ICT4S 2013: Proceedings of the First International Conference on Information and Communication Technologies for Sustainability, ETH Zurich, February 14-16, 2013. Edited by Lorenz M. Hilty, Bernard Aebischer, Göran Andersson and Wolfgang Lohmann.
DOI: http://dx.doi.org/10.3929/ethz-a-007337628

analyze the impact of transforming legacy systems with respect to new business demands on the carbon dioxide emission. As the case study, the data compression feature of a large scale legacy software system is tested. We have analyzed the total energy consumption of the software when processing a specific workload in two different scenarios: 1. Without data compression feature, 2. With the data compression feature. It is also remarkable to state that the same business goal can be reached using different amount of energy consumption. We have shown that the feature increases the total consumption, but decreases average consumption per statement due to significant reduction of execution time.

The rest of the paper is organized as follows. Section 2 discusses the related work in the literature and their differences from this one. Moreover, brief introductions to the concepts of green IT and legacy system's modernization are given in Section 2. In Section 3, the simulations are given. The results are discussed at the end of the Section 3. Finally, conclusions and future work are given in Section 4.

2. BACKGROUND AND RELATED WORKS

2.1 IT and Environment

IT affects on environment in many different ways. Each stage of a computer's life, from its production, throughout its use, and into its disposal, presents environmental problems. Manufacturing computers and their various electronic and non-electronic components consumes electricity, raw materials, chemicals, and water, and generates hazardous waste. All these directly or indirectly increase carbon dioxide emissions and impact the environment. The total electrical energy consumption by servers, computers, monitors, data communications equipment, and cooling systems for data centers is steadily increasing. This increase in energy consumption results in increase greenhouse gas emissions. Each PC in use generates about a ton of carbon dioxide every year [24].

At the Forum for the Future in 2002, the ascertained effects of IT on environmental sustainability were structured mainly in three levels. First order effects (direct *effects*) include all environmental impacts resulting from hardware during the product life cycle, covering production, use and disposal. Second order effects (*indirect effects*) are impacts and opportunities created by the ongoing use and application of IT. Finally, third order effects includes impacts and opportunities created by the aggregated effects of large numbers of people using and adapting of behavior (e.g. consumption patterns) or economic structures due to the stable availability of IT and the services it provides over the medium to long term [13].

Analyzing the direct environmental impacts of ITs are considerable in areas such as energy use, materials throughput and end-of-life treatment. After the financial crises of 2008-2009 green economy is seen as a framework for restoring economic growth meanwhile responding to the climate change and other subjects of environmental sustainability [9]. Governments' "green IT" policies can be instrumental in promoting life-cycle approaches for improved R&D and design of IT goods, services and systems national and international level. Green IT is a topic and initiative that has emerged recently to address this role of IT on environment [2].

New CompTIA "Green IT Insights and Opportunities-2011" survey results indicate that businesses today are hungry for eco-friendly technology solutions, and that most firms now make hiring decisions based on IT providers' green credentials. It showed that energy conservation is a key factor when reviewing IT companies. Moreover IT company's green credentials and how knowledgeable they are about implementing green initiatives are important hiring factors. The study also reveals that 35% organizations report having a comprehensive environmental strategy for practices such as reducing energy consumption, equipment usage/design, recycling/product disposal, carbon footprint and employee behaviors. Additionally, 42% have a partial environmental strategy [11].

2.2 Green IT

Green IT is a study and practice of using computing resources efficiently [21] to reduce carbon dioxide emissions of information technology. Recently, the more broad definition of green IT is provided by Murugesan [25] "*Green IT is the study and practice of designing, manufacturing, using, and disposing of computers, servers, and associated subsystems (such as monitors, printers, storage devices, and networking and communications systems) efficiently and effectively with minimal or no impact on the environment*". These definitions of Green IT's initial focus on reducing the carbon emissions emanating from IT equipment and infrastructure [34]. The most significant and immediate reductions come from hardware solutions. The whole lifecycle of computers and related equipment, such as the manufacturing process and disposal and recycling are mostly focused areas. There are numerous studies on material level of IT (hardware) for energy efficiency. Most of the studies have been done on the hardware and Life Cycle Assessment for computers, mobile phone networks and monitors [14, 18, 32, 34].

