EAM: Ecosystemability Assessment Method

Eric Knauss and Imed Hammouda
Department of Computer Science and Engineering
Chalmers | University of Gothenburg, Sweden
{eric.knauss,imed.hammouda}@cse.gu.se

Abstract—**In this extended abstract, we present the ecosystemability assessment method as a means to assess the extent to which a software system, represented by its architecture and its development environment, supports the vision of ecosystem.**

Index Terms—**software ecosystem, architecture**

I. Introduction

A *software ecosystem* (SECO) has been defined as a set of businesses functioning as a unit and interacting with a shared market for software and services, together with relationships among them [1]. These relationships are frequently underpinned by a common technological platform and operate through the exchange of information, resources, and artifacts [1]. The technological platform is often represented by a software system enjoying various levels of openness. We thus define *ecosystemability* as the extent to which a software system, represented by its architecture and its development environment, supports the vision of ecosystem. For companies, evolving an ecosystem's technological platform represents a complex process, as it needs deep understanding of the SECO phenomenon and the impact of its various facets on platform design decisions. Furthermore, since the SECO trend is relatively recent, there are little technical guidelines on how to grow and maintain a sustainable ecosystem.

In this paper we present ongoing work to design a method for assessing the ecosystemability of a technological platform as well as preliminary results. This method will allow to i) analyze business requirements and architectural drivers in SECOs and ii) to derive and assess architectural candidate solutions for technological platform and development environment. The research questions we would like to explore in this research include the following:

- What constitutes a quality model for ecosystemability?
- What kind of evaluation criteria could be used to assess the ecosystemability of a technological platform?
- How should the assessment be planned and which ecosystem actors should be involved?
- How to obtain data for the assessment process?
- How to exploit the results of the assessment process?

The closest works to our study are existing architecture-centric assessment methods for software systems such as ATAM [2]. These assessment methods, however, focus on analyzing specific software artifacts and quality requirements. What is needed is a review framework that addresses the various dimensions and needs of SECOs. Examples of such dimensions include technical qualities (e.g. architecture has been adequately designed and documented so that others can contribute to the SECO), licensing issues (e.g. the architecture enables integration of different third party and proprietary components), and the community perspective (e.g. the development environment supports knowledge flow and feedback between ecosystem stakeholders).

EAM (the envisioned Ecosystemability Assessment Method) extends beyond ATAM with respect to the following:

- *Input artifacts:* In addition to architecture design documents, the method considers artifacts, which are used to communicate with other SECO actors, e.g. partnership documents, interfaces, licenses, usage data, etc.
- *Quality attributes:* In addition to quality attributes like modifiability, security, and availability, the new method gives importance to other quality attributes that are significant to the ecosystem vision such as affordability, scalability, and interoperability.
- *Stakeholders:* In addition to the internal stakeholders and the evaluation team, the envisioned method integrates external SECO stakeholders, because anticipating needs and goals of external partners is vital for SECO success.
- *Output:* The envisioned method will extend reports on sensitivity points, tradeoff points, risks, and non-risks to include interactions with external stakeholders and their relative systems.
- *Process:* SECOs are dynamic systems, formed by the continuous interactions of its participants. It is crucial that the envisioned method provides mechanisms for continuous and agile architecture evaluation.

II. Methodology

Design of the EAM is based on empirical research methods. We started with a series of workshops with representatives of two large Swedish companies. By this, we contextualized the research problem to the companies' ecosystems and identified a set of themes with potential impact on the design of the technological platform along which software ecosystems can be analyzed. Example themes include nature of the ecosystem, governance models, and software platform (if there is one).

Our next step was to conduct a combination of literature review and archival study to refine the set of themes and identify the relevant discussion subtopics for each theme. As the perspective of this work is software ecosystems, we have started with reviewing two recent summative works on SECOs: a systematic literature review [3] and an edited book on the state-of-the-art of software ecosystems [4].

978-1-4799-3033-3/14 © 2014 IEEE

Having identified the subtopics of interest, our next step was to conduct a detailed archival study of example software ecosystems (e.g. Eclipse, Android, SAP). The goal was to illustrate the topics of interests identified in the form of a comparative study of the example ecosystemability.

