Computers and Electronics in Agriculture 100 (2014) 168–179

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

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

A cloud-based Farm Management System: Architecture and implementation

Alexandros Kaloxylos [a,*], Aggelos Groumas [b], Vassilis Sarris [b], Lampros Katsikas [b], Panagis Magdalinos [b], Eleni Antoniou [c], Zoi Politopoulou [c], Sjaak Wolfert [d], Christopher Brewster [e], Robert Eigenmann [f], Carlos Maestre Terol [g]

[a] Department of Informatics and Telecommunications, University of Peloponnese, Tripolis, Greece
[b] Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Athens, Greece
[c] OPEKEPE, Domokou 5, Athens, Greece
[d] LEI and Information Technology Group, Part of Wageningen UR, Wageningen, The Netherlands
[e] Aston Business School, Aston University, Birmingham, UK
[f] Huawei Technologies Duesseldorf GmbH, Germany
[g] Sector of Manufacturing and Retail, ATOS Research & Innovation, Spain

ARTICLE INFO

Article history:
Received 26 May 2013
Received in revised form 16 November 2013
Accepted 27 November 2013

Keywords:
Farm Management System
Future Internet
Generic enablers
Services' marketplace

ABSTRACT

Recent technological advances have paved the way for developing and offering advanced services for the stakeholders in the agricultural sector. A paradigm shift is underway from proprietary and monolithic tools to Internet-based, cloud hosted, open systems that will enable more effective collaboration between stakeholders. This new paradigm includes the technological support of application developers to create specialized services that will seamlessly interoperate, thus creating a sophisticated and customisable working environment for the end users. We present the implementation of an open architecture that instantiates such an approach, based on a set of domain independent software tools called "generic enablers" that have been developed in the context of the FI-WARE project. The implementation is used to validate a number of innovative concepts for the agricultural sector such as the notion of a services' market place and the system's adaptation to network failures. During the design and implementation phase, the system has been evaluated by end users, offering us valuable feedback. The results of the evaluation process validate the acceptance of such a system and the need of farmers to have access to sophisticated services at affordable prices. A summary of this evaluation process is also presented in this paper.

1. Introduction

It is not long ago that farmers started using information systems to organize their financial data and keep track of their transactions with third parties (Batte, 2005). In developed countries, it is now commonplace for farmers to use sophisticated systems to monitor their crops. Data is collected from locally installed hardware (which may provide temperature, humidity, soil moisture, luminosity measurements, etc.) or from third parties such as meteorological services (Wang et al., 2006). A number of proprietary systems are used to process these data and assist farmers to manage and even control their farms in an efficient manner (Allen and Wolfert, 2011; Nikkilä et al., 2010; Wolfert et al., 2010). These systems are called Farm Management Information Systems (FMISs) – (Robbemond and Kruize, 2012). A FMIS is a system used for collecting and processing data to execute the operations of a farm. These operations include strategic, tactical and operational planning; implementation and documentation; assessment and optimization of the work performed in the fields or on the farms. To improve the execution of these functions, various management systems, databases, software architectures and decision models have been proposed to serve these purposes (Beck, 2001; Nikkilä et al., 2010; Sørensen et al., 2011; Fountas et al., 2006).

Existing and future systems in general, operate under a specific business model (Teye, 2011; Sørensen et al., 2010). Their main goal is to provide or collect information to/from farmers, process it and provide a number of intelligent services. These services are usually tightly integrated with the system. Existing systems are proprietary solutions that use closed specifications. This imposes a constraint on farmers since they do not have the freedom to enhance or tailor their systems according to their needs. They are, consequently, forced to use only the functionalities from the commercial products they have purchased. Furthermore, external

* Corresponding author. Address: Department of Informatics and Telecommunications, University of Peloponnese, 22100 Tripolis, Greece. Tel.: +30 2710372205; fax: +30 2710372242.
E-mail address: kaloxyl@uop.gr (A. Kaloxylos).

0168-1699/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.compag.2013.11.014

service providers cannot develop new services and make them interoperate with existing systems since these do not provide the necessary means such as Application Programming Interfaces – APIs (Kruize et al., in press). The creation of a marketplace of applications developed by independent service providers and which end users can choose to use by giving them access, in a secure way, to their data could alleviate the aforementioned problems.

In Kaloxylos et al. (2012) we presented the notion of a Farm Management System (FMS) as a framework that can accommodate modular services and enable their interoperation. The services can be either simple ones (e.g., a meteorological service) or even complex ones like an existing FMIS. In this way a marketplace of services and applications offered by different providers can be created. While this idea is similar to Google Play and Apple's App Store, the main difference is that the services are not executed independently, but rather they can interact by having access to the same data sets in the cloud and even exchange information through the FMS.

