Issues and Challenges in Ecosystems for Federated Embedded Systems

Efi Papatheocharous
Swedish Institute of Computer Science (SICS)
SE-164 29 Kista, Sweden
efi.papatheocharous@sics.se

Jakob Axelsson
Swedish Institute of Computer Science & Mälardalen University
SE-164 29 Kista, Sweden
jakob.axelsson@sics.se

Jesper Andrersson
Department of Computer Science
Linneaus University
Växjö, Sweden
jesper.andersson@lnu.se

ABSTRACT
This paper discusses how Systems of Systems (SoS) can be constructed by linking together embedded computers in constituent systems to create complex but more flexible and adaptable systems. The approach of software system development is called Federated Embedded Systems (FES) and their revolved ecosystem of players is presented, aiming to ensure quality in engineering SoS. Ecosystems for Federated Embedded Systems (EcoFES) comprise a new area of research that scales component-based software development for embedded software into new dimensions. The proposed ecosystem dimension introduces an open, flexible and adaptable SoS architecture for improving the process of FES development. In the paper, we identify some architectural challenges and discuss the implications of scaling from a closed ecosystem to an open one, providing open collaboration and innovation in the context of FES.

Categories and Subject Descriptors
D.2.11 [**Software Architectures**]: Domain-specific architectures.

General Terms
Management, Performance, Design, Economics, Reliability.

Keywords
System of Systems, Embedded Systems, Product Development, Ecosystems, Product-Lines, Open Source, Business Models, Software Architectures, Software Components.

1. INTRODUCTION
The challenge of developing Systems of Systems (SoS) has received an increased attention from industry and the research community in recent years. SoS development is the process of combining dedicated complex systems for a specific target, i.e., developing high-quality complex systems in a rapid and efficient way. SoS are typically composed of software systems that are discovered, selected and composed at runtime. Each constituent system has a value on its own, even when used outside the SoS, may change over time and may be independently deployed and delivered by different providers.

Software ecosystems provide a complementary organisational view to SoS development, which introduces roles and rules of interaction, collaboration and synergistic capabilities for its constituent systems [5]. It requires explicit modelling of roles in different organisations and the rules that govern their internal and external interactions with respect to each organisation, for instance, when an organisation collaborates with independent third-party organisations. Such ecosystems aim to offer added functionality, innovation, flexibility, openness, and adaptability through orchestrations and choreographies of multiple heterogeneous systems, as well as increased productivity and efficiency in development.

Embedded systems (ES) are key components in many everyday systems, including automotive [3], home automation, energy, transportation, healthcare, and manufacturing [8]. External connectivity introduces a new dimension of technical challenges for ES manufacturers, as products become explicit configurations of subsystems developed by independent providers. As a consequence, these providers must introduce support and coordination for connected services in their product-lines, while contributing to satisfying ES specific constraints on the SoS level, such as real-time, limited resources and additional pre-defined platform technologies.

The ES industry exemplifies a relatively closed market where a direct chain exists only between suppliers and manufacturers. Recent works discuss how players could benefit if their ecosystems, including their development organisations, processes and software assets, were more open [1, 3]. However, the business decision of a closed market to move towards open ecosystems, thus changing the overall structure and infrastructure of the industry, has considerable impact on the related organisations' structure, processes and software assets. These effects, especially in the complex intersection of SoS and ecosystems, have not been analysed thoroughly in any previous work.

An example in the automobile industry is the software standard AUTOSAR which is a component-based framework that only allows flexibility at design time. Recent research [1] opens up this model to allow addition of plug-in software, allowing third-party add-on developers, through certain interfaces, to access sensors and actuators. In this way, a SoS can be constructed by adding plug-in software to ES configurations, that together with a base system realise critical SoS functions. We refer to this class of system of embedded systems as Federated Embedded Systems (FES).The notion is closely related to Cyber-Physical Systems (CPS) consisting of networked systems relying on physical processes, computation and communication technologies to extend the capabilities of physical systems.

