Computers and Electronics in Agriculture 89 (2012) 130–144

Contents lists available at SciVerse ScienceDirect

Computers and Electronics in Agriculture

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

Farm management systems and the Future Internet era

Alexandros Kaloxylos [a,b,*], Robert Eigenmann [c], Frederick Teye [d], Zoi Politopoulou [e], Sjaak Wolfert [f], Claudia Shrank [g], Markus Dillinger [c], Ioanna Lampropoulou [a], Eleni Antoniou [e], Liisa Pesonen [d], Huether Nicole [g], Floerchinger Thomas [g], Nancy Alonistioti [a], George Kormentzas [e]

[a] Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Athens, Greece
[b] Department of Telecommunications Science and Technology, University of Peloponnese, Greece
[c] Huawei Technologies Duesseldorf GmbH, Munich, Germany
[d] MTT Agrifood Research Finland, Plant Production Research, Vihti, Finland
[e] OPEKEPE, Domokou 5, Athens, Greece
[f] LEI Wageningen UR, Wageningen, The Netherlands
[g] John Deere, Advanced Technology and Engineering, John Deere European Technology & Innovation Center, Deere & Company, Kaiserslautern, Germany

ARTICLE INFO

Article history:
Received 16 February 2012
Received in revised form 28 August 2012
Accepted 3 September 2012

Keywords:
Farm management system
Future Internet
Internet of Things
Generic enablers

ABSTRACT

Smart/precision farming systems are expected to play an important role in improving farming activities. During the past years, sophisticated farm management systems have emerged to replace outdated complex and monolithic farm systems and software tools. The latest trend is to enable these management systems to operate over the Internet. However, the Internet, in its current operation form, faces a number of shortcomings especially in handling vast numbers of networked devices (i.e., Internet of Things) or allowing a simplified integration of systems and services developed by different players. Currently, a number of research initiatives aim at addressing these shortcomings. Such an example is the "Future Internet" program launched by the European Commission. In the context of our work, we have specified a farm management system that takes advantage of the new characteristics that "Future Internet" offers. These come in terms of generic software modules that can be used to build farming related specialized modules. We present the functional architecture of this farm management system and provide an operational example. We also analyze the technological enablers that will make this architecture a reality.

1. Introduction

Agriculture is one of the most important areas of human activity worldwide. As the population rises there is a need to increase the agricultural production. Agricultural modernization due to commercialization, land-saving and labor intensive production between 1870 and the 1920s doubled agricultural production per land area (Olmstead and Rhode, 2009). Between 1920 and 1970, the total inputs used in agriculture increased 20%, while total output increased 179%. A few decades ago it was already noted (Duncan and Harshbarger, 1979) that the output increase was clearly not just an increase in the amount of inputs used but rather the technology knowhow for efficient agricultural inputs utilization. Recently, in Martin-Retortillo and Pinilla (2012) it was concluded in their research that the use of chemical fertilizers, biological innovations, harvesting and threshing machines, and mechanical technology mainly caused the increase in agricultural productivity per worker three folds between 1970 and the 2000s. Over the past 15 years however, farmers started using computers and software systems to organize their financial data and keep track of their transactions with third parties (Batte, 2005) and also monitor their crops more effectively. In the Internet era, where information plays a key role in people's lives, agriculture is rapidly becoming a very data intensive industry where farmers need to collect and evaluate a huge amount of information from a diverse number of devices (e.g., sensors, farming machinery, meteorological sensors, etc.) in order to become more efficient in production and communicating appropriate information (Csótó, 2010). These efforts deal with a number of factors such as ecological footprint, product safety, labor welfare, nutritional responsibility, plants' and animals' health and welfare, economic responsibility and local market presence. The efforts cover almost all steps in the production chain concerning the daily agricultural tasks, the transactional activities for all involved stakeholders and the support of information transparency in the food chain.

As reported in Sørensen et al. (2010), farmers often experience an overload of information, which originates from different data

* Corresponding author. Address: Department of Informatics and Telecommunications, Panepistimiopolis, Ilissia, 15784 Athens, Greece. Tel.: +30 2710372205; fax: +30 2710372242.
E-mail address: agk@di.uoa.gr (A. Kaloxylos).

0168-1699/$ - see front matter
http://dx.doi.org/10.1016/j.compag.2012.09.002

sources and is represented in various forms. Information brought to farmers originate from systems installed by third parties such as meteorological stations or specialized infrastructure, e.g. sensors for measuring temperature, humidity and soil moisture (Wang et al., 2006). Farmers need to combine all these data effortlessly and take precise decisions to produce qualitative products, improve their income and adhere to governmental regulations and principles. Further discussed in McCown (2012), all this information should also be combined with the ‘‘farmer's internal system of practical knowing and learning'', building thus a real cognitive system.

Nowadays, a number of proprietary solutions have been developed to help farmers manage their farms in an effective way[1] (Allen and Wolfert, 2011). More sophisticated systems track geographical areas, weather patterns and perform numerous advanced predictions (Nikkilä et al., 2010). Most of the latter mentioned systems, known as Farm Management Information Systems (Lewis, 1998), focus on specific tasks and use their own specifications to implement the functionality provided. Currently, these systems are slowly moving into the Internet era and are starting to use some of the well-established networking solutions to improve what they offer to the end users. However, it is widely accepted that the Internet faces a number of shortcomings, especially in handling vast numbers of networked devices (i.e., Internet of Things) or stakeholders. Moreover, there is still no standardized solution to enable a simple and cohesive interoperability among services and stakeholders. The Future Internet (FI) infrastructures are expected to handle these shortcomings. The aim of this paper is to propose a functional architecture of a farm management system (FMS) utilizing Future Internet capabilities. Our goal was not to build a complete management system but rather to focus on those functionalities that can be improved with the use of the innovative FI's capabilities. Using these capabilities the farmer should be able to perform a number of tasks that are not possible today (e.g., advertise his products effortlessly, discover trustable stakeholders, information and services, combine functionalities from different management systems and services, cope automatically with unstable data network links, etc.).

Future Internet as used in this paper is the planned infrastructure by the European Future Internet Initiative (EFII) combining Public–Private Partnership (FI-PPP) with an objective of significantly advancing the implementation and uptake of Future Internet services by 2015 and, in doing so, establish European-scale markets for smart infrastructures with integrated communications functionality. To achieve this, a number of general-purpose software modules are developed to be used in different sectors of everyday life. The functional architecture proposed in this paper is designed in the context of the SmartAgriFood EU project[2] that is part of the Future Internet Public–Private Partnership program. The main contribution of our paper is the adoption of the general-purpose software modules and their extension into farming specific ones thus, providing a cloud operating system that can integrate different services and applications.

The paper is organized as follows. Section 2 provides information about existing Farm Management Information Systems (FMISs) and explains what is currently not supported. The third section describes the methodology we have followed to analyze the problem area and extract functional requirements for our architecture. In Section 4, we present the proposed extensions, the operation principles of the future system and how these can be realized using appropriate technological enablers. In Section 5, we discuss which FI generic enablers and agriculture related domain specific enablers will be used and we also present in detail the functional architecture we have designed. After presenting an operation example of the architecture, Section 6 concludes the paper.

2. Related work and open issues

Farm management deals with the organization and operation of a farm with the objective of making a livelihood whilst dealing with global trade, traceability and consumer requirements, agricultural policies, environmental requirements, and the multi-functionality of agricultural enterprise as a whole. A Farm Management Information System (FMIS) is a system for ‘‘collecting, processing, storing and disseminating of data in the form of information needed to carry out the operations functions of the farm'' (Salami and Ahmadi, 2010). These functions include strategic, tactical and operational planning, implementation, and documentation, assessment and optimization of the performed work on the fields or on the farms. To improve the execution of these functions, various management systems, database network structures and software architecture have been proposed to serve these purposes (Beck, 2001; Nikkilä et al., 2010; Sørensen et al., 2010).

FMISs have also started to become ‘‘coupled'' mainly with some farming equipment (e.g., actuators) to allow the automatic execution of decisions if this is desirable from the farmers.