III. Ecosystemability and Other Quality Attributes

Similar to existing assessment methods, EAM is quality-centric. It is thus important to depict where ecosystemability stands with respect to other quality attributes and quality models such as the widely used ISO/IEC standard [5].

In their systematic literature review on software ecosystems [3], Manikas et al. present a set of qualities for SECO architecture and identified the following quality characteristics:
Understandability. SECO actors should be able to easily comprehend the ecosystem and its underlying technological platform, e.g. the nature of the ecosystem, its business rules and restrictions, as well as extension and configuration options.
Expandability. A good SECO architecture should allow different ecosystem actors to accommodate additions to the core capabilities. Expandability also means to be able to customize and configure existing platform features.
Parallel development and testability. SECO actors typically develop their platform additions independently. A good technological platform should enable parallel independent development. In particular it should support testing of platform additions in the context of the SECO actors.
Maintainability. One of the major requirements for SECO architecture is to support ecosystem actors through improving and evolving the original technical platform to be able to create new business value.
Deployment and time to market. On the business side, a SECO architecture should allow for fast and seamless deployment of different versions of the product in different usage contexts. Easier deployment means faster time to market, which is a critical business factor for an ecosystem's industrial actors.

We believe that the above list of quality properties is a good starting point for a quality model for ecosystemability. Through collaboration with industrial partners we intend to construct a refined model and assessment framework.

IV. Example Ecosystemability Assessment Themes

Table 1 gives examples of our EAM assessment themes. The first column relates to the theme, the second gives examples of assessment criteria, and for some (those with white background) we give a short explanation on how this can affect architectural decisions about the technological platform. Together, these themes form an elicitation and analysis guide that allows characterizing a SECO under investigation as well as capturing business requirements and contextual properties that impact decisions about the evolution of the technological platform. We successfully applied this guide in a workshop setting (60-90 min.), which allowed us to compare the SECO under investigation with other SECOs that we previously assessed (e.g. in our archival study, see Sect. II) and to give examples on how others dealt with similar challenges. In the future EAM will allow comparing the ecosystemability of different architectural alternatives.

TABLE I
Examples of EAM Assessment Themes

Theme	Item	Significance
Nature of ecosystem	– **Degree of openness** – Level of ecosystem stack – Extension market	There is a tradeoff between the need to protect intellectual property and the need to be open with partners in the SECO. Certain architectural styles (e.g. core periphery) promise mitigation.
Governance	– Roadmapping – Niche creation – **Entry barriers**	A SECO with a high entry barrier (e.g. SAP) can afford lower understandability than a SECO with a low entry barrier (e.g. Android), aiming for not discouraging a continuous stream of new actors.
Software platform	– **Extension model** – Deployment activities supported – Operation supported by extensions	Reasons to limit the freedom and power of extensions include the ability for non-functional testing of the whole system or to protect the business model of the SECOs market place. The architecture of the technological platform could be designed to enforce such decisions.

V. Conclusion and Outlook

In this paper we present our research plan and preliminary results towards the development of the ecosystemability assessment method (EAM). EAM will offer platform providers in SECOs a checklist for significant aspects of ecosystemability and their assessment, best practices and guidelines for evolving the underlying technological platform, and a way to uncover opportunities in the ecosystem and to articulate its goals. Based on our preliminary results, we strongly believe that this will strengthen communication between ecosystem actors and help to make informed technical decisions in software ecosystems.

Acknowledgement

This research is funded by the Software Center and we thank the companies and their representatives who participated in our work so far for their enthusiasm and support.

References

[1] S. Jansen and G. Capelleveen, *Quality review and approval methods for extensions in software ecosystems*. Edward Elgar Publishing, 2013, ch. Software Ecosystems, pp. 187–217.

[2] "ATAM: Method for Architecture Evaluation." last accessed April 2014. [Online]. Available: http://www.sei.cmu.edu/library/abstracts/reports/00tr004.cfm

[3] K. Manikas and K. M. Hansen, "Software ecosystems - a systematic literature review," *Journal of Systems and Software*, vol. 86, no. 5, pp. 1294–1306, 2013.

[4] S. Jansen, M. Cusumano, and S. Brinkkemper, Eds., *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry*. Edward Elgar Publishing, 2013.

[5] "ISO/IEC 25010:2011," last accessed April 2014. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=35733