Moreover, the cost to maintain data centers and their enormous use of power to both run and cool their infrastructure makes data centers a prime target for greening efforts [27]. According to the U.S. Environmental Protection Agency (EPA) Report, the electricity use of the nation's servers and data centers has a rapid upward trend from 2000 to 2005. However, the growth trend has slowed significantly between 2005 and 2010 [20].

In comparing with attention of the hardware solutions, there are rare cases in which they focus on software [12]. However, software also plays an important role having a considerable impact on energy consumption. Recently, Naumann et al. [29] stated that there is a lack of models and descriptions regarding environmental sustainability in the area of computer software. Thus, they contributed a definition of the green and sustainable software, *"Green and Sustainable software is software, whose direct and indirect negative impacts on economy, society, human beings, and environment that result from development, deployment, and usage of the software are minimal and/or which has a positive effect on sustainable development".* They also developed a reference "GreenSoft" model for software products, sustainable metrics and criteria for software design and development [29]. There are various frameworks and models of evolving software, such as sustainable computing concept classifies different direction of hardware and software [22], common software quality aspects and developments of new metrics to measure these quality aspects in order to achieve environmental effects [3], investigation of different software systems on IT energy consumption [5] and analyze software carbon footprints from a typical software lifecycle and estimation of how large carbon footprint each step produces [33].

Chen et al. [8] explained organizational motivation for Green IT supported in the context of ecological sustainability with eco-efficiency, eco-equity and eco-effectiveness. Those concepts were further developed by [23]. There is an on-going process of Green IT adoption by companies. Most businesses are re-considering their IT architectures and installations by emphasizing the Corporate Social Responsibility (CSR) aspect of green IT. In this manner, a framework proposed by Viaro and Viccaro [35] considers that environmental changing requirements, strategic requirements and dynamic capabilities are the forces which move organizations toward green practices and foster innovation.

2.3 Legacy Systems' Modernization

The old IT systems that are modernized or replaced are called *legacy systems* [31]. Especially, discovering the service-oriented architecture (SOA) as the future technological underpinning of enterprise information technology, software modernization has become challenging for the most of the software engineers [1]. Those systems are, in any cases, vital to the organization that uses them. They are, however, hard and expensive to manage. It has been realized that the maintenance and evolution costs of legacy systems are normally somewhat between 40% and 90% of the total costs of the life cycle of the system [31]. According to the National Association of State Chief Information Officers [NASCIO CIO Priorities] surveys of the state CIOs to identify and prioritize the top policy and technology issues facing state government, legacy application modernization and upgrade is one of the forth priority technologies, applications and tools since 2006 while Green IT technologies and solutions is one of the priorities in 2009 [28]. Similarly, in 2007, IBM's seven countries across six industries survey shows that legacy applications and legacy modernization represents a major challenges across all the industries. Nearly half of all respondents report that the inflexibility of legacy systems poses a significant challenge [4].

The modernization of state IT legacy systems is emerging as a significant financial, technical and programmatic challenge to states' ability to deliver services to citizens, and conduct day-to-day business. From a survey conducted by NASCIO, back in 2008, legacy systems in operation have been labeled and given criteria according to the potential problems expected when dealing with them. 82.8% of the respondents have identified legacy systems as those that can no longer be "adequately supported, maintained or enhanced"[26].

Some researches show that most of the modernizing effort of the software fails [30]. Legacy system modernization efforts fail for variety of reason such as complexity, software technology and engineering processes, risk, commercial components and business objectives. Most of these problems arise when handling upgrades and maintenance on the legacy software. Naturally, most of applications may no longer be adaptable, extensible or agile enough to compete or interact with modern web-based systems. Moreover, the application tools and documentation may be far out-of-date or non-existent at all to aid our IT staffs in handling problems that may crop up prior to modernizing the systems, like the potential security-risks in accessing the information stored in some of these legacy systems [16].

A high-quality legacy system that provides a competitive advantage is worth nurturing unless external business pressures dictate change [8]. System evolution which covers maintenance, modernization and replacement and software reengineering and modernization which covers retargeting, revamping, use of commercial components, source code translation, code reduction, and functional transformation have been studying to reduce the overall effort required to maintain the ever-increasing amount of legacy software code [30]. The modernizing alternatives are not mutually exclusive and the decision of the approach to use is generally based on an assessment of the quality and business value of the system [10].

As a result, software modernization requires making trade-offs. These trade-offs are multifaced and include technical and organizational considerations that may strain and organization's decision making abilities [30].

