

# Identification of Practices for Green Software Development in Multi-Sourced Project: A Systematic Literature Review Protocol

Muhammad Salam  
Department of Computer Science and IT  
University of Malakand, Lower Dir, Pakistan

Siffat Ullah Khan  
Department of Computer Science and IT  
University of Malakand, Lower Dir, Pakistan

## ABSTRACT

**Context:** Green software engineering (GSE) is a research area gaining reputation to develop green and sustainable software. Whereas software multi-sourcing is the allocation of software development processes to external (offshore) multiple vendors in order to reduce cost, improve quality, and minimize the development time. Green software multi-sourcing is a recent research area which lies at the intersection of both the aforementioned research areas.

**Objective:** The objective of this study is to develop systematic literature review (SLR) protocol to identify practices/solutions to assist software multisourcing vendors in the development of green and sustainable software.

**Method:** We have used systematic literature review (SLR) process as research methodology and have developed the SLR protocol.

**Outcome:** The expected outcomes of the implementation of SLR protocol will be the identification of solutions/practices for multisourcing vendors in the development of green and sustainable software in multi-sourced projects.

## Keywords

Green Software Multi-sourcing; SLR Protocol; practices/solutions, Vendors

## 1. INTRODUCTION

This section presents a review of the various research related to green and sustainable software development. We have also considered state-of-the-art research trends. The objective is to summarize and discuss the results of each study in order to better understand the problem context.

Green and Sustainable Software has been defined in the literature as “The 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” [1, 2]. Where Green and Sustainable Software Engineering is the art of developing green and sustainable software engineering process [3]. Various components of green and sustainable software are shown in Figure 1.1. Currently many efforts have been done in obtaining green and sustainable software in general context [2, 4-8]. Software which has longer life time is considered as sustainable software. According to [9] the term sustainable applies to both longer life and greener aspects of software. Our focus in this thesis is specific to the development of green and sustainable software in multi-sourced software projects, which is explained in the subsequent sections. Some associated broad areas terms with Green Software are Green Computing, Green IT, Green Hardware, and Green Infrastructure. In order to transfer the general concept of green

and sustainable into the computer systems (hardware as well as software) the term green or sustainable computing is used [10, 11]. Green computing can be defined as the practice of maximizing the efficient use of computing resources to minimize negative impact on environment [12, 13].

In other words, green computing or green IT is the study and practice of designing, manufacturing, using and disposing computing resources efficiently and effectively with minimal environment damage [14]. In [15], the authors define green computing as “environmentally sustainable computing which studies and practices virtually all computing efficiently and effectively with little or no impact on the environment”. According to Tushi and Bonny [16] “green computing is the practice of implementing policies and procedures that improve the efficiency of computing resources in such a way as to reduce the energy consumption and environmental impact of their utilization”. The current literature defines the term green IT in variety of ways. According to C. Calero and M. Piattini [10] the term green IT denotes the connection between energy efficiency and IT. In [10, 17], the term green IT has been defined as “the study and practice of designing, manufacturing, and using computer hardware, software, and communication systems efficiently and effectively with no or minimal impact on the environment”.

According to [18] the goal of green IT is to yield as less possible waste throughout the lifecycle of IT including (development, operation and disposal). The best use of ICTs to manage enterprise activities in eco-friendly manner comprising its product, services and resources throughout their life cycle is the focus of green IT. The principal objective of green IT approach is to decrease the energy consumption, uphold the operational costs and minimize environmental impacts [16]. Though, it is noteworthy that there are two aspects of Green IT. Primarily, IT can be the reason of ecological problems, and otherwise IT can be used to resolve ecological problems [16, 19]. According to Donnellan et al., [20] the terms green IT and green computing are same. Two key paybacks of green IT are, ecological benefits and cost saving benefits. According to S.S.Shenoy and R. Eeratta [6] infrastructure plays an important role in the development of software in eco-friendly manner i.e. use meeting rooms with natural lights, avoid the of air-conditioners, minimize the travelling and use modern ICTs tools, establish paperless offices, use the concept of cloud computing in the software development houses. Till recently, the larger portion of the efforts done in the era of Green IT was linked to hardware, concentrating mostly on improving the hardware energy efficiency. Thus it is obvious that research work needs to focus on the software aspect of Green IT as well [10, 18]. The tendency has been changed in the last few years, and research on the new theme of ‘Green Software’ is emerging. In this study we have focused on the

development of green software in multi-sourced software projects from vendor's perspective.