The contribution of this paper is an analysis of Ecosystems for Federated Embedded Systems (EcoFES) which carries several novel challenges that motivate special consideration. We pin-point a number of challenges on the organisational, technical, and business architecture level. The identified challenges are primarily related to the ecosystems' processes, methods, and tools. The organisation of the remainder of the paper is as follows. Section 2 describes the EcoFES concept, the motivation and its main benefits in detail. Section 3 describes the architectural challenges. Section 4 summarises related work. Finally, Section 5 describes on-going and future work and discusses our conclusions.

2. THE ECOFES CONCEPT

The EcoFES roles and various layers are shown in Figure 1. It involves a combination of players (individuals and organisations) which make use or offer synergistic capabilities from a business, services and components perspective. The proposed structure promotes openness through collaboration, flexibility, innovation and adaptability for ES development, through the formation of federations. In order to best support collaborative development scenarios in the ecosystem, the identification of roles, strategies, processes, and resources need to be studied and orchestrated so that they co-evolve [9].

The EcoFES players (i.e., supplier, developer, end-user and manufacturer) will require a technical infrastructure to enable the formation and operation of these federations. For instance, end-users may use services that are provided by the manufacturer or may find the services offered by another player (e.g., a third-party developer or supplier) more attractive. In other scenarios, developers use internal or external APIs, frameworks, or Web services provided to the federation. This exemplifies that third-parties may provide new or refined business processes, services or components to the ecosystem.

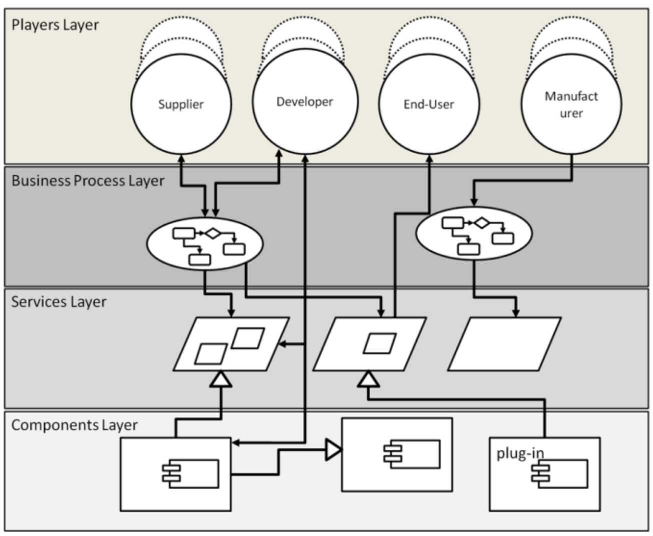

Figure 1. EcoFES Conceptualisation Schema.

Furthermore, from a SoS viewpoint, the software components that constitute the federation's services may be enhanced by many different innovative technologies, for instance, open-source plug-ins. In addition, EcoFES services and components may be made available to third-party players in the ecosystem that develop new plug-in software components.

2.1 Motivation

The motivation for conducting research in the context of EcoFES stems from the recent trends in software markets to open up for networked mass customisation and production [8]. Examples include apps for mobile devices, and software product-lines [2] that adopt an ecosystem approach and provide extended features to users. Failure to open up, according to Jansen et al. [8], causes decrease in sales, loss of jobs, jeopardises intellectual property and results to unhealthy markets. From an ES perspective we find several issues and concerns that motivate further investigation:

- Identifying players, roles, rules and supporting processes, methods, tools, and the technical infrastructure, that cultivate an open, innovative and collaborative environment for next generation ES.
- Forming ethical and democratic strategies that facilitate control, coordination and integration, release planning and project management, as well as evaluating alternative software architectures, development methods, reuse levels and make-or-buy options between manufacturers, suppliers and other third-party developers.
- Forming principles based on state-of-the-art practices and the research frontier. This requires empirical data from current practices and research, including experiences from researchers and practitioners regarding FES development.

