International Journal of Advance Engineering and Research Development
Scientific Journal of Impact Factor (SJIF): 4.72
Special Issue SIEICON-2017,April -2017
e-ISSN : 2348-4470 p-ISSN : 2348-6406

Green in Software Engineering – Efficiency Metrics

Pinkal Shah[1], Nidhi Shah[2]

[1] Asst. Prof. Department of Information Technology, Sigma Institute of Engineering
[2] Asst. Prof.Department of Information Technology, Sigma Institute of Engineering

Abstract *— Green software engineering is an important software engineering process in the today's world. In past developers were concentrating on aim an objective of software and hardware rather than sustainability. There was no focus on technical, social and sustainability of the environment. In US, Canada or we can say developed countries researches are mainly focused on Green IT. But developing countries like India and china there is no focus set on green engineering, so more research works are to be carried on Green software engineering. The developing countries are working or creating more software applications and Information and Communication Technology (ICT) products alone for the developed countries, but Green IT software engineering process is a main and going to be major issue that is to be solved to make efficient software models. The development of energy efficient software requires metrics, which measure the software's energy consumption as well as models to monitor and minimize it. This paper represents data about Green IT and the sustainability of software engineering approach in the life cycle models for software development process. This work surveys the existing software models and their impact on environments.*

Keywords-*Software Development Life Cycle, Information and Communication Technology (ICT), Green Software, Energy Efficiency, Metrics*

I. INTRODUCTION

Sustainability is becoming more important worldwide, supported by several initiatives with wide media coverage. The conference was on Sustainable Development. In Conference it was approved on an agreement entitled 'The Future We Want', and it was stated that 'We understands the vital role of technology as well as the importance of promoting innovation. We request countries, as appropriate, to create enabling frameworks that promote environmentally sound technology, research and development and innovation'. As far as the IT sector is concerned, it contributes about 2 % of the global carbon dioxide emissions and is responsible for approximately 8 % of the electricity use. At last, the startup that promotes respect for the environment by means of ICT, IT, software, etc., are called Green IT. The problem is that, for each and every discipline or fresh discipline, there is no clear map of concepts and definitions. As [1] points out, however, the fact is that Green IT is not only a trend; it had become a necessity as more and more organizations are implementing some form of sustainable solutions.

Activities in the field of Green IT have set certain standards and policies for ICT. The developments in Green IT, which primarily focus on computer physical parts, led to the understanding that "The most focus on strategies to aspect of energy efficient computing and it will be done by evaluation of software" [2, p.58]. This represents a method to develop and apply metrics and measurement methods to measure the energy consumption that is directly related to the software. These standards here should enable developers to measure and monitor the energy consumption of the software during the development process. It seamlessly integrates into the Model (Figure 1).

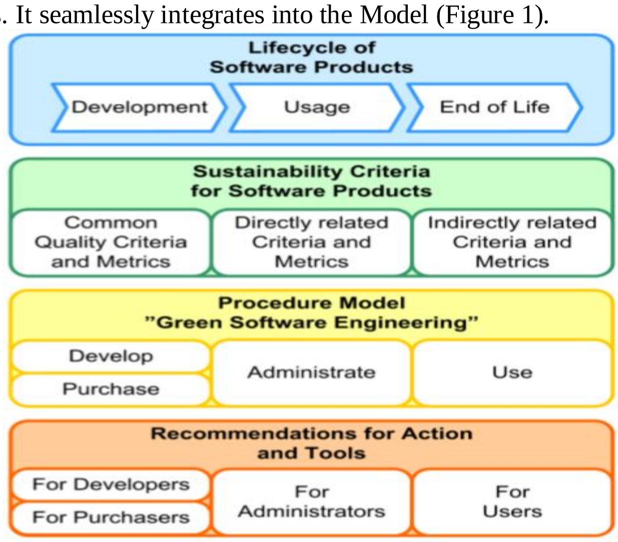

Figure 1. The Model

II. SOFTWARE SUSTAINABILITY AND MEASURMENT

2.1 Software Engineering Sustainability

Sustainability should be taken into account from the initial stages of software development. That is not approachable easily because of working method of developers. Moreover, there is little guidance on Green software engineering [5]. Here we have considered five dimensions of sustainability that are important for the analysis of software systems:

- Individual: For maintenance of the private good of individual human capital. Education, Health, Knowledge, Skills and access to services constitute human capital [4].
- Social: For preserving the solidarity of societal communities and maintaining social capital. Social capital is investments and services that create the basic framework for society [4].
- Economic: This aims to maintain assets. Assets include values too. It define income as the "capital one can consume during a time and still be as well off at the end of the project, as it transfer on consuming added value (interest), rather than capital" [4].
- Environmental: This get to improve welfare of citizen by protecting resources such as land, water, minerals ,air minerals and ecosystem services. Environment includes the sources of raw materials used for human needs [5].
- Technical: This has the central objective of long usage of systems and their timely evolution with changing environment and requirements.