Currently, FMISs are providing significant services but their capabilities can be greatly improved. Wide-spread adaptation and exploitation of all potentials of existing information management systems for farms are hindered by certain issues. Existing systems are proprietary solutions that mostly have their own specifications about the functionality they provide and the means to interwork with external services. For this reason, there is no clarity and full transparency in technology and communication within the agricultural food supply chain.

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 services. These services are usually integrated in the system or more rarely provided by other service providers. These service providers may include governmental agencies, meteorological services, advisory services (agriculturists, veterinarians), spraying contractors and even logistics services, distributors and end customers. However as reported in (Sørensen et al., 2010) ‘‘…farmers report significant problems in using current agricultural information management systems, and particularly in transferring information between systems…''. A solution would be to build one platform where all services could be integrated. The obvious issue is that even if one tries to follow this approach[3] even in a really small number of these platforms per country, it is not feasible to integrate all services from all stakeholders of a global market. Even if the governmental agencies and meteorological services will be few and could register to such a platform, we expect that software developers for specialized services, advisory specialists, manufacturers, distributors and more notably the end consumers, etc. will be present in large numbers and it is not realistic to expect that they will be served by single platform providers (Wolfert et al., 2010). Thus, we need to enable the cooperation of the users and the application providers that may belong to different FMISs. The goal is that future systems should provide universal market places for the

[1] At the website https://sites.google.com/site/agrilabreferences/home recent results are published from research on a worldwide overview of used FMISs in the context of data exchange and standards used.

[2] http://smartagrifood.eu.

[3] Denmark is such an example, more info can be found in http://www.dina.kvl.dk/efita-conf/program/paperspdf/x_c_2.pdf and http://www.landbrugsinfo.dk/itvaerktoejer/sider/startside.aspx.

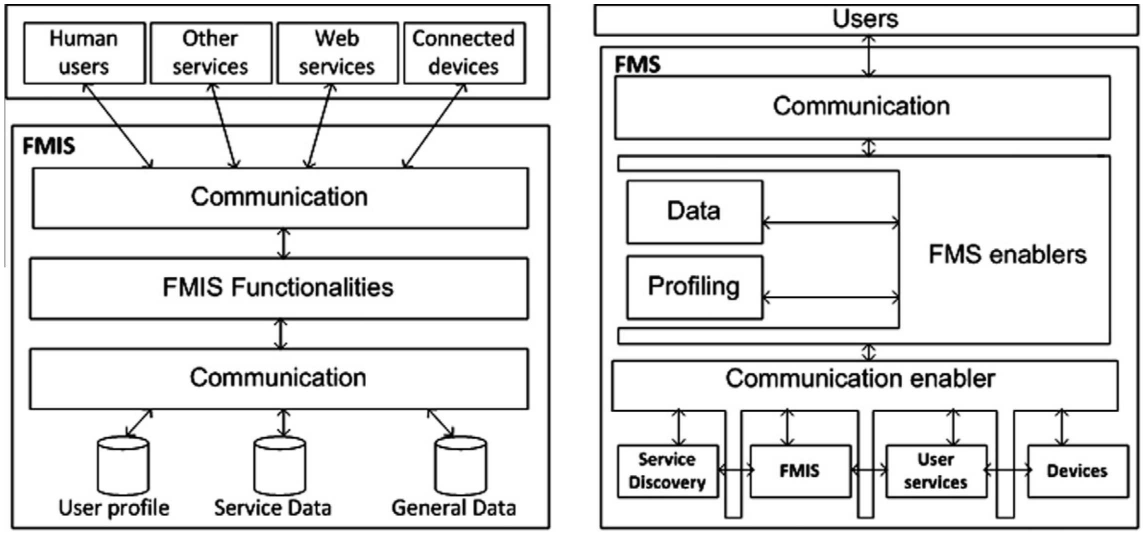

Fig. 1. The difference between FMIS and FMS.

publicizing, evaluating and subscribing for agricultural related services (being it FMIS or single service) in a plug and play manner.

Fig. 1, summarizes the core differences between existing FMIS's and the envisioned framework for the FMS platform.

Presented in Fig. 1, current FMISs on one hand are independent software running on the user's computer with connectivity to the FMIS provider's database or a complete web-based FMIS application. In current FMISs, functionalities of included applications such as farm accounting, weather services or planning of variable rate farm applications are orchestrated by the FMIS provider himself. The FMIS provider makes contracts with third party service providers if needed and offers these services to users. The FMIS stores users profile and data generate by services in their own format in their databases. The FMS on the other hand is an application framework that provides generic functionalities for service providers to offer different services to users. The FMS provides functionalities for registering services into a marketplace where users can discover and use these services. Because FMS is a mere framework that accommodates modular services, it means that an FMS can include both services and multiple FMISs. Every component of the FMS is portable and is packaged as a service offering from a provider, for example user farm data service, user profile service or weather service. Furthermore the FMS provides a vertical communication (interface) enabler for communication between different services registered into the FMS based of contractual or service usage agreement.

For the food supply chain, there is a need to define a system that addresses the above-mentioned shortcomings. This will allow the development of a standardized and transparent method of communication among different services and stakeholders. In addition, the integration of an automated method for discovering and subscribing to available external services will help in widening and utilizing the potential of various services available on the cloud.

Another interesting issue is how to enhance the existing functionalities, provided by proposed or implemented systems (Sørensen et al., 2011). In the current situation, many farmers still make their decisions based on implicit knowledge, intuition or routines that were established in previous generations (Nuthall, 2009). This is due to the fact that many problems in farm decision-making are unstructured problems that require a heuristic solving approach (Simon, 1996; Wolfert, 2002). At the same time, many processes and related decisions in farming (e.g. sowing, harvesting) occur only at a low frequency and the natural environment causes different conditions for each occurrence, so this makes it extremely difficult to learn from previous occurrences and situations. In literature, methods have been proposed to support and improve this implicit learning cycle (cf. Wolfert 2002; Fountas et al., 2006), but so far these methods are not incorporated in current FMISs. In many developed countries there is the trend of increasing farm sizes and decreasing the number of traditional family farms. In that case, a whole team of workers is managing the property and sometimes these persons move from one farm to the other. Because of specialization, land is rented sometimes by different farmers each year. These developments make it even more challenging to support the learning process, because decision-making does not take place in the head of one single farmer anymore and the history of land use is fragmented. At the same time, current monitoring techniques, with the potential of collecting huge amounts of site- and time-specific data, provide the opportunity and capability to improve this situation more than ever before. However, this puts high technical requirements on the architecture and infrastructures of these monitoring systems (e.g. data storage, transfer of data ownership, etc.), but moreover it requires business intelligence in order to find useful relationships between decision-making and data of the farm system in order to improve the learning cycle. Also, this intelligence is located in a central point. This can be catastrophic if the farmer has an unstable network link and looses very often, the connection to the Internet or if the network link capacity is limited. These network connectivity problems are to be expected in wireless and mobile networks especially in rural areas (Subramanian et al., 2006).

Another point is how to improve the management of a FMIS itself. Current trends in cloud computing consider the use of autonomic computing (AUTO-COM). In simple terms this means that the software components are equipped with context awareness and autonomic functions to automatically manage their mode of operation without any human intervention whenever this is needed. This simplifies the work for the FMS provider and reduces his operating expenses (OPEX) in terms of experts needed for managing and fine-tuning the operation of the FMIS modules. Finally, there is a need to extend the functionality of the FMISs by collecting data related to the end users devices as well as the status of the underlying data networks. These issues are further described below and specific solutions are proposed.

In the rest of the paper we use the term farm management system to describe a new Internet platform that operates as a typical

FMIS but it also allows the simple integration of external services and even existing FMISs. The platform allows stakeholders (e.g., farmers, traders, spraying contractors, etc.) to communicate with each other and also form dynamic trustable and secure relationships among them. Finally, each FMS is enhanced with autonomic and cognitive features and better interworking with the underlying network infrastructure.