There are some survey studies conducted on legacy modernization approaches. According to the Aberdeen Group survey in 2006 given a choice of legacy modernization approaches, 58% of the companies' options for the more economical, less complicated maintenance option over replacing and modernizing [1]. Survey results also indicate that the major "driver" is to improve IT time to deliver changes to the business (about 70%) followed by lower IT integration costs (about 50%). Similarly, according to the NASCIO survey in 2008, in the States major driving force of moving towards modernization of IT systems and applications is changing or re-engineering of business processes with 86% followed by "line-of-business" requirements with 83%. It is also important to show that end user demand has been 52% and a green IT initiatives has been 28%. Moreover, primary concerns around legacy systems are software maintenance/upgrades, extensibility, adaptability, agility, and application development tools. This appears to demonstrate that primary concerns are centered on application software and its ability to be adapted, upgraded and maintained [26].

3. CASE STUDY

3.1 Description of Software under Study

Our software under study is DB2 for Linux, UNIX and Windows Version 10.1. DB2 is a good candidate for our analysis, because it is a large software product present on the market since 1992 with a considerable market share. Its ancestor, DB2 for mainframe, was born in 1983 (and research prototypes have been created in 1970s). The product is constantly evolving. Examples of features added in the last decade are enhanced business analytics and data visualization.

We have focused on measuring efficiency of the DB2 workload in the presence and absence of the data compression feature. Let us look at the feature in details.

Disk storage systems can often be the most expensive components of a database solution. For large warehouses or databases with huge volumes of data, the cost of the storage subsystem can easily exceed the combined cost of the hardware server and the data server software. Therefore, even a small reduction in the storage subsystem can result in substantial cost savings for the entire database solution. Data compression reduces storage requirements, improves I/O efficiency, and provides quicker access to the data from the disk.

The latest release of DB2 introduced new type of data compression: "adaptive data compression". This feature utilizes a number of compression techniques (including table-wide and age wide compression [DB2 Manual[1]]) leading to significant reduction of storage space. However its usage can lead to CPU overhead associated with compression and decompression of the data.

Another positive "side effect" of this technology is speeding up of I/O intensive workloads (in spite of CPU overhead). Reading data

[1]http://pic.dhe.ibm.com/infocenter/db2luw/v10r1/index.jsp?topic=%2Fcom.ibm.db2.luw.admin.perf.doc%2Fdoc%2Fc0053982.html

from disk to memory for processing is one of the slowest database operations. Storage of compressed data on disk leads to fewer I/O operations needed to retrieve or store the same amount of data (in comparison with the uncompressed dataset). Therefore, for disk I/O-bound workloads (for instance, when the system is waiting/idling for data to be accessed from the disk), the query processing time can be noticeably improved. Furthermore, DB2 processes buffered data in memory in its compressed form, thereby reducing the amount of memory consumed compared to uncompressed data. This has the effect of immediate increase in the amount of memory available for the database without having to increase physical memory capacity. Allowing the additional memory to remain or freeing it up for other database or system operations. This can further improve database performance for queries and other operations. Compression can be turned on when tables are created using the "COMPRESS YES" option. Alternative, administrator would enable compression of an existing table *T* by executing "ALTER TABLE T COMPRESS YES".

3.2 Workload Description and Case Study Setup

We have focused to measure the effect of two actions; no compression of data and compression of data, on the amount of resources (time and electricity) needed to complete a certain workload.

Our reference workload is TPC-H[2]. It is created by the Transaction Processing Performance Council[3] and is used as the industry standard for measuring database performance. The workload consists of a set of business-oriented ad-hoc queries. The database has been designed to have broad industry-wide relevance (see TPC-H specifications[4] for details).

Using the tools provided with the workload, we have populated the database with 1GB of raw data and have generated 240 distinct queries associated with this dataset.

The queries were executed sequentially for approximately two hours in a circular fashion on a Lenovo ThinkPad T60 laptop with 3GB of RAM. Some of the statements consume significant amount of time. For instance, a query started at 1:59 can take 5 minutes, finishing at 2:04. Two hour interval is selected to reduce measurement error associated with low precision (0.01 kWh) of the energy meter (UPM EM100).

