Computers and Electronics in Agriculture 125 (2016) 12–28

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

A reference architecture for Farm Software Ecosystems

J.W. Kruize [a,b,*], J. Wolfert [a,b], H. Scholten [a], C.N. Verdouw [a,b], A. Kassahun [a], A.J.M. Beulens [a]

[a] Information Technology, Wageningen University, Hollandseweg 1, 6706 KN Wageningen, The Netherlands
[b] Wageningen UR, Agricultural Economics Research Institute, Alexanderveld 5, 2585 DB Den Haag, The Netherlands

ARTICLE INFO

Article history:
Received 11 June 2015
Received in revised form 24 December 2015
Accepted 16 April 2016
Available online 4 May 2016

Keywords:
Farm Management Information Systems
Software Ecosystems
Open Software Enterprise
Interoperability
Precision agriculture
Smart farming

ABSTRACT

Smart farming is a management style that includes smart monitoring, planning and control of agricultural processes. This management style requires the use of a wide variety of software and hardware systems from multiple vendors. Adoption of smart farming is hampered because of a poor interoperability and data exchange between ICT components hindering integration. Software Ecosystems is a recent emerging concept in software engineering that addresses these integration challenges. Currently, several Software Ecosystems for farming are emerging. To guide and accelerate these developments, this paper provides a reference architecture for Farm Software Ecosystems. This reference architecture should be used to map, assess design and implement Farm Software Ecosystems. A key feature of this architecture is a particular configuration approach to connect ICT components developed by multiple vendors in a meaningful, feasible and coherent way. The reference architecture is evaluated by verification of the design with the requirements and by mapping two existing Farm Software Ecosystems using the Farm Software Ecosystem Reference Architecture. This mapping showed that the reference architecture provides insight into Farm Software Ecosystems as it can describe similarities and differences. A main conclusion is that the two existing Farm Software Ecosystems can improve configuration of different ICT components. Future research is needed to enhance configuration in Farm Software Ecosystems.

1. Introduction

Agri-food supply chain networks are confronted with a growing world population and increasing prosperity and associated changing demands. These developments are challenging because the demand on food is increasing while there are stricter requirements regarding food safety, sustainable food production and transparent supply chains. Therefore, farm enterprises[1] are pushed to improve their production processes by smart monitoring and control. Smart monitoring, -planning and -control of production processes, which can be referred to as smart farming, can be supported by a broad spectrum of technologies, ICT components, and their constituent hard- and software systems (Aubert et al., 2012; Cox, 2002; Lamb et al., 2008; Wolfert et al., 2010). Examples of these ICT components are all kinds of sensors, terminals, implement assemblies, computers and software applications. For smart monitoring and control an integrated information system is required that enables seamless interaction and sharing of data between different ICT components. However, a lack of interoperability is currently severely hindering smart farming because ICT components of multiple vendors do not operate as one integrated farm information system (Aubert et al., 2012; Fountas et al., 2005; Pedersen et al., 2004; Pierce and Nowak, 1999).

To overcome this, Wolfert et al. (2014) identified five main challenges (i) handling the increasingly large amounts of data, especially from all kind of agricultural equipment, (ii) interoperability between various systems at farm level and in the whole supply chain network surrounding the farm, (iii) standardization of data, (iv) go beyond the small scale and the regional focus of farm software development while at the same time (v) comply with national or regional differences in farming practices. More specifically for interoperability, the systematic analysis of Kruize et al. (2013) showed that ICT components used within the same farm enterprise (i) have partly overlapping and partly unique services, functions and interfaces, (ii) are missing required application services, functions and interfaces, (iii) have separated data repositories and (iv) have inadequate and incomplete data exchange. In conclusion, most of the available ICT components are lacking both technical and semantic interoperability, resulting in data sharing

* Corresponding author at: Information Technology, Wageningen University, Hollandseweg 1, PO Box 8130, 6700 EW Wageningen, The Netherlands.
E-mail address: janwillem.kruize@wur.nl (J.W. Kruize).

[1] A farm enterprise can be an arable farm, livestock farm or horticultural farm. In this paper we focus on arable farm enterprises however it is expected that the concept Farm Software Ecosystem can address software integration challenges for the other type of farms as well.

http://dx.doi.org/10.1016/j.compag.2016.04.011
0168-1699/

issues and non-coherent user interfaces (Kruize et al., 2013). Consequently, current ICT components often hamper farm enterprise integration as they do not sufficiently support the monitoring, planning and control processes to enable smart farming. Supporting these processes by making a combination of multiple ICT Components is currently challenging. In addition, the creation of one overarching system developed by one software vendor that overcomes all mentioned challenges is neither a feasible nor – from a competitive point of view – a desirable solution. Hence, a promising method to achieve such integrated solutions is a best of breed approach, which allows users to configure customized software systems from standardized components that are supplied by multiple vendors (Light et al., 2001; Verdouw et al., 2010). As a consequence, software systems are not supplied by single companies, but by a set of independent actors which collaborate and can compete via an integration platform (Light et al., 2001). This integration approach requires an advanced infrastructure that covers both organizational and technological aspects (Wolfert et al., 2010). An organizational infrastructure is required that enables and facilitates both collaboration and competition between actors. In such infrastructure, actors collaborate in their development to provide interoperable ICT components that are based on their core competences and compete with ICT components that provide similar functionalities. A technological infrastructure is required that can support the linkage of ICT components into integrated FMISs. Both the organizational and technological infrastructure should enable and ensure a sustainable collaboration and competition in which all actors, including software developers, farm enterprises, contractors, technology providers and others, can flourish.