3. Analysis and design methodology

To initiate the formulation of the architecture, an analysis of a large set of requirements defined by end users and solution suppliers was performed. The architecture is designed taking into account the technology platform of the FI. The proposed architecture involves a diverse set of stakeholders and devices along with information that has to be exchanged among them.

To ensure that the architecture will be designed in a meaningful way, a standardized process of analysis has been followed and adapted for our needs (Robertson, 2004). The first step was to produce a significant number of use cases that describe different usage scenarios. By the definition of Carroll (1995), a use case is a concrete description of an activity that the user engages in when performing a specific task. The description is sufficiently detailed so that design implications can be inferred and discussed. In the method presented in Robertson (2004), use cases include scenarios, providing content for the use case. Alexander (2004) state that scenarios vary from brief stories to richly structured analyses, but are almost always based on the idea of a sequence of actions carried out by intelligent agents.

The use cases we have produced followed these principles. They were captured on pre-defined templates where their authors had a number of record fields to fill in (e.g., use case description, use case goals, involved players, technical requirements, assumptions, preconditions, triggering event, sequence of actions, terminating condition, expected benefits for the stakeholders, etc.). In the analysis applied in this study, the use cases include scenarios presented as sequence of actions and scenario steps.

Our goal was not to build another FMIS that would cover all aspects for farmers. Our goal was rather to identify those functions that are currently underperforming or those that can be considerably improved with the use of FI attributes. For this purpose a group of specialists from a variety of fields (e.g., standardization bodies, agriculture researchers, software developers, and network experts) produced the list with the most relevant or challenging use cases that were related to FI. Moreover, to be consistent with the needs of farmers, the input to these studies was based also on a series of on the spot visits to arable and dairy farms, greenhouses and other indoor cultivations, phone and personal interviews as well as scientific research concerning farm management systems that have already been developed in previous European or national projects (e.g., FutureFarm, agriXchange, PPL, iGreen, etc.). The result was 29 use cases which are presented in Table 1.

From these use cases, information was collected based on the analysis of the detailed actions specified to handle specific events. The result was that each action was mapped into a functional requirement that the architecture should provide. For example, when analyzing the text describing an action such as "…the farmer receives a notification in the end device he is currently using" we extracted related requirements such as "automatic selection of end terminals should be available". Similar requirements from different UCs were merged into one requirement while for traceability reasons a pointer to these UCs was also kept. These functional requirements were further categorized and grouped into functional blocks. Examples of the produced functional requirements and their grouping into functional blocks are presented in Table 3 in the *Appendix* of the paper. This step enabled us to identify which are the functional areas we should focus and even draw some generic operation principles of the overall FMS architecture. These operation principles are presented in Section 4.2.

The next step was to envision how these functional blocks will communicate with each other and with other external services. In this design process, special attention has been paid on how we can combine well-established technologies with now developing mechanisms and agricultural specific tasks to produce a really evolved functional architecture. This architecture is presented in Section 5.

4. Extending the FMS functionalities

In this section we discuss which are the three main functional areas of a FMS that need to be improved in the future. These areas are chosen based on the requirements analysis described in the previous section. We also provide the envisaged generic operation principles these systems should follow and we discuss the tools that will be provided by the Future Internet Core Platform to enable us to implement these systems.

4.1. Improved FMS functionalities

4.1.1. Improving the cooperation among stakeholders

A first point is that the farmer should be able to change a FMS provider with less possible impact to his operation. This means that all his raw data stored in the FMS's DB should be available to be transferred automatically to another FMS, if this is desired. Moreover, the farmer may require taking advantage of an alternative service that is not registered/provided in/by the FMS, where he is currently subscribed (e.g., find alternative spraying contractors in the area). The system should support the discovery (using appropriate repositories and registries), the evaluation and the incorporation of such a service. The data of such a service should be able to be used transparently by the "intelligent" FMS system. This calls for some standardization on the communication among these services, using service composition schemes. The interworking of services registered to different FMSs also calls for accounting procedures that will apportion a fee among FMSs and their services.

A capability to evaluate, rate or give comments on subscribed services is also desired. Also, the system should support mechanisms that are adaptable to the location of users and enable the cooperation of farmers registered to the same or different FMSs. This cooperation is expected to increase their understanding of a situation and be aware of best practices effortlessly. Also, real time recommendations about new services, offers, and opportunities should be supported. Finally, in the Internet eco-system, the ability to evaluate stakeholders, calls for the design of appropriate mechanisms (from simple voting and reputation schemes to opinion mining schemes).

To summarize, the farmer should have access to the global market of services and stakeholders irrespective of the management system that is currently serving him. Moreover, mechanisms are needed among the management systems to allow the farmer's related information to be easily accessible to end users, traders and also by specialized stakeholders (e.g., spraying contractors, agriculturists, etc.). All these requirements call for new schemes that will automate the cooperation among FMSs, services and stakeholders related to farming activities.

4.1.2. Enhancing and distributing a FMS's functionality

The decision making process is the heart of a management system. Until now, intelligence and knowledge used in FMISs is

Table 1
Use cases.

Name of use case	Description
1. Yield measurements system	Collecting quantitative and qualitative information about crops, fruits and vegetables before harvest
2. Extraneous and foreign bodies identification	Identify and remove foreign bodies that can occur and cause hazards during the processes of arable crop and vegetables production and processing
3. System for milk quota	To distribute the national quota with the assistance of monitoring and recording the milk quantity in order to help every member operate his production volumes
4. Collaborative spraying	Coordinate several tractors within a fleet to work together for spraying an area
5. Plant disease forecast for spraying	Provide a forecast and warning on the onset of plant disease. Also provide recommendation on disease spraying agent and ways of executing spraying task using tractors
6. Preparation and setup for plant disease spraying	Prepare a Variable Rate Application equipped sprayer for a Precision Agriculture spraying operation. The scenario shows the need and importance of fluent and reliable information flow between several actors, services and machinery
7. Dealing with bad weather during spraying	Help in deciding how to perform spraying operation as a result of weather conditions and provide information about spraying process and other spraying fleet for informed decision making
8. Cooperating harvesting	Handle issues with unreliable networks/limited network coverage (rural areas) through virtual servers in the cloud, use prognostic tools/model for extrapolation (on vehicles and in the cloud)
9. Online firmware update	Keep farming machinery (e.g., tractors) up to date with the latest firmware releases in an automated way.
10. Analysis of logged date for process optimization	Collect data logs from machinery (e.g., tractors) and perform data mining operations to find optimal settings for a given machine and task
11. Remote machine control	Operate a tractor with a trailer remotely by a SPFH during loading
12. Remote machine diagnostic	Using remote machine diagnostic over a wireless link to identify a possible problem on a tractor
13. Greenhouse management – normal operation – local data storage – system data storage	Define information flows and interfaces among a number of involved entities. To identify possible areas where automation and information management is needed to be specified
14. Faulty operation of sensors inside a farm	Automatic identification and isolation of a faulty sensor
15. Agricultural related news coming from the outside world	Personalized and automated collection of information from different external sources
16. Providing a farmers' information to different external entities/players	Farmers should be able to advertise their products effortlessly and allow other stakeholders to have authorized access to their data (e.g., a spraying contractor) to complete a task
17. Subscription to an electronic advisory service over the Internet	A farmer should be able to discover a service and subscribe to it and even give to it authorize access to some of his collected data
18. Different farmers exchange data	Collect and use the knowledge of other farmers (e.g., their environmental status, their actions and results, their opinions, etc.) to achieve better results
19. Statistics management	Define automated ways to received statistical data for a number of monitored parameters
20. Multimedia transfer	This scenario describes the capability of the system to leverage multimedia transfer (photos, videos HD)
21. Notifications are sent to more than one available end-terminals	The farmer should receive any notifications to one or more suitable devices at a time
22. Decision making is provided by the local system	Describe how the system should operate in case the Internet link goes down
23. Farm management – small scale barcode/RFID system – traceability system	Identify "tracing solutions" and discuss how they can be combined with the overall system
24. Production of a cultivation plan for new farmers	Help new farmers find information about cultivation practices, equipment, etc.
25. Advanced search engine	Help farmers and visitors find information easily and present the information in a user friendly way
26. Access to common infrastructure	Help farmers reduce cost of own an infrastructure by allowing an easy way to share it
27. Providing a farmers' information to certification authorities – players	To have a more transparent system and to help the authorities access their data without losing a lot of time to search to different FMIS
28. Information service for farmers interested in selling/buying animals	Help farmers to identify other farmers that are interested in selling/buying animals or are interested in fertilization of their animals
29. Qualitative products of dairy farms	Help customers and intermediate suppliers to identify farmers with qualitative products and farmers to promote their products and achieve better prices

mainly static and its efficiency is related to the skills of the agricultural specialists and the application developers. Technology can improve this status by introducing "cognitive functions" to the system. This can be achieved by allowing the outcomes of each farming related data and corresponding farm management actions to be recorded and further analyzed to produce new rules. This process is dynamic and is actually a closed control and management loop. Models, statistical analysis and data mining are used to create to cognitive rules.