It is argued that the existing research mainly focuses on hardware-oriented solutions for environmental sustainability. However, few have focused on software sustainability [21].

Literature shows that some researchers have worked on the various aspects of green software and highlighted the importance of green software development in general [22-24]. These studies are reported as follow:

Intiaz and Mahmoud [3] have presented a green model with two levels, for sustainable software engineering, the first level presents guidelines for green software development. While the second level is comprises of approaches taken by software itself to subsidize to green software development. Furthermore the authors have classified how each software engineering phase could be ecologically sustainable through green processes and guidelines hence ending up with a green/sustainable software product. Finally the authors have suggested the relevant metrics for measuring the greenness of each phase of the software development. The authors have categorized the identified approaches/concepts into five categories. Lastly, the authors have related both levels to each other for the development of green and sustainable software.

K. Erdelyi [18] has presented sustainability guidelines for developing sustainable software/green software. The study has suggested that throughout the software development life cycle the various activities should be examined in order to support ecological sustainability. The author father recommended that both software and software development should support environmental sustainability. The study has also proposed the need for further investigations on the green and sustainable software development.

GREENSOFT model for sustainable software has been proposed by S. Naumann et al., [2], the model supports software developers, administrators, and software users in creating, maintaining, and using software in a more sustainable way. This model has covered the software development life cycle from cradle-to-grave. GREENSOFT model also supports sustainable software engineering methods, sustainable software design and development, as well as suitable supervision. The final module of the model comprises tools and recommendations to support stakeholders (developers, administrators and users) with different professional skill level in applying green and sustainable methods in general.

Shenoy and Earatta [6] have offered green software development model along with some guidelines and practices that support sustainability. The authors have suggested changes in the existing software development life cycle (SDLC) and have proposed appropriate guidelines which support green and sustainable software development, products, processes and technologies.

R. Ahmad et al., [25] have conducted a systematic literature review on sustainable development in software engineering to identify the research trends, limitations, approaches, methods, and available studies in this area. The authors argue that few studies are available on sustainability in software engineering whereas the concept of sustainability is very vigorous in other disciplines. The authors have also suggested future work to investigate the evaluation method for measuring sustainability in software engineering.

However, little attention has been paid to the development of green software in multi-sourced projects. Multi-sourcing is a modern paradigm in software outsourcing domain which offers the benefits of using multiple vendors for the development of software in a shorter time span. The outsourced software is developed in modules/components by various software vendors to expedite the development process [26]. Consequently, Green Software Multi-sourcing has developed a new area of research interest in Global Software Engineering domain in order to develop green and sustainable software in multi-sourced software development projects. Research in this area will assist multi-sourcing industry to deliver green software in general and to gauge vendors' readiness for the development of greener software in particular. For this purpose we have proposed Green Software Multi-sourcing Readiness Model. The proposed structure of the model has been published [27]. In this paper we have presented our systematic literature review (SLR) protocol. The preliminary results of the SLR protocol are also presented.

## **2. RESEARCH METHODOLOGY**

For the identification of green and sustainable software practices, we followed the systematic literature review guidelines [28]. Development of SLR protocol is the first phase of SLR methodology and has been discussed in the subsequent sections.

## **3. SYSTEMATIC LITERATURE REVIEW PROTOCOL**

The Systematic Literature Review (SLR) protocol is a set of tasks that have to be performed in order to answer the research question listed in the sub-section 3.1. Our review protocol is made of six components i.e. research questions and constructing search terms, literature search strategy, literature selection criteria, literature quality assessment, Data extraction, data synthesis. Each component is described in the following sections. SLR is considered to be a prominent research method in the field of software engineering and Computer Science [29-31]. We have followed all the SLR guidelines [28] by initially developing the SLR protocol as we did in our previous work [32].