2.2 Measuring the Energy Consumption

There are available approaches for measuring software energy consumption, such as black box and White box measurements. Here measure a system using a black box and can explain on how the entire system (Hardware and software) performs. When it comes to measuring software, one cannot apply these breakpoints or patterns because each of them is customized for one specific group of tasks. One approach [6] states that prepare or apply individual scenarios for similar kind of software and then measure energy consumption of that software. Here in Black Box measurement does not allow to look into code and this represents energy consumption of entire software whereas white box measurements can tell which part is potential for energy savings. A white box method allow to access codes so this technique is better suited to find resource intensive parts of programs and to improve them.

2.3 Sorting Application

Since sorting is a approachable task in computer, and is easy to find the useful work quickly. For Example: - For sorting any items we use formula as SortedItems/UsedEnergy. For comparison we are going to compare 2 different sorting algorithms 1. Bubble Sort Technique with complexity of $O(n^2)$ and 2. Heap Sort Technique with complexity of O(n.log(n)). This both techniques are used to sort randomly generated numbers using an array. For the instrumentation, two counters are used. One counter as it counts the items that are already sorted, the other is used as a phase counter to generate a positive edge when sorting starts and a negative edge when sorting ends.

Bubble sort used nested for loops in outer loop works on arrays from 0 to n with increment in counter item gets sorted on each loop. Heap sort is also similar with transforming array into binary heap and then sort the item with increment in counter on each loop. Since the system has to performed several background processes and services, which cause some peaks in the processor time (red) and in the energy consumption (green), which was measured average idle consumption which is about 65 Watt. Then the actual measurement of the algorithms with bubble sort is started, which is shown in Figure 2. Bubble sort algorithms sort 100,000 double values. The Phase (Blue) counter represents the start of sorting with positive edge, which suddenly stresses the CPU (Red) and hiked the consumed energy (green). When all items are sorted (black), the phase counter generates the negative edge which is the end of the measurement. The sorting time was 55 seconds and the average energy consumption was 50 Watts[7].

$$50W \times 55s = 2750Ws = 0.76 = 2750J$$

The whole algorithm consumed 5,400 Joule for the sorting of 100,000 double values, which results in

$$100{,}000/10{,}800J = 18\times(SortedItems/Joules)$$

The Heap Sort algorithm also sorted 100,000 double values and the result is shown in Figure 2, where the algorithm worked as per expectation and faster. The values were sorted in only 5 seconds. Due to the resolution (1 second) the power consumption line (green) did not immediately follow the phase counter, but here the average power consumption during processing was 50 Watts,

$$50W \times 5s = 250Ws = 0.07Wh = 250J.$$

For a better results it is normal to give the algorithms the same time frame and calculate the power consumption of the idle time of the faster algorithm and add it to the total power consumption. Since the idle time is 50−5 = 45, and the idle power consumption is 37W, the power consumption is 37W × 45s = 1665Ws = 0.46Wh = 1665J. Adding the energy consumed during sorting the result is 2500 J.

100,000 /(250J + 2500J) = 36.4 (SortedItems/Joule)

Algorithm with better run time tends to also have better energy efficiency, which states than higher performance consumers higher energy.

2.4 Multi User Web Applications

Usually there are multi user applications such as browser, web-server and database. Therefore, we chose a interactive system between client/server application. Purpose of the application is to collect, retrieve and visualize consumption data of the campus buildings [8]. The server stores the data in a database that is managed by a web interface. For demonstration purposes only a cut-out of the software is used.

The consumption and measurement load will be analyzed by decides number of users. They will send multiple numbers of requests to the server and because of this source code should calculated at the point of entry and exit of requests. The count should vary with incoming and outgoing of requests. Overall we should know how many requests are being processed currently. Here we will try to depict energy consumption or change in efficiency while processing different amount of users.

In a first dry run a load of 15 users was generated. The result is shown in Figure 3. After the completion period the users that are in pipeline for their response (blue) even out at six users. In this stage the average power rating of the server is about 110 Watts (green), while simultaneously serving 15 users with a waiting time way less than one second. In one hour the system with 15 simultaneous users consumes 110 Wh = 396kJ. The same calculations is done with 25 one on one served users (Figure 3), whereas the pipelined users are 17 approximately, It means average waiting time is more compared to 15 users despite this they are all getting served in less than 1 second. The energy consumed is also 110 Watts and users can still be served at a satisfactory time. As one can see, the claimed response time plays a key role: the higher the demand for fast response, the lower is the energy efficiency.