Moreover, moving all demanding functions to the cloud may create solutions where the intelligence will be solely located in the cloud. This is a good choice if a farmer has a stable and fast Internet link. But this is not expected to be the case for open fields in rural areas, where unstable wireless links are often the only way to access the Internet. The overall operation of a FMS cannot of course rely on the quality of the communication link. This calls for some distribution of intelligence among the cloud and the farm. The existing processing power of even small devices can provide some decision functions that can be executed on a field where needed. This requires some appropriate transfer of information during suitable time periods. For example, some of the devices on a farm or a field need to always have some summarized information about the status of tasks to be executed, data from external services such as meteorological data, with whom a farmer should contact with if things do not go as scheduled. All these are needed to have even some "limited" intelligence inside the farm when the connection with the Internet will not be possible. This also calls for the need to standardize interfaces among the cloud services and the underlying network infrastructure or even the end devices.

4.1.3. Introducing autonomicity in FMSs

Managing a FMS is going to be a complex task. The significant number of services, users and end devices to be supported as well as the interoperation with the communication network infrastructure, calls for some automation on the management of the management system itself. This automation requires the system to be context aware and act autonomously when a situation arises that need corrective actions. The notion of "self-management" is currently used by both the computer science (Kephart and Chess, 2003) and the telecommunications world (Samaan and Karmouch, 2009) in an attempt to introduce an autonomous management process for systems and devices. Below we explain the main categories of self-management mechanisms and how these can be applied in a FMS.

Self management mechanisms can be divided into self-(re)configuration, self-healing, self-optimization and self-protecting. Self-(re)configuration can include actions for adding components to the farm (e.g., sensor, tractor) and let them becoming automatically configured. Initially, this configuration requires that they will acquire some sort of address. Then, they will automatically identify where to store their data (into a data aggregation point or a database). Also, in case a component can receive commands from a software module of the FMS it has to be configured to do so. This automatic configuration can be also applied when adding a new service in the FMS, since it needs to automatically allocate resources such as physical space and recognize services it needs to cooperate with.

Self-healing mechanisms are needed to handle malfunctioning equipment (e.g., isolate a malfunctioning sensor) or to handle a troublesome network situation. For example, the loss of the communication link with the Internet will require some reconfiguration to allow the system inside the farm to continue its operation even with limited functionality. In such a case, a dynamic reconfiguration of the system will be executed (e.g. re-configure the flow of information, find alternatives for storing data) as long as the main communication link is disrupted.

Self-optimization deals with the evaluation of executed tasks and the continuous searching to optimize the operations inside the farm. This sort of optimization can take place at any time. Modeling, statistical analysis and evaluation across key performance indicators are required for these tasks.

Finally, self-protecting refers to all automated actions needed to ensure the security of the system and self-explaining contains all mechanisms used to explain to users why actions are taken and decisions are made in an easy and transparent way.

4.2. Generic operation principles

Based on the functional requirements analysis, the general concept of the smart farming system as well as the main functional entities were identified. A first observation is that all use cases follow a generic pattern. Specifically, the first action is to collect information (i.e., monitor the farm/field) about the current status (e.g., plant and animal status, machinery status, etc.). This information is mainly collected from sensors, tracking systems, and agricultural machinery and Internet services that produce raw data which in turn have to be processed (e.g., filtered and aggregated) and analyzed. Secondly, the system stores this information and it analyses and processes the collected data (i.e., information and knowledge building). Then, according to the analyzed information the system takes simple or more complex decisions for certain actions and executes them. Finally, all the information, as well as the executed actions and their results, must also be stored for further use. In this way, through appropriate mechanisms (e.g., data mining), we can learn more things about the overall performance of the FMS and optimize the decision making process. Fig. 2 summarizes all the aforesaid phases that actually constitute a typical autonomic and cognitive management loop.

One important aspect is how the architecture of the FMS should be organized. A straightforward approach is to follow a hierarchical organization (Fig. 3). The purpose of this organization is to distribute the intelligence for decision-making across two levels. Firstly, the FMS will operate in an Internet of Things (IoT) environment. Farm management units (FMUs) may be simple devices with limited capabilities (e.g., sensors) or may have added intelligence (e.g., tractors that may collect data, and support some self-x capabilities such as limited decision making and automatic firmware update). The "Local FMS" is expected to be an on-site node that can aggregate, pre-process and implement simple corrective actions or produce suggestions for the farmer. It can even take the overall control of equipment and machinery installed in a farm when the link to the Internet is lost. The "Cloud FMS" will be based on a cloud implementation that can provide Infrastructure as a Service (IaaS), Platform as a Service (PaaS) or Software as a Service (SaaS) to application developers, service providers and farmers. It will have the overall control under normal operation conditions. Also, note that key performance indicators can be reported to Local and Cloud FMSs so as to be analyzed and checked against reference values or patterns. If deviations are observed, corrections need to be implemented to optimize given targets.

The two FMS subsystems (Cloud and Local) will be implemented using both generic software modules that are applicable for tasks not solely related to the farm management tasks as well as software modules specifically used for agricultural tasks. We call the former "generic enablers" and latter "domain specific enablers". These notions are further explained in the following section.

The vision is that different providers will offer FMS services to their subscribers (i.e., farmers). They will operate in accordance to the abovementioned hierarchical structure and autonomic principles. They will integrate a number of additional services, developed and offered by other service providers. These FMS associated services will form part of a marketplace where clear Service Level Agreements (SLAs) will be formed among the FMS provider, the service providers and the end users. Appropriate charging/billing schemes will also be defined. The FMS provider will "certify" that these services have been tested to operate without any problems for the FMS users, even when they are composed or their data are mashed up with those of other services.

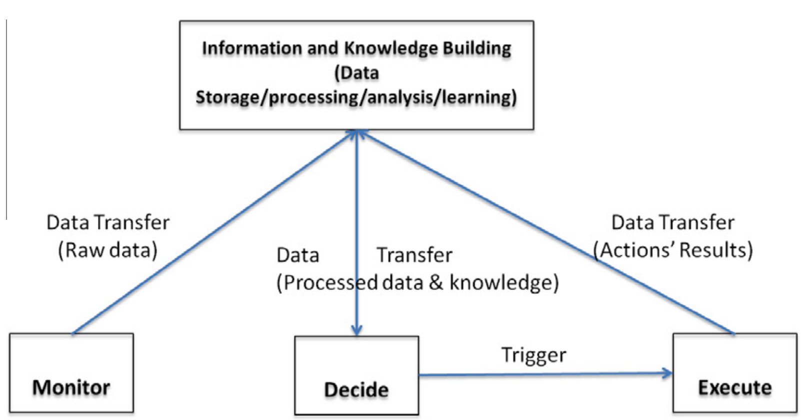

Fig. 2. Generic operation principles.