In this paper, we present a proof of concept implementation that was developed in the context of the SmartAgrifood Project.[1] Our implementation assisted us to identify the technical issues of a cloud-based system that could serve as a marketplace for services for the farmers. The implementation of this architecture as well some of the supported services has been based on a set of domain independent software tools called Generic Enablers (GEs) that have been developed in the context of the FI-WARE Project.[2] The purpose of these tools is to provide to software developers the means to develop in a fast and reliable manner a variety of cloud-based services for the Future Internet.

The FMS has been designed from a usage-driven perspective. This means that end-users' needs were identified and user requirements were formulated as central design goals. Recurrent design workshops and repeated end-user evaluations during the entire development process have been undertaken. The process of a usage-driven design and evaluation process were based on a seven step design approach (Nurkka et al., 2007; Brewster et al., 2012) by which research and design efforts were combined to deliver a gradually maturing design output. During these steps, we organized national panels with farmers and ICT experts, performed interviews with end users, created mock-up GUIs and videos, and asked users to fill-in electronic questionnaires. For our use case, we focused on developing a number of modular services related to greenhouses. The final system has been installed and used by a greenhouse in Crete, Greece over a period of nine months where the system was evaluated.

The rest of this paper is organized as follows. First, we briefly describe the FMS architecture and provide an operational example. In Section 3, we provide the system specification and discuss implementation details. We also discuss how the GEs have been used and provide some insight into their usefulness during the development phase. Section 4 presents the evaluation results from the end users and finally, Section 5 concludes the paper and presents our future plans.

2. FMS architecture

As described in Kaloxylos et al. (2012), the overall FMS architecture consists of two main entities i.e., the Cloud FMS and the Local FMS (Fig. 1). The Cloud FMS is equipped with a number of GEs that are used to support operations related to the management of a greenhouse. It contains a services' repository so that developers of services can upload their services for users to discover and use them. The Cloud FMS also contains a module called "FMS Controller" that consists of a number of sub-modules concerned with data collection, statistical analysis of data, coordination of activities, and the creation of notifications and commands to be executed by farming equipment. The "Management Functions" module provides information about the underlying network infrastructure, so as to fine-tune the operation of the overall system according to the current networking conditions. This module also contains functionality for recording any activity between services and users so as to apply specific charging schemes. All services that are delivered and, cooperating through the Cloud FMS, are able to access the same data sets stored in the FMS controller. The services access these data in a secure way. Appropriate permissions are granted to a service when a user registers with it. This communication is realized through a Service Oriented Architecture – SOA layer that sets the communication links between the core modules of the FMS and any specific service.

The Local FMS is located with the end-users (for example inside a greenhouse) and is mainly used to aggregate sensors' and possibly machinery data and forward them to the Cloud FMS. Also it contains some of the FMS Controller functionality that is used when there is no Internet connectivity.

3. Operational example

Fig. 2 illustrates an operational example using a representation produced by the Archimate (2013) tool. On the business layer level, a farmer needs to monitor the crop and machinery, and in case there are some out of the ordinary events, there is a need to come up with a correction plan and implement appropriate actions (see Figs. 3 and 4).

On the service level, data are collected from a plethora of devices and machinery and are transferred through the Local FMS to the monitoring service of the Cloud FMS. These data are then analyzed and when any abnormal situation is detected, then appropriate alerts are produced. At the same time, the coordination module communicates with the appropriate services (e.g., an expert system such as an e-agriculturist, a task scheduling service, or a meteorological service) and communicates their recommendations and reports either to the farmer or directly to systems installed on the farm so as to be automatically executed if the farmer has configured the system to do so. All these steps and actions can be recorded by the FMS database. The type and the amount of information to be used by these external services is configurable by the farmer by giving appropriate permissions to the services. The access from external services to a farmer's information can be configured in way similar to the one used by Android and Apple applications (e.g., grant permission to collect automatically the location information, acceptance to access specific data like sensor values, etc.).

4. System specification: the case for a greenhouse

In this section we present in detail an instantiation of the FMS architecture that has been developed and used for the management of a greenhouse. The Greenhouse Management prototype is a Future Internet (FI) compliant framework that takes into account real data from sensors and provides them to a Farm Management System (FMS) in order to take smart decisions regarding actions that need to be taken. External services have access to the real data collected and produce recommendations and reports related to the smart management of the greenhouse. Notifications and alerts about the current situation and actions are forwarded to the farmer. A farmer is thus able to have total surveillance and management of his farm using services developed by different service providers. Our prototype has been implemented in order to integrate a number of innovative concepts. In particular:

[1] http://www.smartagrifood.eu/.
[2] http://catalogue.fi-ware.eu/enablers.