We have counted the number of statements executed in a given time interval and have measured the amount of electricity consumed by DB2 running in the following two configurations:

1. Without compression
2. With compression

The workload has been executed against each configuration three times to estimate measurement error (expressed using relative standard error). Relative standard error is calculated as the sample estimate of the population (in this case, 3 workload runs) standard deviation divided by the square root of the sample size and by the population mean. Compression ratio is defined as (1-*compressed size/uncompressed size*). The energy metric is measured in Watt per transaction per second (*W*/*tps*) and is calculated as *Energy consumption*/*Work completed*[5].

To summarize, for each DB2 configuration, we run the same workload three times for approximately two hours, measuring 1) the number of statements executed and 2) the amount of electricity consumed in a given time frame.

3.3 Results and Discussions

The measurements are given in Table 1. First, we have looked at the average execution time per statement. The slowest configuration of the database is, as expected, the one with the feature disabled. We have treated this configuration as a baseline. Compression of data reduces tables' size by 61% and leads to 97% performance improvement in comparison with the reference configuration.

Compression feature increases overall power consumption by 29% (in comparison with the reference configuration). However, energy consumption per unit of work is reduced by 34%, due to increased query throughput.

Based on the data, we have concluded that legacy system modernization (increasing the system functionality) has a mixed effect on energy consumption: consumption per unit of time goes up; consumption per unit of work goes down. Theoretically, if the workload size is fixed, hardware is dedicated only to this workload, performance of the workload is not critical, and space savings are irrelevant, it may be beneficial to run queries against uncompressed version of the database. These requirements are rarely met in practice: workloads scale up, hardware is shared between multiple tasks (especially in the virtualized / cloud computing environments), and space saving is critical.

3.4 Threats to Validity

This case study shows that the method can be successfully applied to a particular dataset. Following the paradigm of the "representative" or "typical" case advanced in [38], this suggests that the same approach may be extrapolated to other products.

Our test system is not designed for use in a production environment. It is a laptop (tuned to minimize electricity consumption, sacrificing efficacy) with consumer-grade operating system. However, we believe that the results can be extrapolated to a production system, since data compression is system-agnostic. The amount of savings for the production system will vary with system's setup and with associated workload.

4. CONCLUSION

Most IT companies need to transform their applications to meet new business demands. Sometimes these efforts come at a cost of consuming other resources within the organization. So far, IT industry has been focusing on IT equipment processing power and associated equipment spending. Recently it's been concerned with other requirements such as power, cooling, and data center space. Companies have also been under pressure to develop environment friendly practices in their project life cycle. However, going forward, they will need to deal with all of the infrastructure requirements and the environmental impact of IT itself (hardware and software) and its use. The challenges of green IT are immense; however, recent developments indicate that the IT industry has the will and conviction to tackle our environmental issues head-on. Companies can benefit by taking these challenges

[2] http://www.tpc.org/tpch/

[3] http://www.tpc.org/information/about/abouttpc.asp

[4] http://www.tpc.org/tpch/spec/tpch2.14.4.pdf

[5] TPC-Energy-Specification-1.2.0.pdf, page21

as strategic opportunities [24]. As a result, software modernization requires making trade-offs between end-user demands and the requirements for corporate social responsibility initiatives. In this paper, we conclude that considering environmental sustainability legacy system modernization regarding increasing the system functionality has a mixed effect on energy consumption. Therefore requirements to design new systems or modify existing ones are a complex task. Improving existing software functionality or leveraging legacy systems for end-users also has significant impact on environment. Moreover different features can have a significant impact on the energy consumption of software. We have related the use of software modernization efforts and software energy consumption suggesting that there may be a trade-off between legacy system modernizations and energy consumption.

In summary, we recommend that software development organizations should take into consideration the need for energy consumption from software especially from legacy systems which has strong end-user demand. We believe that there is a strong need for more input from software engineering on how to greening software toward creating more sustainable environment.

As a future work, individual effects and joint effects of different features on energy consumption will be examined. Going forward, trade-off models can be also examined in software development for environmental sustainability.

5. ACKNOWLEDGEMENT

This research is in part supported by NSERC Discovery Grant no. 402003-2012. We would like to thank IBM Canada Lab-Toronto site for making their development data available for research and strategic help during all phases of this research. The opinions expressed in this paper are those of the authors and not necessarily of IBM Corporation.