2.2 EcoFES Benefits

The benefits of open ecosystems in the FES context include:

- Creating better and smarter products, at lower costs and more efficiently. Collaboration promotes innovation and shares the costs of adopting the most recent technology. It minimises lead time for new products and development and maintenance costs may be amortised on multiple parties. The products have more flexible architectures and are thus more easily adapted to respond to change, even after the initial deployment.
- Moving development closer to the markets. This will create an ecosystem that is more sensitive to market trends, thus more likely to offer products that are better aligned with market expectations and requirements, as well as better support recent mass customisation trends.
- Creating a community for collaboration that attracts and locks-in new and existing players. It offers organisation, product and application platform sustainability, once players are enabled to work collaboratively and dynamically for longer periods rather than on only one product or on a project-basis.

3. ARCHITECTURAL CHALLENGES

In this section we discuss the main challenges related to the EcoFES architecture. EcoFES combines several architectural views. Challenges are discussed reflecting three categories that we consider the most important; a) *organisational*, b) *technical* and, c) *business*. They are expressed and discussed from the particular viewpoint where dedicated ecosystems targeting specific markets participate in a larger federated ecosystem.

3.1 Organisational Challenges

The organisational architecture involves several challenges. Firstly, the players and their roles need to be identified and classified within the EcoFES. New organisational structures and ways of working (protocols) are required with regards to processes and means of interaction used. New collaboration mechanisms need to be defined for developing, integrating and managing products throughout their life-cycle.

For instance, by defining a key role in the ecosystem (e.g., the manufacturer) to make sure that the various technical challenges, some of which are discussed later, like dependability, interoperability, data management, architectural design, heterogeneity of components, life-cycle management and efficient coordination, are adequately addressed, is a way to tackle the overall complexity of aligning and supporting external and internal players' collaborations. The manufacturer must also provide a certain level of tool/platform support, e.g., simulators, which can be understood and used by third-party software developers to test their plug-ins without having access to all the details of the base product. This ensures that the manufacturer can maintain some control over the products. Thus, the organisational challenge includes designing and providing stable structures so that the business model can work efficiently, protect the interests/rights of all related players, ensure sustainability and ethical governance, and at the same time, satisfy the players' overall goals.

3.2 Technical Challenges

The technical challenges are identified in relation to the phase of development, i.e., a) initiation and definition, b) implementation, and c) integration and delivery. The main technical concerns are discussed below.

3.2.1 Initiation and Definition

During the definition of the product the architecture subsumes traditional product-line software architecture. Thus, in terms of technical and procedural specifications, the EcoFES concept will need to tackle several challenges that cross-cut more than one system and components. The processes need to support building multiple complex units of distributed functionality as the systems developed in the case of ES. Thus, in terms of specifications, qualitative descriptions need to be supported and automated as much as possible. This will reduce risks and costs of modifying specifications later in the life-cycle.

3.2.2 Implementation

Dependability is a key property for embedded and real-time systems. In an EcoFES, components and services will address cross-cutting, sometimes conflicting domains and be provided by different players. This triggers an abundance of challenges, involving composability, adaptability, flexibility, robustness and reliability of the processes and tools used. The particular type of required composability needs to be addressed in the service/component design, orchestration and choreography level. Adaptability and flexibility refer to offering explicit variability at different levels to respond to environment and domain specific requirements. Variability occurs in most concerns, including functionality and different quality concerns. An additional challenge at the definition level is to support mixed criticality.

Support for real-time properties and physical resource constraints of ES require special attention at the implementation level. A common implementation platform that supports the SoS's as well as its constituent parts' functionalities must be established by the ecosystem. This introduces additional dependencies and increases architecture complexity. Quality attributes are inherently cross-cutting and in an EcoFES, the dependencies will cross-cut ecosystems and players. This will create several risks that must be mitigated at the implementation level of the EcoFES platform. Thus, such a platform needs to be dependable, i.e., support criticality, be continuously responsive and robust. Criticality relates to availability of the SoS, robustness to failure handling and system recovery, while reliability considers continuity and integrity of the services provided.

Another challenge for the technical architecture is reusability, i.e., developing to support reuse and developing with reuse. This challenge requires that a reuse strategy is defined on the federation level. The strategy specifies the required level of reuse in each constituent ecosystem and provides adequate methodology to support synchronisation and coordination. In addition, dependencies between constituent ecosystems' assets, such as software product-lines, software libraries, and toolkits should be made explicit. However, in collaborative environments such as EcoFES, reaching an agreement on reuse strategies will involve complex business strategy considerations and will be influenced by the business models players select. This, all together, make reuse a key challenge for successful implementation of EcoFES.