### **3.1 Research Question**

The core objective of this research study is to find out state-of-the-art practices for green and sustainable software development in multi-sourced software projects. In order to achieve this goal; we outlined the following research question (RQ):

*RQ1:* What are the practices, as identified in the literature, to be adopted by software multi-sourcing vendor organizations which can contribute towards the development of green and sustainable software?

### **3.2 Constructing Search Term**

The following details will help in designing a search term relevant to our research questions.

Population: Software Multi-sourcing Vendor Organization

Intervention: practices/solutions

Outcomes of relevance: Practices/Solutions for Green Software Development. This will provide input to the development of our proposed, Multi-sourcing Readiness Model (GSMRM) [33].

Experimental Design: Systematic literature review, empirical studies, theoretical studies, case studies, expert’s opinions.

### 3.3 Search strategy and search

The following search strategy is used for the construction of search terms.

- a) Use the Research Questions for the derivation of major terms, by identifying population, intervention and outcome;
- b) For these major terms, find the alternative spellings and synonyms;
- c) Verify the key words in any relevant paper;
- d) Use of Boolean Operators for conjunction if the database allows, in such a way, to use ‘OR’ operator for the concatenation of alternative spellings and synonyms whereas ‘AND’ for the concatenation of major terms.

We have designed the following two search strings for searching our selected online digital libraries.

$\lambda_1$ : (“Green software” OR “sustainable software”) AND (“practices” OR “solutions” OR “techniques”) AND (“multi-sourcing”)

$\lambda_2$ : (“Green software” OR “sustainable software”) AND (“practices” OR “solutions” OR “techniques”)

Where  $\lambda_1$  denotes search string to retrieve empirical studies regarding the practices for the development of green software multi-sourced software projects, while  $\lambda_2$  denotes search string to retrieve empirical studies regarding the practices for the development of green software in general context. The results of search string ( $\lambda_1$ ) were very poor and almost negligible as shown in Table 1. Consequently we decided, to implement search string  $\lambda_2$  after detailed discussions with experts of the software engineering research group (SERG-UOM) at the university, to implement search string  $\lambda_2$ . The search results of  $\lambda_2$  are showed in Table 2. The practices, identified through the SLR (using search string  $\lambda_2$ ), will be validated through empirical studies in multi-sourcing software industry in order to know whether these findings are applicable specifically, or can be adopted, in software multi-sourcing environment. Moreover limited numbers of empirical research studies have been conducted in the context of GSD in general and software multi-sourcing in particular [34].

**Table 1  $\lambda_1$  results**

Digital Libraries	Total publication Found
Science Direct	00
ACM	00
IEEE Xplore	00
Springer Link	02
Google Scholar	14
<b>Total</b>	<b>14</b>

**Table 2  $\lambda_2$  results**

Digital Libraries	Total publication Found
Google Scholar	429
ACM	164
IEEE Xplore	114
Springer Link	149
Science Direct	16
<b>Total</b>	<b>872</b>

### 3.4 Resources Searched

The following digital libraries and databases have been searched, using  $\lambda_2$  as a search strings.

- Google Scholar
- ACM
- IEEE Xplore
- Springer Link
- Science Direct

### 3.5 Publication Selection

The main purpose of the publication selection procedure is to choose only those search results which are relevant to our research questions.

The following inclusion and exclusion criteria have been used for the relevant literature selection.

#### 1) Inclusion Criteria

- Research studies and articles that are related to our research questions.
- Research work that describe green software in multi sourcing environment/General context.
- The research papers/ articles /books/review paper is in English Language.
- The article/paper is available in full text.
- Research papers that describes success factors, risk factors and practices of green software development in multi-sourcing/General context.

#### 2) Exclusion Criteria

- Article/papers/books etc. that do not fulfill inclusion criteria as mentioned above will be excluded.