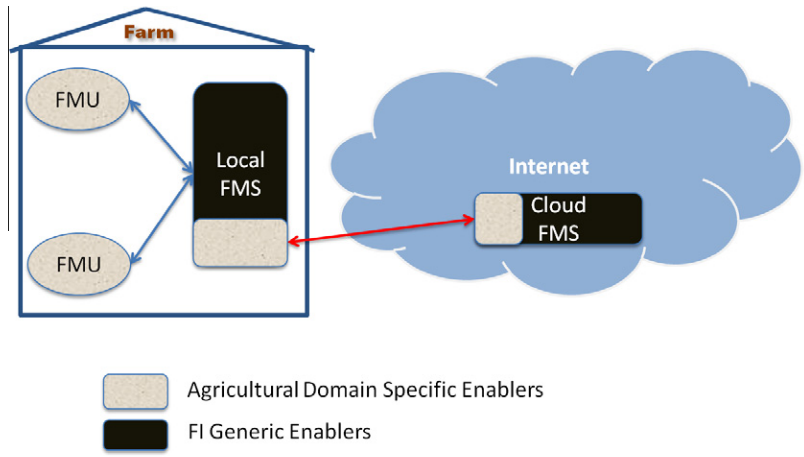

Fig. 3. Organization of smart farming system.

The FMSs will also give the opportunity to their subscribed farmers to discover and have access to services located outside their domain of operation. All these services will be offered through different FMS providers or even as independent services offered by service providers not related (i.e., registered) to any FMS.

Apart from services, the Cloud FMS is specified to be a marketplace for stakeholders (e.g., agriculturists, spraying contractors, manufacturers, etc.). As with services, farmers will be able to discover these stakeholders. This discovery process will be performed in a trusted way that will allow the farmer to evaluate the different stakeholders through for example opinion mining schemes. Stakeholders that are not associated with an FMS provider will be also accessible through public registries. Thus, in a sense, the public registry is a yellow pages system for both services and stakeholders that are not associated with a FMS.

This concept is presented in Fig. 4, where farmers can subscribe to one of the two illustrated FMS providers and also can have access to other services and stakeholders through a public registry.

4.3. Generic and domain specific enablers: the "Future Internet Core Platform" case

As discussed earlier, future FMSs need to ensure a seamless integration and mashup of different supporting services. They also need to enable an exchange of information and business processes among stakeholders. It is expected to process distributed sources of information and provide a scalable platform with standardized interfaces. As today's FMS solutions are proprietary they implement specific stovepipe solutions of a national scope and are not able to easily achieve the aforementioned desired characteristics. Thus, a future-proof framework is required, on top of which FMS vendors and solution providers will implement their systems and services.

This platform should be part of the "Future Internet", which is more than just a bit pipe of IP packets: it will provide solutions for today's shortcomings of the Internet (e.g. features like security, performance, service integration and scalability). Whereas "Future Internet" is a general term for research activities in the area of the Internet architecture, the European Commission has launched the

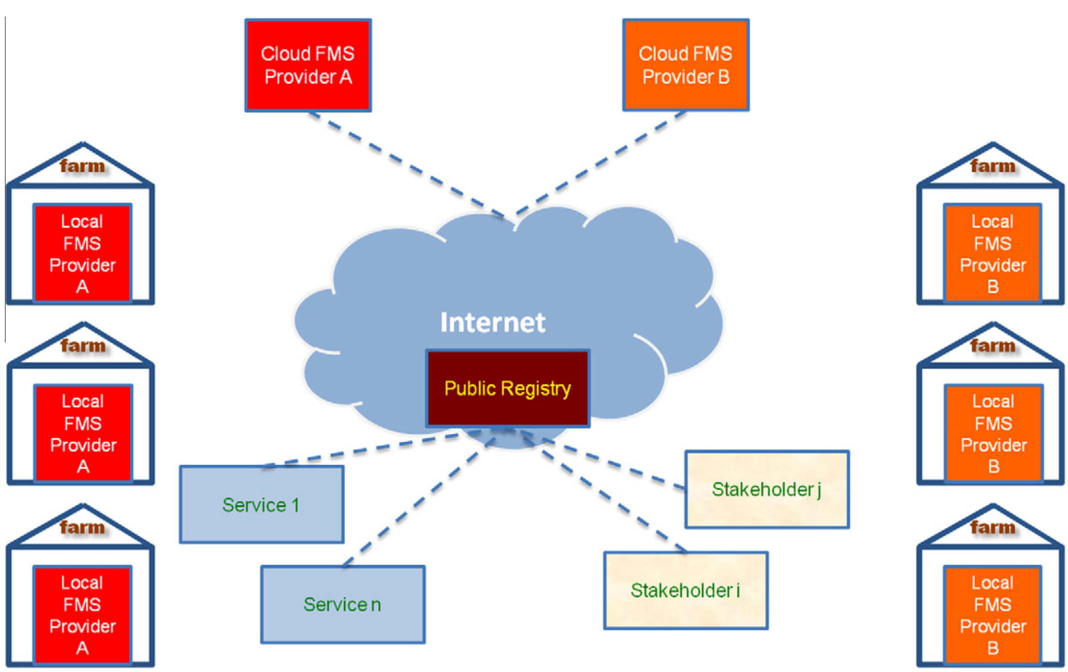

Fig. 4. Accessing services over the Internet.

Future Internet Public–Private Partnership Programme (FI-PPP) that is co-funded by the European Commission's Information Society and Media Directorate General within the ICT work programme of the Seventh Framework Programme.

With Call 1 of the FI-PPP programme, eight use case areas will define requirements for the Future Internet Architecture. Every use case is organized as a single project, where ''Smart AgriFood'' is one of the eight use case projects to provide requirements derived from the smart farming area (as well as for smart logistics and food awareness). All FI-PPP requirements are handled in the FI-WARE[4] project that builds the Core Platform for the Future Internet. FI-WARE defines and specifies the functionality of the Core Platform as a set of ''generic enablers'' (i.e., appropriate software modules), which are common to all future Internet applications. Those generic enablers are:

- Cloud Hosting – the fundamental layer that provides the computation, storage and network resources, upon which services are provisioned and managed.
- Data/Context Management Services – the facilities for effective accessing, processing, and analyzing massive streams of data, and semantically classifying them into valuable knowledge.
- Service Delivery Framework – the infrastructure to create, publish, manage and consume FI services across their life cycle, addressing all technical and business aspects.
- IoT Services Enablement – the bridge whereby FI services interface and leverage the ubiquity of heterogeneous, resource-constrained devices in the Internet of Things.
- Interface to the Network and Devices – open interfaces to networks and devices, providing the connectivity needs of services delivered across the platform.
- Security – the mechanisms that ensure that the delivery and usage of services is trustworthy and meets security and privacy requirements.

Looking at today's existing IT and communication solutions, a large variety of products and technologies could be applied for composing them to future farm management systems. Regarding the generic enablers for the future Internet, FI-WARE has provided an overview about so-called ''Baseline Assets'' supporting the concepts (FI-WARE Mediawiki, 2012). However, apart from assembling technological components, one of the main challenges in realizing a farm management system may be the definition of an open, standardized infrastructure that supports the integration of vendor independent solutions and services specifically for agriculture. Thus, in the context of ''Smart Agrifood'' we define a set of ''domain specific enablers''. These are software modules that are applicable in the agricultural sector. These enablers may be totally independent from the generic enablers (e.g., coordinating the execution of an farming advisory service) or base their operation on the functionality offered by the generic enablers. For example, it is expected that the generic enablers will provide the tools to perform statistical analysis. These tools can actually provide a library of generic functions (e.g., average value, deviation, etc.). A domain specific enabler for statistical analysis for the ''Smart Agrifood'' sector will use these generic functions to provide the required functionality for agricultural tasks. So, both the domain specific and the generic enablers will be a main part of the FMS.

5. Building the evolved FMS

As mentioned in the previous section, the generic and the domain specific enablers will play an important role for the design and implementation of the FMS. Thus, in this section we further discuss how these will be used by the proposed FMS functional architecture. Then, we proceed with the detailed presentation of the FMS functional architecture together with an operation example.