3.2.3 Integration and Delivery

As the manufacturer is responsible for integration and systems testing at product delivery, the overall complexity and criticality of the products, add to the overall complexity of the process. Openness in collaboration to promote innovation is considered a major challenge, especially in the case of closed markets where a manufacturer requires control over products and processes considering them as core business assets and intellectual property. Finding ways to share knowledge and technology in a secure, trusted, collaborative and creative way is a fundamental challenge due to the lack of supporting framework and mechanisms that guarantee the controlled internal and external flow of information within and across organisations and to independent third-parties [7].

3.3 Business Challenges

3.3.1 Analyse Competitive Landscape

Identifying new and innovative business models for ES in the context of federated ecosystems involves several challenges. The EcoFES structures will be established in highly competitive landscapes, where players demand increase in connected systems and services, efficiency, productivity and quality, but also reduced costs, time-to-market, and delivery [7]. In many cases, such conflicting requirements need to be satisfied as the involved players come from different domains (e.g., manufacturers, suppliers, end-users). Additional barriers for a successful adoption of EcoFES include trust, conflicting business goals and strategies. Thus, a prerequisite for the successful formation of EcoFES is to study the players. This involves identifying their roles, policies, communication processes and tools, and aligning them with other players and their characteristics. A likely scenario for a specific EcoFES is to analyse multiple dimensions and define rules that mitigate risks connected to competition, negotiation, and licensing and ownership conflicts between the players [7].

3.3.2 Formalise the Business Models

A challenge that is subsequent to the analysis is the definition of business models that EcoFES should support. The models manifest the contracts negotiated and agreed to by current and future players. The models should be specified to ensure delivery of value, development of strategies to preserve economic health in the market, and formally serve players, their roles, and interactions. Jensen et al. [7] explain the challenges of formalising such an ecosystem, players need to realise their role, and manufacturers need to face risks such as opening their product interfaces, knowledge bases and source code to other parties. Thus, the formalisation of business rules that enforce the required openness to support open innovation while preserving the intricate properties specific to ES, such as real-time constraints and safety, is required. A novel challenge in this context is to support new and refined models after the EcoFES has been activated. This requires that the effects of a proposed change is analysed and communicated and, if necessary, negotiated among affected players before its introduction to the EcoFES.

3.3.3 Enforce Sustainability

The third business challenge relates to ecosystem sustainability and includes the ecosystem's processes and products. EcoFES services should provide the necessary business opportunities for external players to develop added value. However, EcoFES has inherent weaknesses in its fundamental principles that must be mitigated before it is activated. The openness promoted by EcoFES is not just a strength, but it is also a weakness. A player may for instance decide to leave the ecosystem federation, which affects an EcoFES at many levels. Achieving long-term sustainability requires an understanding of the challenges and risks involved upfront. Sharing assets and information while depending on others creates a degree of uncertainty that should be mitigated. This could influence business decisions, models and strategies of the players collaborating with respect to market and technology shifts. This situation is similar to a reuse strategy that involves external players. However, EcoFES principles such as the degree of openness and the tight coupling between systems in the targeted product domains amplify the complexity.

4. RELATED WORK

The term "ecosystem" was introduced by Iansiti and Levien that described the notion of business ecosystems through the examples of Wal-Mart and Microsoft [6]. In their article the authors explained the benefits of adopting the "ecosystem-thinking" from a business perspective and discussed various strategies organisations may utilise, based on their role in the ecosystem.

The term "software ecosystem" was introduced by Messerschmidt and Szyperski [9] but was extended by Bosch [2]. In particular, Bosch extended the classical "product-line-thinking" of software products. Bosch identified two reasons for the trend towards open platforms: firstly, it is too expensive for a manufacturer to develop alone all the functionality that customers would wish for, and secondly gathering the requirements for customisation could potentially be done more efficiently through an open platform. He also classified ecosystems according to two dimensions: the type of platform used (desktop, web or mobile) and also the level at which the ecosystem is created (on an operating system, application software, or own user-developed code).

