



A systematic literature review on energy efficiency in cloud software architectures



Giuseppe Procaccianti^{a,b,*}, Patricia Lago^a, Stefano Bevini^c

^a VU University Amsterdam, De Boelelaan 1081a, Amsterdam, The Netherlands

^b Politecnico di Torino, Corso Duca degli Abruzzi, 24, Torino, Italy

^c IBM Rome Software Lab, Via Sciangai 53, Roma, Italy

ARTICLE INFO

Article history:

Received 18 February 2014

Received in revised form 30 October 2014

Accepted 16 November 2014

Keywords:

Cloud computing

Energy efficiency

Green software

Software architecture

ABSTRACT

Cloud-based software architectures introduce more complexity and require new competences for migration, maintenance, and evolution. Although cloud computing is often considered as an energy-efficient technology, the implications of cloud-based software on energy efficiency lack scientific evidence. At the same time, energy efficiency is becoming a crucial requirement for cloud service provisioning, as energy costs significantly contribute to the Total Cost of Ownership (TCO) of a data center. In this paper, we present the results of a systematic literature review that investigates cloud software architectures addressing energy efficiency as a primary concern. The aim is to provide an analysis of the state-of-the-art in the field of energy-efficient software architectures.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

Information and Communication Technologies (ICT) energy demand is continuously increasing. Recent projections show that the fraction of commercial electricity consumed by ICT will be 10% of the total commercial electricity in the U.S. and almost 20% in Germany. In particular, projections for data centers in the U.S. indicate a growth in demand from 60 TWh/y in 2005 to 250 TWh/y in 2017 [1]. These figures show the need for more sustainable and energy efficient ICT technologies. Cloud computing is often regarded as to be one of those [2]. Indeed, one of the principles of cloud computing is on-demand provisioning of virtual resources, which can be aggregated on fewer physical machines. This allows to improve hardware utilization, thus increase energy efficiency.

Nowadays, energy efficiency is starting to be considered as a Service-Level Objective (SLO), i.e. a specific, measurable characteristic of a service, to be described as achievement values in Service-Level Agreement (SLA)¹. An example would be: “The energy bill of the client should be reduced by 20% in one year”. Cloud

service providers could benefit from representing energy efficiency as a SLO.

However, in order to offer cloud services, providers rely on very complex software architectures. The impact of architecture characteristics on energy efficiency is yet unclear and possibly unexplored. Also, we still miss explicit or implicit reference architectures that can help in increasing energy efficiency.

The role of software in energy consumption is widely discussed among the scientific community, and a number of metrics for software energy efficiency have been proposed [3]. Our work tries to advance to the next step: whether it is possible to quantify the effects on energy consumption when adopting a certain software architecture, and what architectural solutions can be adopted to increase energy efficiency in cloud-based software. We performed a Systematic Literature Review (SLR) [4] to investigate the relationship between cloud-based software architectures and energy efficiency.

The preliminary results of our SLO were reported on an initial publication [5]. In this paper we extend our initial work, as follows: Section 2 describes our review protocol in detail. Section 3 presents the results of a demographic analysis conducted on our primary studies. Section 4 provides insights about the state-of-the-art of energy efficiency in cloud software architectures. Section 5 gives an overview of the stakeholders for energy efficiency we identified during our research. In Section 6 we discuss the threats to validity that might affect our study. Section 7 concludes the paper with future outlooks and follow-up studies.

* Corresponding author at: VU University Amsterdam, De Boelelaan 1081a, Amsterdam, The Netherlands. Tel.: +31 (0)20 598 7788.

E-mail addresses: g.procaccianti@vu.nl (G. Procaccianti), p.lago@vu.nl (P. Lago), stefano.bevini@gmail.com (S. Bevini).

¹ <http://www.greenbiz.com/news/2009/01/12/energy-efficiency-new-sla>, last visited on June 12th, 2013.

2. Review protocol

Based on the motivation introduced in Section 1, we identified the following Research Question (RQ) driving our study:

RQ. *What software architectural solutions for cloud service provisioning can be adopted to achieve Service Level Objectives on energy efficiency?*

In order to answer our RQ, we followed a systematic literature review process. We performed a preliminary analysis of the research space, and we identified 306 hits (i.e. potentially related studies). We formulated a review protocol for our study, by defining a search query for academic databases and inclusion and exclusion criteria. Applying the protocol, we selected the primary studies for our research. We subsequently classified and analyzed these studies in order to extract relevant results.

In this section, we extensively describe our protocol, for the sake of reproducibility. All the main components of the protocol will be discussed: search strategy, study selection, data extraction, data analysis and traceability.