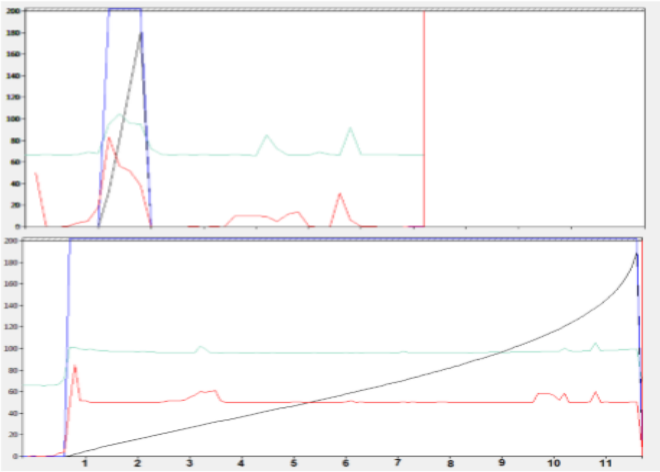

Figure 2. Energy Consumption of Heap Sort (top) and Bubble Sort (Bottom)

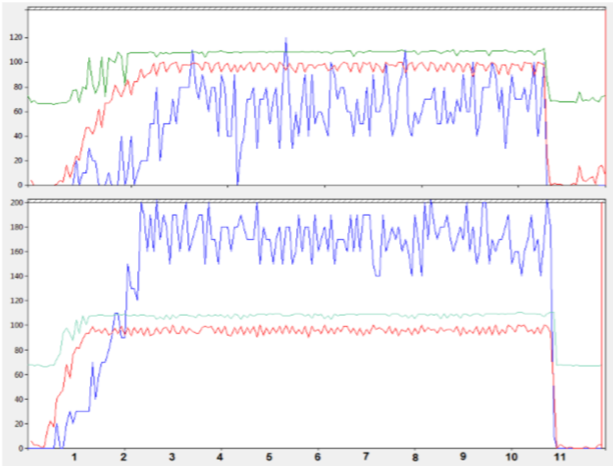

Figure 3. Energy Consumption of Serving 15 (top) and 25 (Bottom) users.

IV. CONCLUSION

Our task is to raise knowledge and awareness among software developers. Moreover who holds power of choosing and demanding for software which are more respectful for climate (Environment). If we achieve any result then entire community will start developing greener software. This paper concludes that it is possible to measure energy consumption using white box method[9]. Due to high affinity of method of performance measurement, It is work load can be re-used. It is possible to combine energy efficiency and performance in order to continuous integration paradigms. This leads to optimization of databases of software product. The amount of publications in field of Green IT or sustainable software development will result demand for metrics to measure aspects of Green IT. Eventually it will show importance of guidelines for developers.

In the future, Measurement methods should be centralized and results should be aggregated to have more efficient power models [10]. Developers will be able to use this result for consumption of software. In combination with power models, developers or analyzers should be able to give accurate prediction of power consumptions. Based on robust metrics and measuring methods, the development of seals of quality will be an important step for sustainable software.

REFERENCES

[1] Brodkin J, “Economy driving Green IT initiatives”. Netw World 25(48):16, 2008

[2] Brown David,Reams Charles, “Toward Energy- Efficient Computing: What will it take to make server-side computing more energy efficient?” Communications of the ACM, vol. 53, no. 3, pp. 50–59, 2010

[3] Penzenstadler B, Femmer H (2013), "A generic model for sustainability with process and product-specific instances". ACM, Fukuoka, Japan, pp 3–8, 2013

[4] Wiley,Goodland R, "Sustainability: human, social, economic and environmental, Encyclopedia of global environmental change", 2002

[5] Mocigemba D, Sustainable computing. Poiesis & Praxis Int J Technol Assess Ethics Sci 4(3):163–184,2006

[6] M. Dick, E. Kern, J. Drangmeister, S. Naumann, and T. Johann, "Measurement and Rating of Software-induced Energy Consumption of Desktop PCs and Servers," in Innovations in sharing environmental observations and information, W. Pillmann, S. Schade, and P. Smits, Eds. Aachen: Shaker, 2011, pp. 290–299.

[7] S. Naumann, M. Dick, E. Kern, and T. Johann, "The GREENSOFT Model: A Reference Model for Green and Sustainable Software and its Engineering," Sustainable Computing: Informatics and Systems, vol. 1, no. 4, pp. 294–304, 2011. Available: doi:10.1016/j.suscom. 2011.06.004

[8] J. Tayeb, K. Bross, B. S. Chang, L. Cong, and S. Rogers, "Intel Energy Checker Software Development Kit User Guide," 2010-12-15. [Online]. Available: http://software.intel. com/file/32957

[9] ISO/IEC/IEEE 24765 (2010) Systems and software engineering.

[10] Bourque, P. and Fairley," Guide to the Software Engineering Body of Knowledge". R.E. (eds.), IEEE,2014