5.1. Adopting the generic and domain specific enablers concept

As mentioned in the previous section, FI-WARE defines six large areas for which generic enablers will be provided. Some of these will be so fundamental that will be applicable for a diverse set of tasks. Other enablers will have to be adapted and fine tuned for the agricultural case and thus they will provide advanced domain specific capabilities. In the subsequent subsection we analyze the six FI-WARE chapters and identify which are the generic and which are the domain specific enablers.

5.1.1. Cloud Hosting

The ''Cloud Computing'' concept is the basic *generic enabler* for ensuring the smart farming user requirements, providing scalable computation, software, data access, and storage services. With a cloud, users can rent software as a service or only an infrastructure (e.g. data storage), keeping investments for small companies low. The end-user does not have to care about the data backup, administration or maintenance of the infrastructure. In addition, as all data from any stakeholder (farmers, contractors, suppliers, business partners, etc.) are stored in the cloud, users can access and share data via fixed access (e.g. DSL) or mobile access (e.g. WiFi or cellular networks) from any place with portable devices. Of course, access policies have to ensure information security and integrity, allowing users to control permissions for accessing their data.

Additional services, like weather forecasting, can be rented as a service from the cloud and even integrated on the fly with other services. Application developers will have access to generic and standardized interfaces so they will not care about the behind lying cloud infrastructure and will be able to provide solutions and services with low development effort.

Morgan Stanley's blue paper (Holt et al., 2011) provides a comprehensive overview about suppliers and service providers in the cloud business with their different focus areas on infrastructure (e.g. Amazon's Enterprise Compute Cloud, EC2), Platform (e.g. Microsoft's Azure and Google's AppEngine) or Software (e.g. Salesforce.com Sales Force Automation application). In addition, telecommunication companies provide E2E cloud services including consulting, planning, design and delivery on top of their cloud-based infrastructure and communication networks. Thus, for smart farming, a lot of existing solutions and services could be applied already or designed according to the use-case or individual needs.

5.1.2. Data/context management

In future FMSs, huge amount of data has to be transferred, converted, stored, analyzed and accessed through the cloud. Further, instead of just processing ''data'', the meaning of the data is linked, allowing end-users to interpret the data in the context. As an example, soil-sensor data have context elements like ''soil temperature in degrees Celsius'' aligned with the location, the owner of the sensor, manufacturer, etc. The Future Internet therefore has to provide functional blocks that will be re-used as a service from the FMS. In FI-WARE functional blocks like big data analysis, multimedia analysis, semantic annotation, etc., are going to be totally integrated with intelligent agricultural management mechanisms that will appear as *domain specific enablers*.

Generic enablers that are specified to build generic intelligent services with a wider scope of appliance (e.g., social network analysis, opinion mining, mobility analysis, etc.) will be seen as

[4] http://www.fi-ware.eu/.

independent *generic enablers* to be used selectively from the domain specific modules.

5.1.3. Applications/services ecosystem and delivery

The primary task of the application/services ecosystem and delivery enabler is to improve the creation of the applications in FI, support various business models behind these applications, and enable them to be accessible from a variety of end-user devices. It covers the key business roles of Broker, Aggregator, Gateway, and Channel Maker.

The *Broker* function means establishing connections between the providers of various services and end consumers by creating the corresponding stores (individually owned) and, more generally, marketplaces offering goods from a number of stores. Examples for existing marketplaces are eBay or Amazon, while in the area of software applications there is a number of them (e.g., Apple Store, Android market). The FI-WARE will include *generic enablers* for discovering and installing services through appropriate registers and repositories and also it will provide for charging and billing mechanisms as well as SLA management functions.

Next, *aggregators* will provide composition and mashup functions, which are supposed to enrich the applications and make them more attractive to end-users. While current composition and mashup applications are mainly in the context of information and multimedia, the Future Internet aims to create *generic enablers* that will span applications and services as well. The mediation enabler will build the Gateway functionality as another *generic enabler*. Currently, there exist many data formats in the Internet, and this block will resolve the interoperability among them. While performing this function, the privacy of the data will be taken into account.

The benefits for the farming community by the adoption of these enablers are obvious. The goods produced on a farm will find a more direct way to the customers. The farmer can also benefit from advertising its products and using sophisticated services. For the recipients of goods, it will be significantly easier to search, browse and compare the offers.

Enablers, related to the Multi-Channel/Device Access; which account for ever-increasing variety of devices present in the current global network need to be integrated seamlessly. The Channel Maker block will encompass the functionalities of creating the channel/device specific interfaces. This can be reflected in adaptation to users' preferences and profiles. For the case of our system architecture this may end up as a *domain specific enabler* for discovering the best way to communicate with a stakeholder or any farming machinery.

5.1.4. Internet of Things

Currently technical solutions for monitoring, controlling and documenting agricultural farming processes, logistics, transportation, environments are just a few examples of the new wave of services available in proprietary FMISs (Sørensen et al., 2010). Different farm sensors, devices and actuators (''things'') are implemented using various communication standards. Existing farming technologies support a number of communication standards such as Zigbee, WiFi, and ISOBUS to exchange information (Thessler et al., 2011). These communication technologies are heterogeneous, making fluid intercommunication presently a hard task between devices.

The FI Internet of Things (IoT) is a visionary concept that is on its way to become reality also in the future FMS. The IoT generic enablers will provide the means to unify and interpret different network and communication protocols of ''things'' securely whilst addressing data needs, and service related functionalities. An instance of the FI IoT can be deployed in simple farm sensors, sensor gateways, and also in more sophisticated Internet frontends for monitoring and controlling different sensors and ''things''. This will enhance the functionalities of existing equipment with ubiquitous networking and innovative embedded systems making the physical world itself a relevant part of the information system in agriculture.

The potential applications are numerous and include monitoring of greenhouses, animals, and agricultural machines, and gathering of information from various sensors and RFID tags. Unfortunately, the variety of the technologies of interest presents also the main hindrance in deploying the concept on a great majority of farms. Namely, most of the solutions have small target areas, are proprietary and of an ad hoc nature. FI-WARE will create a unified architecture that might enable our farms to become smarter. To achieve this both *generic enablers* (e.g., for aggregating and managing sensors data) or *domain specific enablers* (i.e., provide middleware solutions to hide device specific details though appropriate APIs) are expected to be used.

The ''things'' inside Internet can generate a very high amount of data. Consider for example, the project of a Dutch start-up Sparked with sensors implanted in cows' ears that can measure its health conditions and transmit this data to the farmer. Such an application is expected to generate 200 MB of data per year and per cow (Evans, 2011). Therefore, it is particularly important to determine where and how will all this data be stored. This issue will be partially covered by the IoT Data Handling generic enablers. Data handling in IoT will include also the security aspects. For example, it might be preferred to keep some sensitive data only at the farmer's premises and not in the cloud.

As far as IoT communication aspects are concerned, the technologies of interest, such as net-worked RFID systems, wireless sensor networks, near-field communication (NFC), and machine-to-machine (M2M) systems will be integrated in a broader IoT architecture. The distributed things will communicate with the gateway (typically on the farmer's premises) using interoperable solutions. Finally, the enabler for the management of IoT resources will allow an easier discovery of the devices of interest on farms by providing an efficient resolution infrastructure. This *generic enabler* will in combination with the IoT Communication *generic enablers* resolve the addressing issues of present sensors, by applying the address translating methods.

5.1.5. Interface to networks and devices

For the FI-WARE to offer really advanced solutions it is imperative to specify interfaces with the end devices as well as the underlying telecommunications network infrastructure.

The Connected Devices Interface (CDI) *generic enabler* is mainly focused on creating smart communication pipes by exploiting the status of devices (e.g., the battery status, device features, location, etc.).

The Network Information and Control (NetIC) *generic enabler* utilizes the information about the network in order to optimize it and ensure that SLAs promised to the farmer are satisfied. The service provider will obtain the information about the interfaces, topologies, statistical information (paths, traffic), and control of paths and traffic.