### 3.6 Publication Quality Assessment

The main purpose of quality assessment is to check and assess the quality of finally selected papers. The quality checklist contains the following questions:

- a. Is the objective of the research is clearly defined?
- b. Is the conclusion of the research is related to the objective of the research?
- c. Whether the term green software in multi sourced projects/General software development context is discussed clearly?
- d. Is it clear how the practice/solution was identified?
- e. Each of the above questions will be marked as ‘YES’, ‘NO’, or ‘N.A’.

Apart from the above quality criteria we will also use the following quality criterion, for further analysis, which is based on our own experiences. We have used the same criteria in our previous research [35].

*Criteria for A-Quality papers:* In this category we list those papers which fulfill the following criteria's:

- Paper published in impact factor Journal
- Having clear methodology
- Having sample size of the following condition
  - a. Case Study  $\geq 3$
  - b. Interviews  $\geq 12$
  - c. Survey  $\geq 50$

d. Literature Review  $\geq 50$

*Criteria for B-Quality papers:* In this category we list those papers which fulfill the following criteria's:

- Paper published in well reputed conference
- Having clear methodology
- Having sample size of the following condition
  - a. Case Study = 2
  - b. Interviews:  $\geq 5$  and  $\leq 11$
  - c. Survey:  $\geq 30$  and  $\leq 49$
  - d. Literature Review:  $\geq 30$  and  $\leq 49$

*Criteria for C-Quality papers:* In this category we list those papers which fulfill the following criteria's:

- Experienced reports/ articles, published in less reputed venues (Journal, Conference)
- Having clear methodology
- Having sample size of the following condition
  - a. Case Study: 1
  - b. Interviews  $\leq 5$
  - c. Survey:  $\geq 1$  and  $\leq 29$
  - d. Literature Review:  $\geq 1$  and  $\leq 29$

### 3.7 Data Extraction Process

The process of extracting data from finally selected relevant literature is called data extraction. The core aim of our protocol is to extract the data which satisfies our research question. A data extraction form has been generated to extract data from each of the finally selected paper. The data extraction form contains attributes, such as, Date of review, paper id, paper title, methodology, library, practices/solutions etc. The extraction will be commenced by the primary researcher (the first author), who will be responsible for the data extraction. A secondary reviewer/supervisor (the co-author) will be approached for guidance in case of an issue regarding the data extraction. The secondary reviewer will select few publications randomly from the list of publication already chosen by the primary reviewer. The secondary reviewer will independently extract data from the randomly selected publications. The results will then be compared with the results produced by the primary reviewer. We have used similar approach in previous research [36].

### 3.8 Data Synthesis

The analysis of extracted data from finally selected papers is called data synthesis. For the RQ1, the data will be synthesized by creating one summary table having the columns (*S.No, Factors/practices, Frequency, Percentages*) showing the list of all the factors along with their frequencies and percentages.

## 4. RESULTS OF SYSTEMATIC LITERATURE REVIEW PROTOCOL

The protocol preliminary search results are shown in Table 3. The protocol has retrieved total  $N=872$  research papers. From which  $N=123$  were selected initially by reading title and abstract of the papers. Finally we selected 44 research papers. After merging the papers of our previous SLR protocol [37] we got a final list of  $N=102$  research papers. These results are

shown in Table 3. We are now in the process of data extraction. The expected outcomes of the SLR will be the identification of practices/solutions that can contribute vendors towards the development of green software in multi-sourced projects.

**Table 3 preliminary results of the protocol**

Digital Libraries	Total publication Found	Initial selection	Final selection
Google Scholar	429	47	19
ACM	164	29	12
IEEE Xplore	114	20	04
Springer Link	149	23	06
Science Direct	16	04	03
Total	872	123	44
Total papers through previous SLR: 58			
Total papers through SLR-P: 44			
Final sample size after merging SLR results with SLR-P: (N=102)			

## 5. CONCLUSION AND FUTURE WORK

Software development multi-sourcing is emerging global software engineering (GSE) paradigm for producing quality software at minimum cost and time, by hiring more than one software vendors. Consequently, green software multi-sourcing has raised a novel research area in the context of GSE domain which aims to produce greener software in multi-sourced software development projects. Research in this area will assist multi-sourcing industry to deliver green software in general and to gauge multisourcing vendors' readiness for the development of greener software in particular. In this paper we have presented our SLR protocol to identify practices/solution for green software development in multi-sourced projects. During implementation of the protocol, we have identified ( $N=872$ ) research papers out of them ( $N=123$ ) has been selected initially by reading title and abstract of the papers. We have shown the search results of the protocol in Table 3. We are now in the process of data extraction. The expected outcomes of the SLR will be the identification of practices/solutions that can contribute towards the development of green software in multi-sourced projects. Our ultimate goal is to develop Green Software Multi-Sourcing Readiness Model (GSMRM) [33].