Table 1. Workload and database characteristics with and without adaptive compression feature (± denotes relative standard error)

Database Feature	Average Statement Count per hour	Average Electricity Consumption (kWh)	Space Consumption (Mb)	Compression ratio	Watts per statement per second[6]
no compression	415 ± 1.8%	0.035 ± 0.5%	1168.2	100%	302
adaptive compression	814 ± 4.0%	0.045 ± 0.0%	455.6	61%	199

6. REFERENCES

[1] Aberdeen Group. 2006. The Legacy Application Modernization Benchmark Report: Transforming Mainframe, AS/400, and Unix Applications in an SOA World, Available at: http://www.microfocus.com/000/RA_SOLA_PK_spf100306a_tcm21-12798.pdf.

[2] Ahola, J., Ahlqvist, T., & Ermes M. 2010. ICT for Environmental Sustainability, Green ICT Roadmap, VTT Tiedotteita, Finland– Research Notes 2532 Available at: http://www.vtt.fi/publications/index.jsp.

[3] Albertao, J., Xiao, C. Tian, Y. Lu, K.Q. Zhang, C. Liu, 2010. Measuring the sustainability performance of software projects, *In: IEEE Computer Society (Ed.), IEEE 7th International Conference on e-Business Engineering* (ICEBE 2010-Shanghai, China, 2010), 369–373.

[4] Barnett G. 2008. Application Management Research Study, The Bathwick Group, IBM Application Services. Available at:https://www.304.ibm.com/easyaccess/fileserve?contentid=141806.

[5] Capra, E., Formenti, G., Francalanci, C., and Gallazzi, S. 2010. The impact of MIS software on IT energy consumption. In: 18th European Conference on Information Systems, (June 7–9, 2010, Pretoria, South Africa),Available at: http://web.up.ac.za/ecis/ECIS2010PR/ECIS2010/Content/Papers/0073.R1.pdf.

[6] Capra, E., Francalanci, C., and Slaughter, S. 2012. Is software ''green''? Application development environments and energy_efficiency in open source applications. *Information and Software Technology*, 54, 60–71.

[7] Carbon Budget 2010. *Annual update of the global carbon* budget and trends. (Released in 5th December 2011). Available at: http://www.globalcarbonproject.org/carbonbudget/index.htm.

[8] Chen, A.J.W., Boudreau, M. and Watson, R. T. 2008. Information Systems and Ecological Sustainability, *Journal of Systems and Information Technology*. 10, 3, 186-201.

[9] Ciocoiu, C. N. 2011. Integrating Digital Economy And Green Economy: Opportunities For Sustainable Development. *Theoretical and Empirical Studies in Urban Management*, 61, 33-43.

[10] Colosimo, M., De Lucia, A., Scanniello, G. and Tortora, G. 2009. Evaluating legacy system migration technologies through empirical studies. *Information and Software Technology*, 51, 433–447.

[11] CompTIA. 2011. Second annual Green IT insights and opportunities study. Availabe at: http://www.comptia.org/news/pressreleases/11-04-19/Green_IT_Trending_Upward_as_a_Priority_for_Organizations_CompTIA_Study_Finds.aspx.

[12] Dick, M. and Naumann S. 2010. Enhancing Software Engineering Processes towards Sustainable Software Product Design. In: Greve K; Cremers AB (eds) *EnviroInfo 2010*, October 6 - 8, 2010, Cologne/Bonn, Germany. Shaker, Aachen, 706–715.

[13] EITO (European Information Technology Observatory). 2002. *The Impact of ICT on Sustainable Development*, (Frankfurt am Main, Germany), 250-283.

[14] Emmenegger, M.F., Frischknecht, R., Stutz, M., et at. 2006. Life Cycle Assessment of the mobile communication

system UMTS-towards eco-efficient systems. *International Journal of Life Cycle Assesment*, 11, 265-276.

[15] Erlikh, L. 2000. Leveraging legacy system dollars for e-business. *IT Pro.* (May/June 2000), 17-23.

[16] Eugster, M., Hischier, R., and Huabo, D. 2007. Key Environmental Impacts of the Chinese EEE-Industry-A Life Cycle Assesment Study. Empa and Tsinghua Eniversity, St.Gallen and Beijing. Available at: http://library.eawagempa.ch/empa_publications_2007_open_access/EMPASG20070177.pdf.

[17] Gartner Inc. 2008. Green IT: The new industry shockwave. In *Presentation at symposium/ITXPO conference 2008*.Available at: http://www.gartner.com/it/page.jsp?id=503867.

