The Green Lab: Experimentation in Software Energy Efficiency

# The Green Lab: Experimentation in Software Energy Efficiency

Giuseppe Procaccianti[*], Patricia Lago[*], Antonio Vetrò[†], Daniel Méndez Fernández[†], Roel Wieringa[‡]
[*]Department of Computer Science,
VU University Amsterdam, The Netherlands
{g.procaccianti, p.lago}@vu.nl
[†]Technische Universität München, Germany
{vetro, mendezfe}@in.tum.de
[†]University of Twente, The Netherlands
r.j.wieringa@utwente.nl

***Abstract*—Software energy efficiency is a research topic where experimentation is widely adopted. Nevertheless, current studies and research approaches struggle to find generalizable findings that can be used to build a consistent knowledge base for energy-efficient software. To this end, we will discuss how to combine the traditional hypothesis-driven (top-down) approach with a bottom-up discovery approach. In this technical briefing, participants will learn the challenges that characterize the research in software energy efficiency. They will experience the complexity in this field and its implications for experimentation.**

## I. Description of the Topic

The quest for software energy efficiency (EE) is primarily motivated by the growing energy demand of ICT and its consequent environmental impact. In response to this, hardware technologies substantially improved their EE in the last decades. This is, however, not the case for software, in spite of the software engineering (SE) research community acknowledging its crucial role in determining energy consumption [1]. Software EE is a research area that lacks of well-defined, validated methods: although there is a significant amount of scientific works in the field, they still show limitations and lack of generalizable principles and results [1].

## II. Aim and structure of the briefing

Our main goal for this briefing is to *raise the awareness on the challenges for software energy efficiency research* and provide a forum to openly discuss and agree on how to approach this research in the future. To this end, we plan the following sessions.

### A. Discover the Challenges

Participants of this technical briefing will learn the state-of-the-art of software EE research and discover its current challenges. Among them:

1) *Chaotic behavior.* Software EE research is characterized by a very large amount of variables to observe and control. Moreover, the causes of energy consumption in computing systems are sometimes counterintuitive [2]. For this reason, the relationship between software behavior and the energy consumption of the underlying hardware is hard to understand and characterize [3].
2) *Complexity of measurement.* The modern IT devices and platforms are highly heterogeneous: mobile devices, data centers, embedded systems and so forth. In order to make hypotheses (and theories) about software behavior, we need a complex measurement infrastructure to monitor energy consumption at different abstraction levels [4].
3) *Anecdotal and contradictory evidence.* Many empirical studies have been conducted to assess the factors that determine software EE. However, current evidence is mostly anecdotal and insufficient to provide generalized principles. This becomes clear from the contradictions in the conclusions drawn by researchers in different studies (e.g., EE versus performance [2], [5])
4) *Lack of a unified approach.* Different research communities tried to tackle the problem of EE within their own expertise, at the cost of precision in other domains. Hence, the studies lack representativity, and are characterized by heterogeneous techniques and analysis methods [1].

### B. Test a new Approach

During the briefing, a mixed approach to conduct experimentation for energy-efficient software will be presented to the participants. The goal of this approach is to identify *energy hotspots*. The concept of "hotspot" has already been introduced from a software architecture perspective in a crosscutting sense, both in performance and evolution, as actionable points of interest, crucial for a certain property. In software EE, similar concepts have been introduced, more specifically "Energy Bug" ("an error in the system, either application, OS, hardware, firmware or external, that causes an unexpected amount of high energy consumption by the system as a whole" [6]) and "Energy Smell" ("an implementation choice that makes the software execution less energy efficient" [7]). However, these definitions have some limitations, i.e. a questionable 'negative' connotation in the first case and a too narrow focus on implementation in the latter one. For this reason, we revised them and defined "energy hotspots" as *elements or properties,* *at any level of abstraction of the system architecture, that have a measurable and significant impact on energy consumption*. In particular, the focus of the briefing will be on *software* energy hotspots, i.e. software-intrinsic properties. To identify them, we test an approach that consists of two phases:

1) *Candidate Hotspots Identification*. In this phase, we measure and analyze a software application at different levels of abstraction, i.e. architecture, code, hardware usage and energy consumption. The software application is instrumented in order to gather software-related measures. Then, it is executed in virtualized environments, distributed over multiple physical machines. This allows to gather data from the virtual machines for resource consumption and from the physical machines for energy consumption. The gathered data is then processed using data mining techniques in order to discover patterns for *candidate* software energy hotspots. To guide our data processing, we leverage domain knowledge from the stakeholders, through interactive visualizations and dashboards. Such mechanism enables an iterative and supervised knowledge acquisition process, which is faster than the traditional experimentation [8], where usually feedback is downstream after results are obtained.
2) *Hotspots Verification*. After the identification, location, and characterization of the candidate hotspots and the mechanisms that possibly produce them, we have to reproduce these mechanisms and their effects in isolation in the lab. In our approach, this is done in a dedicated Energy Lab, where the second part of the research takes place. The validated hotspots are those candidate hotspots whose effect is confirmed in a predictable way for well-defined context variables at the end of this second step.