## 6. REFERENCES

- [1] M. Dick, S. Naumann, and N. Kuhn, "A Model and Selected Instances of Green and Sustainable Software," in *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience* SE - 24, IFIP Advances in Information and Communication Technology, J. Berleur, M. Hercheui, and L. Hilty, Eds., 328 ed, 2010, pp. 248-259 LA - English.
- [2] 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, pp. 294-304, 2011.
- [3] S. S. Mahmoud and I. Ahmad, "A Green Model for Sustainable Software Engineering," *International Journal*

- of Software Engineering and Its Applications, vol. 7, 2013.
- [4] S. Naumann, E. Kern, and M. Dick, "Classifying Green Software Engineering - The GREENSOFT Model," in 2nd Workshop EASED@BUI5 2013 — Energy Aware Software-Engineering and Development — Proceedings: Oldenburg Lecture Notes on Software Engineering (OLNSE) Germany, 2013, pp. 13-14.
- [5] V. K. Singh, K. Dutta, and D. VanderMeer, "Estimating the Energy Consumption of Executing Software Processes," presented at Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), 20.
- [6] S. S. Shenoy and R. Eeratta, "Green software development model: An approach towards sustainable software development," presented at India Conference (INDICON), Annual IEEE, 2011.
- [7] G. Sissa, "Green software," UPGRADE: The European Journal for the Informatics Professional, vol. 11, pp. 53-63, 2010.
- [8] P. Lago, R. Kazman, N. Meyer, M. Morisio, H. A. M ller, and F. Paulisch, "Exploring initial challenges for green software engineering: summary of the first GREENS workshop, at ICSE 2012," ACM SIGSOFT Software Engineering Notes, vol. 38, pp. 31-33, 2013.
- [9] S. Ray, N. Sengupta, K. Maitra, K. Goswami, S. Agarwal, and A. Nath, "Green Software Engineering Process: Moving Towards Sustainable Software Product Design," Journal of Global Research in Computer Science, vol. 4, pp. 25-29, 2013.
- [10] C. Calero and M. Piattini, Green in Software Engineering: Springer, 2015.
- [11] M. SamikshaRavindraSuryawanshi and M. M. M. Chavan, "Green Computing: An essential trend for secure future.," presented at Proceedings of National Conference on Emerging Trends: Innovations and Challenges in IT, 2013.
- [12] Y. Cai, "Integrating sustainability into undergraduate computing education," in Proceedings of the 41st ACM technical symposium on Computer science education. Milwaukee, Wisconsin, USA: ACM, 2010, pp. 524-528.
- [13] R. R. Harmon and N. Auseklis, "Sustainable IT services: Assessing the impact of green computing practices," presented at International Conference on Management of Engineering & Technology, PICMET, 2009.
- [14] S. Cai, X. Chen, and I. Bose, "Exploring the role of IT for environmental sustainability in China: An empirical analysis," International Journal of Production Economics, vol. 146, pp. 491-500, 2013.
- [15] C.-T. D. Lo and K. Qian, "Green computing methodology for next generation computing scientists," presented at Computer Software and Applications Conference (COMPSAC), 2010 IEEE 34th Annual, 2010.
- [16] B. T. Tushi, "An archival analysis of green information technology: The current state and future directions," Queensland University of Technology 2015.
- [17] S. Murugesan, "Making IT green," IT professional, pp. 4-5, 2010.
- [18] K. Erd lyi, "Special factors of development of green software supporting eco sustainability," presented at Intelligent Systems and Informatics (SISY), IEEE 11th International Symposium, 2013.