[18] Hichier, R. and Baudin, I. 2010. LCA study of a plasma television device. *The International Journal of Life Cycle Assessment*, 15, 5, 428-438.

[19] Hilty, L. and Lohmann, W. 2011. The five most neglected issues in "Green IT". CEPIS *Upgrade*, 12, 4, 12-15.

[20] Koomey, J. 2011. *Growth in Data center electricity use 2005 to 2010*. Oakland, CA: Analytics Press. Available at: http://www.analyticspress.com/datacenters.html.

[21] Lamb, J. *The Greening of IT: How Companies Can Make a Difference for the Environment*? Boston: IBM Press. 2009.

[22] Mocigemba, D. 2006. Sustainable computing. *Poiesis & Praxis: International Journal of Technology Assessment and Ethics of Science*, 4, 3,163–184.

[23] Molla, A. 2009.Organizational Motivations for Green IT: Exploring Green IT Matrix and Motivation Models, *Pacific Asia Conference on Information Systems* (Hyderabad, India, July 10-12).

[24] Murugesan, S. 2008. Harnessing Green IT: Principles and Practices. *IT Professional*, IEEE Computer Society, January/February, 24–33.

[25] Murugesan, S. 2010. Making IT Green. *IT Professional*, IEEE Computer Society, 12, 2, 4–5.

[26] NASCIO .2008. Digital States at Risk!: Modernizing Legacy Systems, National Survey on Legacy Systems and Modernization in the States. Available at: http://www.nascio.org/publictions/documents/NASCIO-DigitalStatesAtRisk.pdf.

[27] NASCIO .2008. Green IT in Enterprise Practices: The Essential Role of the State CIO. Available at: http://www.nascio.org/publications/documents/NASCIO-GreenIT.pdf.

[28] NASCIO .2006-2009. State CIO Priorities. Available at: http://www.nascio.org/publications/documents/NASCIO-CIOPriorities.pdf.

[29] Naumann, S., Dick, M., Kern, E., and Johann, T. 2011. The GREENSOFT Model: A reference model for green and sustainable software and its engineering. *Sustainable Computing: Informatics and System*, 1, 294– 304.

[30] Robert, C. S., Plakosh, D. and Lewis G.A. 2003. *Modernizing Legacy Systems: Software Technologies, Engineering Processes and Business Practices*. Carnegie Mellon, The SEI Series in Software Engineering, Pearson Education Inc. Boston MA.

[31] Saarelainen, M. M., Ahonen, J.J., Lintinen, H., Koskinen, J., Kankaanpää, I., Sivula, H., Juutilainen, P., and Tilus, T. 2006. Software Modernization and Replacement Decision Making in Industry: A Qualitative Study, *Evaluation and Assessment in Software Engineering*.

[32] Scharnhors, W., Hilty L.M., and Jolliet, O. 2006. Life Cycle Assessment of Second Generation (3G) Mobile phone networks. *Environment International*, 5, 32, 656-675.

[33] Taina, J .2010. How green is your software? *Software Business*, 51, 1, 4, 151-162.

[34] Unhelkar, B. 2011.Green IT: The Next Five Years. *IT Pro*fessional, March /April, IEEE Computer Society, 56-59.

[35] Viaro, A.T. and Vaccaro G.L.R. 2010. Conceptual Framework to Develop Green It - Going Beyond The Idea Of Environmental Sustainability. *XVI International Conference On Industrial Engineering And Operations Management, Challenges and Maturity of Production Engineering: competitiveness of enterprises, working conditions, environment* (São Carlos, SP, Brazil, 12 to 15 October – 2010). Available at: http://www.abepro.org.br/biblioteca/enegep2010_TI_ST_123_796_16387.pdf.

[36] Xu, L., Ziv, H., Alspaugh, T. and Richardson, D. 2006. An architectural pattern for non functional dependability requirements, *Journal of Systems and Software*, 1370–1378.

[37] Yin, R. K. 2009. *Case study research: design and methods, 4th ed.* (Applied social research methods v.5), SAGE Publication Inc., Thousand Oaks CA.

[38] Zhu, M., Luo, X., Chen, X. and Wu, D.D. 2012. A non-functional requirements tradeoff model in Trustworthy Software. *Information Sciences*, 19, 61–75.