2.1. Search strategy

We adopted Google Scholar² as our data source. We defined a query string by selecting the most appropriate keywords to answer our RQ. We selected five keywords: “software architecture”, “cloud”, “service”, “SLO”, “energy”. Our query was defined after different steps, using the results of our preliminary analysis as pilot to test the coverage of the results. Namely, if one of the studies in our pilot was not retrieved by the query string, we refined it to add more keywords (typically, acronyms or alternative spellings, e.g. “service level agreement” vs. “SLA”).

The final query string was defined as follows:

“software architecture” AND cloud AND service AND “(energy OR power) efficiency” AND (SLA OR SLO OR “service level”)

The query string was applied to titles, abstract and body of the studies, to enlarge the scope as much as possible. The search was conducted in June 2013, with a specified time range from 2000 to 2013.

2.2. Study selection

In order to select our primary studies, we defined a number of criteria for inclusion and exclusion, (see Table B.3 in Appendix B). The criteria select papers in terms of their relevance to our RQ, but also in terms of scientific validity and language. In general, a study is selected if it fulfills all of the inclusion criteria, and excluded if it fulfills any of the exclusion criteria.

2.3. Data extraction

We used an extraction form in order to retrieve and store relevant information about each primary study. Besides general information, the form records how energy efficiency is addressed and which architectural elements were identified in the presented solution. The extraction form is structured as follows:

- **Study identifier:** provides an identifier for the study.
- **Study title:** the publication title.
- **Study type:** the publication type (i.e. journal article, conference article, thesis).

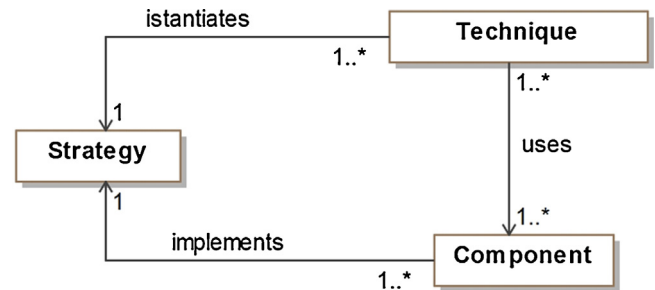


Fig. 1. Conceptual structure of our codes.

- **How energy efficiency is addressed:** a brief summary of how the presented solution addresses the energy efficiency of the cloud infrastructure;
- **Main architectural elements:** the main software elements of the solution.
- **Stakeholders:** stakeholders mentioned in the study that can be affected or involved in the architectural solution presented.
- **Validation:** whether the proposed solution has been validated in an Academic or Industrial setting, or no validation was performed. The validation is considered Academic when the article has been validated through a simulation or a test-bed. The validation is considered Industrial when the article reports a real case study (i.e. the proposed solution is already implemented in a software product).

2.4. Data analysis

Our RQ investigates how cloud software architectures deal with energy efficiency issues. The aim of an SLR is to “identify, analyze and interpret all available evidence related to a specific research question” [6]. Hence, we do not aim at directly providing new reusable solutions or patterns, but rather we aim at classifying the existing body of knowledge in a systematic way.

To elicit this information, we adopted coding. Coding is a qualitative research method, commonly used in social sciences, that interprets data and organizes it in categories or families, using codes, i.e. words or short phrases. Coding allows to capture the fundamental information of qualitative data in a systematic way, and enables us to link it and discover patterns and trends [7]. Our first step was an exploratory study of the selected contributions, in order to define an initial set of codes (or “start-list” [8]). The start-list was built by analyzing reference literature in software architecture [9–12]. We then arranged our codes in a conceptual three-level structure, shown in Fig. 1 and defined as follows:

- **Strategy:** the high-level approach through which a software solution addresses energy efficiency.
- **Technique:** the instantiation, or enactment, of a strategy through a specific technical approach.
- **Component:** an individual architectural component that plays a defined role in the application of a technique.

The concept of architectural strategy [13], technique (or tactic) [9] and component [14] are very well known foundational concepts of software architecture and they are familiar to practitioners and experts in the field. By adopting this conceptual structure, we aim at communicating our findings more effectively to software architects.

Finally, our primary studies were iteratively analyzed by two researchers independently, refining the list at every iteration until general and unambiguous codes were identified.

² <http://scholar.google.com/>

Download English Version:

<https://daneshyari.com/en/article/493791>

Download Persian Version:

<https://daneshyari.com/article/493791>

[Daneshyari.com](https://daneshyari.com)