- [19] K. Nanath and R. R. Pillai, "A Sustainability Model of Green IT Initiatives," 2012.
- [20] B. Donnellan, C. Sheridan, and E. Curry, "A capability maturity framework for sustainable information and communication technology," IT professional, vol. 13, pp. 33-40, 2011.
- [21] K. Sierszecki, T. Mikkonen, M. Steffens, T. Fogdal, and J. Savolainen, "Green Software: Greening What and How Much?," Software, IEEE, vol. 31, pp. 64-68, 2014.
- [22] N. S. Chauhan and A. Saxena, "A Green Software Development Life Cycle for Cloud Computing," IT Professional, vol. 15, pp. 28-34, 2013.
- [23] S. Dustdar, F. Li, H.-L. Truong, S. Sehic, S. Nastic, S. Qanbari, M. Vogler, and M. Claesens, "Green software services: From requirements to business models," presented at Green and Sustainable Software (GREENS), 2013 2nd International Workshop on, 2013.
- [24] R. U. Khan and S. U. Khan, "Green IT-Outsourcing Assurance Model," presented at Global Software Engineering Workshops (ICGSEW), IEEE 8th International Conference, 2013.
- [25] R. Ahmad, F. Baharom, and A. Hussain, "A Systematic Literature Review on Sustainability Studies in Software Engineering," KMICe2014, Universiti Utara Malaysia, 2014.
- [26] H. S. Kehal and V. P. Singh, Outsourcing and offshoring in the 21st Century: A Socio-Economic Perspective.
- [27] M. salam and S. U. khan, "GREEN SOFTWARE MULTI-SOURCING READINESS MODEL (GSM-RM) FROM VENDOR'S PERSPECTIVE," Sci.Int.(Lahore), vol. 26, pp. 1421-1424, 2014.
- [28] B. Kitchenham and C. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering, Keele University and Durham University Joint Report," EBSE 2007-001, 2007.
- [29] J. I. M. de Souza, V. r. M. Silva, D. N. Prata, and P. Letouze, "A Further Analysis of the Importance of Systematic Reviews to Computer Science," International Journal of Social Science and Humanity, vol. 5, pp. 682, 2015.
- [30] M. Staples and M. Niazi, "Experiences Using Systematic Review Guidelines," Journal of Systems and Software., vol. 80, pp. 1425-1437, 2007.
- [31] Y. I. Alzoubi, A. Q. Gill, and A. Al-Ani, "Empirical studies of geographically distributed agile development communication challenges: A systematic review," Information & Management, vol. 53, pp. 22-37, 2016.
- [32] M. Yaseen, S. U. Khan, and A. U. Alam, "Software multi-sourcing risks management from vendors' perspective: A systematic literature review protocol," Gomal University Research Journal, Pakistan, vol. 29, 2013.
- [33] M. Salam and S. U. Khan, "GREEN SOFTWARE MULTI-SOURCING READINESS MODEL (GSM-

- RM) FROM VENDORS'S PERSPECTIVE," *Science International*, vol. 26, 2014.
- [34] R. Sriram and S. K. Mathew, "Global software development using agile methodologies: A review of literature," presented at Management of Innovation and Technology (ICMIT), IEEE International Conference, 2012.
- [35] I. Ali and S. U. Khan, "USABILITY OF AGILE METHODS IN SOFTWARE DEVELOPMENT OUTSOURCING: A SYSTEMATIC LITERATURE REVIEW PROTOCOL," *Sci.Int.(Lahore)*, vol. 5, pp. 2023-2027, 2014.
- [36] S. U. Khan, M. Niazi, and N. Ikram, "Software Development Outsourcing Relationships Trust: A Systematic Literature Review Protocol," presented at Evaluation and Assessment in Software Engineering, EASE 2010, Keele University, UK, 2010.
- [37] M. Salam and S. U. Khan, "Systematic Literature Review Protocol for Green Software Multi-sourcing with Preliminary Results," *Proceedings of the Pakistan Academy of Sciences*, vol. 52, pp. 285-300, 2015.