Hanssen and Dybå [5] described in their work a systematic overview of software ecosystems and explained several related challenges, which overlap to a high degree with those reported by Chesbrough [4] as challenges of open innovation in open business models. Jansen, et al. [7] presented a research agenda for software ecosystems, discussed about the main challenges involved in a technical and business level through three dimensions: a) from a software ecosystem level, b) software supply network level, and c) from the software vendor level, and also mentioned issues of formal modelling, transparency, guidelines, standards, and actions, that are of central importance.

Research on ecosystems for ES is considered highly significant to our work, but looking into the related literature, very limited publications exist. Also, to the best of the authors' knowledge, the concept of "open innovation" in ES, and when Federations are formed, is hardly mentioned and therefore research in that sense lacks a broad and well-established knowledge base on the combination of the topics.

5. CONCLUSIONS AND FUTURE WORK

This paper described the concept, motivation, benefits and challenges for developing System of Systems (SoS) based on the notion of Federated Embedded Systems (FES). In particular, the focus has been on forming an ecosystem to wrap the players, processes, strategies and products. Our work provides foundations for our future work and the research we are currently conducting.

Aim of our research is to cover the following topics: (1) How would an innovative open SoS based on FES be described and modelled? (2) How could the SoS be controlled? (3) How would the virtual players and development teams be better organised? (4) What are the relationships between the different actors? (5) How would quality be ensured? (6) How should technology be structured? (7) How are intellectual property rights handled?, and finally, (8) what are the potential business models that should be established to support the process? The overarching aim of the research is a formal conceptualisation of the SoS for EcoFES which can be used for prototyping, deployment and commercialisation.

Our on-going research activities involve combining knowledge of technology solutions and software engineering with expertise in processes and models to create a deeper understanding of how to best design the ecosystems for open innovation where products will contain FES. Other activities include carrying out a systematic mapping and a systematic literature review to provide a high-level view on the related literature and clarifyg the major streams of research, application domains, industrial parties, and implications on business models, development methods and product architectures.

6. ACKNOWLEDGMENTS

This research was funded by VINNOVA, the Swedish Agency for Innovation Systems (grant no. 2012-03782).

7. REFERENCES

[1] Axelsson, J., and Kobetski, A. 2013. On the Conceptual Design of a Dynamic Component Model for Reconfigurable AUTOSAR Systems. In *Proceedings of the 5th Workshop on Adaptive and Reconfigurable Embedded Systems* (Philadelphia, USA, April 8–13, 2013). APRES'13.

[2] Bosch, J. 2009. From Software Product Lines to Software Ecosystems. In *Proceedings of the 13th International Software Product Line Conference* (San Francisco, CA, USA, August 24-28, 2009). SPLC'09.

[3] Broy, M. 2006. Challenges in Automotive Software Engineering. In *Proceedings of the International Conference of Software Engineering* (Shanghai, China, May 20-28, 2006). ICSE'06.

[4] Chesbrough, H. 2006. *Open Business Models*. Harvard Business School Press.

[5] Hanssen, G. K., and Dybå, T. 2012. Theoretical Foundations of Software Ecosystems. In *Proceedings of the 4th Software Ecosystems Workshop* (Cambridge, MA, USA, June 18). IWSECO'12.

[6] Iansiti, M., and Levien, R. 2004. Strategy as Ecology. *Harvard Business Review* 82, 3, 68-81.

[7] Jansen, S., Finkelstein, A., and Brinkkemper, S. 2009. A Sense of Community: A Research Agenda for Software Ecosystems. *In Proceedings of the 31st International Conference on Software Engineering* (Vancouver, Canada, May 16-24, 2009). ICSE'09

[8] Kobetski, A., and Axelsson, J. 2012. *Federated Robust Embedded Systems: Concepts and Challenges*. Technical Report. Swedish Institute of Computer Science. 2012:05.

[9] Messerschmidt, D.G., and Szyperski, C. 2005. *Software Ecosystems: Understanding an Indispensabl*e Technology and Industry. MIT Press.