The Service, Capability, Connectivity, and Control (S3C) *generic enabler* will support the necessary modification of the core network (Evolved Packet Core – EPC) for FI applications. The EPC was developed as an IP-based control platform that integrated various wireless access technologies (UMTS, LTE, WiFi, etc.). The envisioned applications in Smart Farming, however, might present a significant burden for the EPC. Namely, the EPC was designed with a primary goal of supporting the human-to-human communication. The new FI-enabled services in Smart Farming will assume a proliferation of M2M communication (e.g. among sensors on the farms, gateways, computers in the cloud, etc.). This type of communication is characterized by a traffic model that is very different in comparison to communication among humans. The number of connections is much higher, while the amount of transmitted data per call is typically small (consider a simple temperature measurement).

5.1.6. Security

Security is the most essential functionality to give trust in usage of the future Internet. The farmers have to be sure, that private data remains private, although stored in the cloud. Further, fraud and intrusion detection mechanisms and identity management have to be ensured through the future Internet. Although relevant security functionalities have been identified, several functions still need to be further worked out:

- Authentication and authorization: Single-sign-on for several services offered in the cloud.
- Privacy management: Mechanisms ensuring that applications cannot process data nor subscribe to data without the consent of its owner.
- Service registry and repository: Management of identities and authorization for the publication/management of service descriptions in the repository.
- Revenue settlement and sharing system (applications/services ecosystem and delivery): Confidentiality and integrity during the payment process.

5.2. The functional architecture of a FMS

As mentioned before, the proposed FMS consists of two main modules. The first one (Cloud FMS) is placed in the cloud, and consists of the functional blocks presented in Fig. 5. Its structure follows a layered service oriented architectural approach initially presented in Teye (2011). The basis of the FMS is the use of the aforementioned generic enablers and also their adaptation to domain specific enablers for the sake of agricultural tasks. Thus, in the subsequent text we illustrate how these separate six areas of generic enablers can be combined to provide a complete a coherent tool for agriculture.

The top layer supports different interfaces for different stakeholders and their devices. The next layer, called service management layer, consists of the service flow description in XML, the intelligence to invoke collaborative services as described in the service flow description and the enterprise private service registry. In this registry we store all information related to services (service description, requirements, SLA information, tariff policies, etc.) that can be offered to the end users. These services can be offered either by the FMS provider or external service providers. In the latter case the FMS provider and the external service provider need to establish an explicit association for security and accounting reasons. The following layer consists of the FMS Controller that is the heart of the FMS and is explained below and a number of *generic enablers* as they have been described in the previous sections (vertical blocks). These generic enablers include

- External collaboration: used for the discovery of services not registered in the serving FMS (i.e., provides a link with the public registry presented in Fig. 4).
- Opinion making: analyze user opinions using simple voting and reputation schemes or more advanced opinion mining schemes.
- Social network analysis: analyzes users' social interactions to understand their social relationships and communities and offers personalized services.
- Real time recommendations: analyses the behavior of a user when using a service to make a recommendation for a new product a user would be interesting in subscribing and using.
- Web analysis for profiling: derive profiles that can be extracted from the monitored and analyzed activity of a user.
- Network infrastructure cooperation: Link to the underlying network infrastructure using the NetIC and S3C interfaces to optimize the FMS performance.

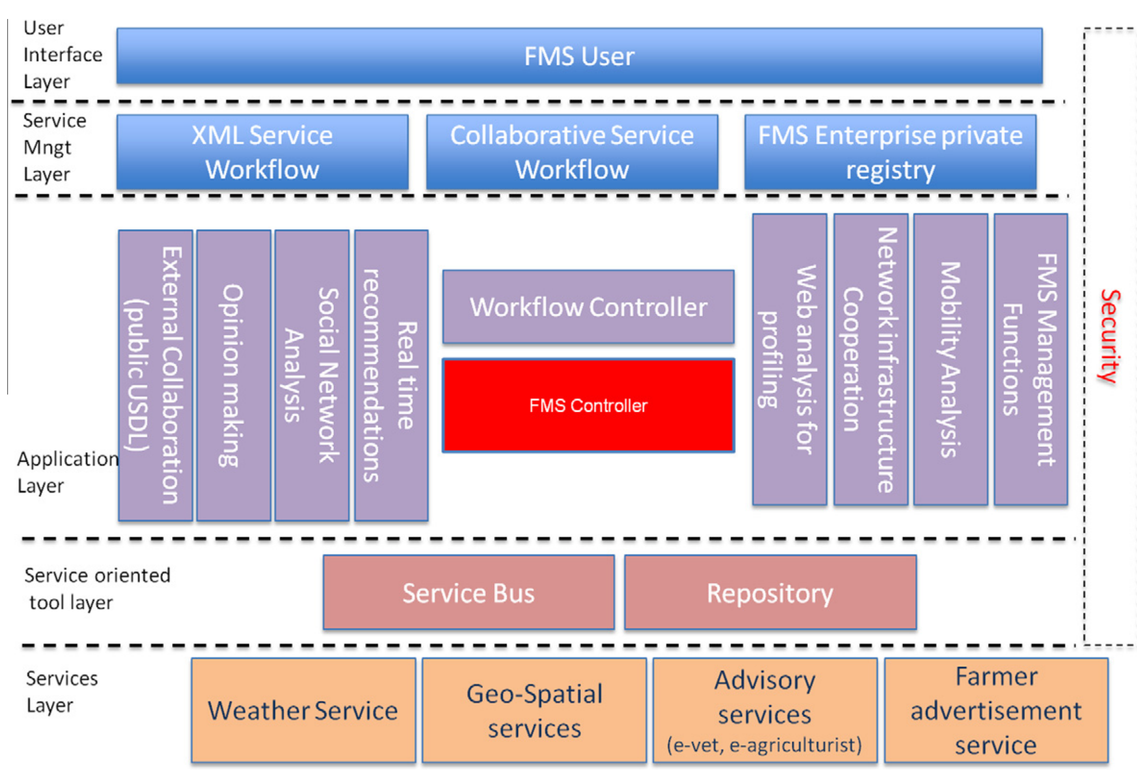

Fig. 5. The structure of cloud FMS.

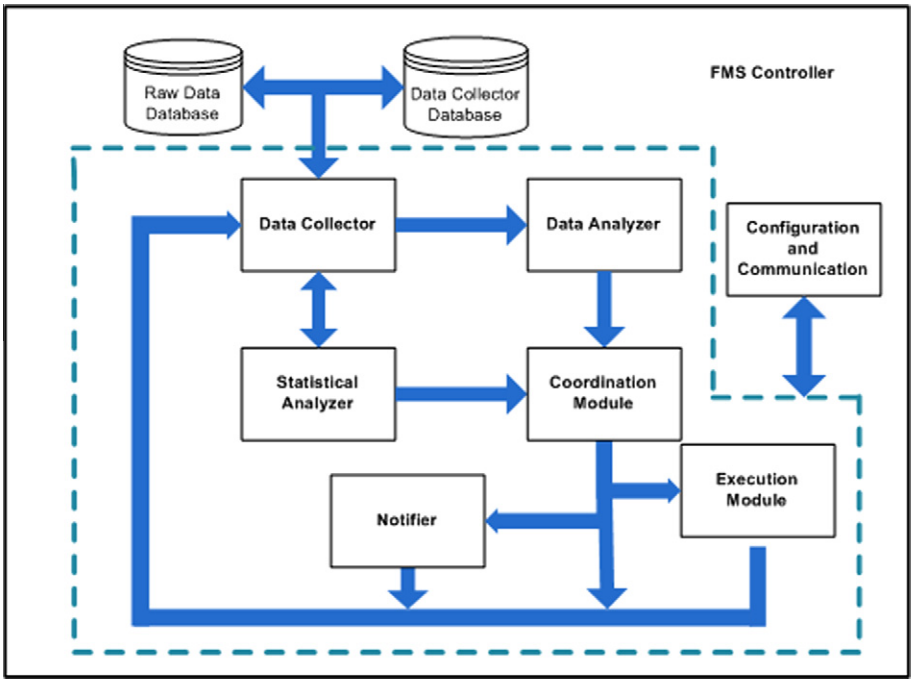

Fig. 6. FMS controller's modules communication.