A concept that addresses such an infrastructure is nowadays called a Software Ecosystem. Currently, Software Ecosystems are becoming more widespread as they are increasingly considered to provide an effective way to construct large software systems on top of a software platform by combining components, developed by actors that are part of different organizations (Bosch, 2009; Manikas and Hansen, 2013; te Molder et al., 2011). Examples of current Software Ecosystems are, among others, Eclipse, Linux/Linux kernel and Android (Manikas and Hansen, 2013). At the moment there are no well-established Software Ecosystems for farming available, although several developments go into this direction. Large agricultural machinery vendors have setup their own proprietary platforms (e.g. John Deere's Farmsight[2] or AGCO's Fuse Technology[3]). With these platforms it is still difficult to establish interoperability with other components that come from other manufacturers. Several multi-vendor platforms (e.g. 365FarmNet[4], Crop-R, AgroSense, FIspace) are recently introduced, but these are still in an early stage of development and sometimes regionally oriented lacking a large international user base.

To gain deeper insights into these developments and to support further development of Farm Software Ecosystems, this paper proposes a reference architecture that can be used to map, assess, design and implement Farm Software Ecosystems that contribute to integrated FMISs. The purpose of the reference architecture is to improve communication and collaboration between multiple actors that are part of real-world Farm Software Ecosystems. It will help them to understand Software Ecosystems and enable them to join, form or improve Farm Software Ecosystems that lead to integrated farm information systems.

The remainder of this paper first introduces literature about Software Ecosystems and the relation to software development for farming. Second, the methodology for designing the reference architecture for Farm Software Ecosystems is described. Next, the requirements for the reference architecture, the reference architecture itself and an example farm information system that can result from a Farm Software Ecosystem is described. This is followed by an evaluation to verify the Reference Architecture based on the requirements and to validate if it can map existing Farm Software Ecosystems to provide insight how it matches and in what extend. This paper concludes with a discussion and outlook for future research and development.

2. Software Ecosystems and software development for farming

In the Internet of Services (IoS) software components are available as interoperable services on the internet. The IoS allows to decouple the possession and ownership of software from its usage and thus to use Software as a Service (Turner et al., 2003). Users do not need to buy and install a large software system, but required functionality is delivered as a set of distributed web services that can be configured and executed when needed. In contrast to traditional non-modular software systems, it is no longer necessary that components are delivered by the same software vendor. Software companies can concentrate on the development of components that fit best to their core competences. Users can configure customized software systems from standardized components that are supplied by multiple vendors that interact via a common technological platform. Such collaborative environments are nowadays referred to as Software Ecosystems. Software Ecosystems are defined as the interaction of a set of actors on top of a common technological platform that results in a coherent set of ICT components or services (Manikas and Hansen, 2013). These components include hardware, software and service modules, along with an architecture that specifies how they fit together (Eisenmann et al., 2008).

In practice, a Software Ecosystem is usually started by a single- or a group of software producing organizations that open up their business processes to become an Open Software Enterprise (Jansen et al., 2012). Such an Open Software Enterprise provides a technical platform and additional (collaboration) artefacts that are essential for the coherence of the software components and for collaboration between multiple actors (Seichter et al., 2010). There are various reasons why actors with different perspectives would like to collaborate in such an environment (Bosch, 2009; Wolfert et al., 2010):

- It increases the value of the core offering to existing users and increases the attractiveness for new users.
- Increase "stickiness" of the technology platform, i.e. it is harder to change the platform when it is widely used (cf. PC operating systems e.g. Windows, iOS, etc.).
- It creates and facilitates a structural and independent environment, developed by partners in the ecosystem that potentially offers a large critical mass of users (once success has been proven).
- Share the costs of innovation by collaborating with other actors and accelerate innovation through open innovation in the ecosystem.
- Decrease total costs of ownership and risks for commoditizing functionality by sharing the maintenance with networking partners.

The concept of Software Ecosystems is new for the agricultural domain. Related literature focuses on the integrating capabilities of farm ICT components by proposing a standardized infrastructure that supports the integration of ICT components of multiple vendors (Iftikhar and Pedersen, 2011; Kaloxylos et al., 2012; Nash et al., 2009; Steinberger et al., 2009; Wolfert et al., 2010).

[2] www.myjohndeere.deere.com.
[3] www.agcotechnologies.com.
[4] www.365farmnet.com.