### C. Experience the Complexity

During the briefing, participants will experience the challenges presented in II-A, through an interactive simulation of phase 1 of the approach sketched in II-B. To this end, they will be provided with experimental data and visual representations of a software architecture (provided by an industrial stakeholder) in order to identify a set of candidate hotspots.

## III. Program of the Briefing

- *Session 1 - Discover the Challenges (30 mins).* In this session, participants will be introduced to the challenges of software EE through a state of the art overview.
- *Session 2 - Test a new Approach (30 mins).* Here, the new approach is presented and explained with examples.
- *Session 3 - Experience the Complexity (30 mins).* In this session, participants will be challenged to find energy hotspots in a case study provided by a real-world industrial partner.

Each session aims both at triggering the reflection of the community on the novelty of the challenges in this field and on collecting feedback and collaboration proposals as a driver for next research developments. The briefing offers also an opportunity to discuss the implications of the challenges on the characteristics of experimentation and on the validity of the obtained results.

## IV. Expected Audience

Due to the nature of our research, we aim at an interdisciplinary audience, coming from different industrial and academic backgrounds, interested in energy efficiency topics.

## V. Short Biographies

**Giuseppe Procaccianti** is a PhD candidate at VU University Amsterdam and Politecnico di Torino. His research interests focus on the interaction between software technologies and energy consumption. More info at: www.s2group.cs.vu.nl/people/giuseppe-procaccianti/

**Patricia Lago** is full professor at VU University Amsterdam. Her research interests include software- and service-oriented architecture, sustainability and green software engineering. More info at: www.cs.vu.nl/~patricia/

**Antonio Vetrò** is post doctoral fellow at TU München specialised in experimentation for software quality and technology transfer. More info at: http://www4.in.tum.de/~vetro/

**Daniel Méndez Fernández** is a post-doctoral senior research fellow at TU München. His research interests include model-based requirements engineering (and improvement), and empirical software engineering with a particular focus on qualitative studies. More info at: http://www4.in.tum.de/~mendezfe/

**Roel Wieringa** is Chair of Information Systems at the University of Twente, the Netherlands. His research interests include requirements engineering, IT security risk assessment, and design science research methodology for software. More info at: http://wwwhome.ewi.utwente.nl/~roelw/

## References


[1] P. Lago, R. Kazman, N. Meyer, M. Morisio, H. A. Müller, F. Paulisch, G. Scanniello, B. Penzenstadler, and O. Zimmermann, "Exploring initial challenges for green software engineering: summary of the first GREENS workshop, at ICSE 2012," *SIGSOFT Softw. Eng. Notes*, vol. 38, no. 1, pp. 31–33, Jan. 2013.

[2] K. W. Cameron, "Energy oddities, part 2: Why green computing is odd," *Computer*, vol. 46, no. 3, pp. 90–93, 2013.

[3] E. Saxe, "Power-efficient software," *Commun. ACM*, vol. 53, no. 2, pp. 44–48, 2010.

[4] P. Lago, S. Akinli Kocak, I. Crnkovic, and B. Penzensradler, "Framing sustainability as a software quality property," *Commun. ACM*, p. 9, to appear 2015.

[5] E. Capra, C. Francalanci, and S. A. Slaughter, "Measuring application software energy efficiency," *IT Professional*, vol. 14, no. 2, pp. 54–61, March/April 2012.

[6] A. Pathak, Y. C. Hu, and M. Zhang, "Bootstrapping energy debugging on smartphones: A first look at energy bugs in mobile devices," in *10th ACM Workshop on Hot Topics in Networks*, ser. HotNets-X. New York, NY, USA: ACM, 2011, pp. 5:1–5:6.

[7] A. Vetrò, L. Ardito, G. Procaccianti, and M. Morisio, "Definition, implementation and validation of energy code smells: an exploratory study on an embedded system," in *ENERGY 2013: The Third International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies*, 2013, pp. 34–39.

[8] A. Vetrò, S. Ognawala, D. Mendez Fernandez, and S. Wagner, "Fast feedback cycles in empirical software engineering research," in *International Conference on Software Engineering (ICSE)*, 2015.