- Mobility analysis: transforms geo-located user activity information into a mobility profile of the user.
- FMS management functions: Overall management functionality for the FMS platform e.g., "rewiring" the system in case of a malfunctioning or slow Internet connection with the farm charging and billing, SLA management, etc.

The Workflow Controller is used as a message dispatcher that integrates all these generic enablers with the FMS controller. The next layer, which is the service oriented tool layer, invokes the different services each customer is entitled to. This layer is actually a realization of an Enterprise Service Bus (ESB) that is used for designing and implementing the communication among interacting applications (Murakami et al., 2007). Finally, the bottom layer contains all services registered in the specific FMS. In almost all these layers security mechanisms are required.

As far as the registered services to the FMS that the farmer can use, our use case analysis has identified that apart from well expected ones such as weather or geospatial services, a number future services desired by the farmers may indicatively include:

- automated advisory services (e.g. agriculturist, veterinarian services) that will analyze collected information and suggest appropriate actions (e.g. spray a field),
- task plan analyzer services for a number of stakeholders (e.g. organize the tractors of a spraying contractor in order to fulfil the contracts in an area),
- state's policies and information service, meteorological data, geospatial data service, farming machine support services (e.g. dynamic firmware update),
- end-user food awareness services that provides information on how the crops have been produced (e.g. treatment with chemicals, treatment with fertilizer, cultivation methods),
- crop availability service that provides information to the supply chain on when the crop will be available as well as the expected quantity.
- a stakeholder advertisement service that will contain a profile service with which a stakeholder (e.g., farmer, spraying contractor, buyer, etc.) could provide additional information about himself by uploading data, such as photos or videos.

This indicative list suggests that our architecture provides the means to create a marketplace similar to the ones created by Apple and Google, where stakeholders will have access to a number of trustworthy, sophisticated and eventually cheap services. Also, using appropriate generic enablers, service composition and mash-up will be also possible bringing to stakeholders a previously unthinkable rich ecosystem of services and products.

As mentioned earlier, the FMS Controller is part of the application layer for a Cloud FMS. It consists of a number of functional modules (Fig. 6) that have been defined through the analysis of the use cases. Their implementation will be heavily based on the generic enablers specified by FI-WARE Mediawiki (2012) (e.g., big data analysis, multimedia analysis, semantic annotation, etc.) adapted for the farming management cases (i.e., adapt them to domain specific enablers). The arrows among the sub-modules identify their interfaces and the flow of messages during operation. The functionality of the sub-modules is defined in Table 2.

Note that the dashed lines denote that all the above mentioned sub-modules communicate with the outside world through the Configuration and Communication sub-module. From the above sub-modules the Raw Data DB, the Data Collector DB, as well as the Data Collector, the Data Analyzer and the Statistical Analyzer are all part of the Information and Knowledge Building phase presented in Fig. 2. The Coordination module is related to the decision phase of the same figure, while the Notifier and the Execution module are part of the execution phase. The monitoring phase is performed either by the Local FMS or directly from any farming machinery that is configured to transmit data directly to the Configuration and Communication module of the FMS Controller.

Note that a third party service (e.g. advisory service, task plan analyzer, etc.) is integrated in the FMS platform through the FMS controller. At a minimum, each service should have an interface with the configuration and communication module for security purposes and for having access to the related set the farming data (e.g. from sensors) stored in the database. Any service may also make use of any of the other sub-modules of the FMS controller if it wishes to. These other modules provide generic functions related to farming so as to simplify the work of service developers.

Table 2
FMS controller's modules functionalities.

Functional block	Supported functionality
Raw Data DB	– Used for storing raw data as collected from the sensors, farming machinery, tracking systems, external services e.g., meteorological data – These data are the property of farmer and can use them when switching from one FMS provider to another
Data collector DB	– Used for placing all processed data (e.g., after statistical processing of raw data) and information related to a farmer (e.g., advices from an advisory system, executed actions, the results of these actions, etc.) – It contains knowledge produced by the cognitive cycle (monitor, decision, execution, learning) or directly by a stakeholder (e.g. farmer, machinery manufacturer). Knowledge storage in the database is similar to the paradigm applied in the case of text-mining (Chakrabarti, 2002). During the latter, text is represented as a vector and associated with a class which in turn can be associated with any set of actions (i.e. web search). Following the analogy, in our case, knowledge is represented as a vector that can be classified and associated with any set of rules or actions
Data collector	– Used to transfer data to and from the Data Collector Database and the Raw Data Database – Provides information for further processing to the Data Analyzer and the Statistical Analyzer module and also communicates with the Notifier.
Data analyzer	– Involved with the processing and analysis of different types of data and different types of context. It contains a multimedia analyzer. – Checks periodically if some rules are violated or not – Checks if some received values are not inside an expected range – Communicates with the coordination module and the statistical analyzer
Statistical analyzer	– It processes an amount of data using statistical functions – Uses data mining techniques to inform about the system's performance – Used to identify malfunctioning farming machinery or equipment (e.g., sensor).
Coordination module	– Receives input from the Data Analyzer and the Statistical Analyzer and has the "intelligence" to handle simple situations (e.g., temperature increase inside a greenhouse) – Coordinates the decisions reached by services the farmer is currently using – Responsible for conflict resolution among services – Triggers the execution module and the notifier. – It is configurable from the Statistical Analyzer module or directly from external entities (e.g., farmers, equipment manufacturers) since it allows them to install "knowledge" in the form of pre-defined rules
Notifier	– Used to inform stakeholders (e.g. farmers, buyers, spraying contractors, agriculturists, etc.) – Adapts any type of information to an appropriate form for end-user's device
Execution module	– Used for actions that can be executed automatically (e.g. open the windows start the ventilation system, initiate a firmware update, etc.). For those actions that cannot or the farmer wishes not to be executed automatically, the Notifier is responsible to inform the farmer with the appropriate information
Configuration and communication	– Sets the communication channels to collect raw data from the sensors and the farming equipment/machinery – Communicates with services provided from other parties – Used to configuring all other modules of the system (e.g. set a threshold to the Data Analyzer) – Is responsible for authentication an authorization – Is the message dispatcher for all other modules of the FMS controller

Finally, Fig. 7 presents the internal structure of the Local FMS. It is expected to have similar functions to its cloud counterpart but in a limited version (e.g., it does not contain any statistical analyzer). The Local FMS mainly aggregates sensor values collected through its interface of the configuration and communication module, it can send commands through the same interface and also it can take control of the overall management if the link to the Internet is not operational.

5.3. Operation example

In this subsection we present a simplified operation example of the abovementioned architecture. In Fig. 8, we illustrate the case where a single sensor transmits its data periodically in the Local FMS. These values can be aggregated and forwarded to the FMS Controller. The FMS Controller processes these data and if the Coordination sub-module cannot handle a situation, it may suggest to the farmer to register to a new type of services (e.g. an advisory service). The farmer will send a generic message to the FMS Controller asking for a list of services that match certain requirements (e.g. "find all tomato related greenhouse advisory services specialized for Greece that have been ranked as the top 5% in their category"). The FMS Controller will discover the services that are already associated with the FMS provider by communicating with the FMS Enterprise Private Registry. It will further communicate with the opinion making module to filter out those services that have low ranking, based on the opinion of other farmers. Eventually, the farmer will be notified for a possible list of services from which he will select one. After registering with this service the farmer will allow the service to have access to his desired set of stored data and start receiving advices and notifications.

6. Conclusions and future work

In this paper we have described a FMS functional architecture that utilizes advanced Future Internet characteristics. Its main characteristics include apart from the support of the typical farming procedures, the seamless support and integration of different stakeholders and services, interworking with the networked infrastructures and the introduction of autonomic and cognitive elements in the overall management process. This architecture was the result of an extensive analysis of a significant number of use cases by researchers from diverse fields of expertise.

With this work we attempt to enable a farmer to step into a new reality, where he becomes an actual "node in an agricultural worldwide web". Optimized methods for managing all tasks inside a farm as well as pioneered Future Internet services for entering the global markets are also provided. Easy access to information and advices, convenient communication among all stakeholders along the food chain, efficient combination and management of

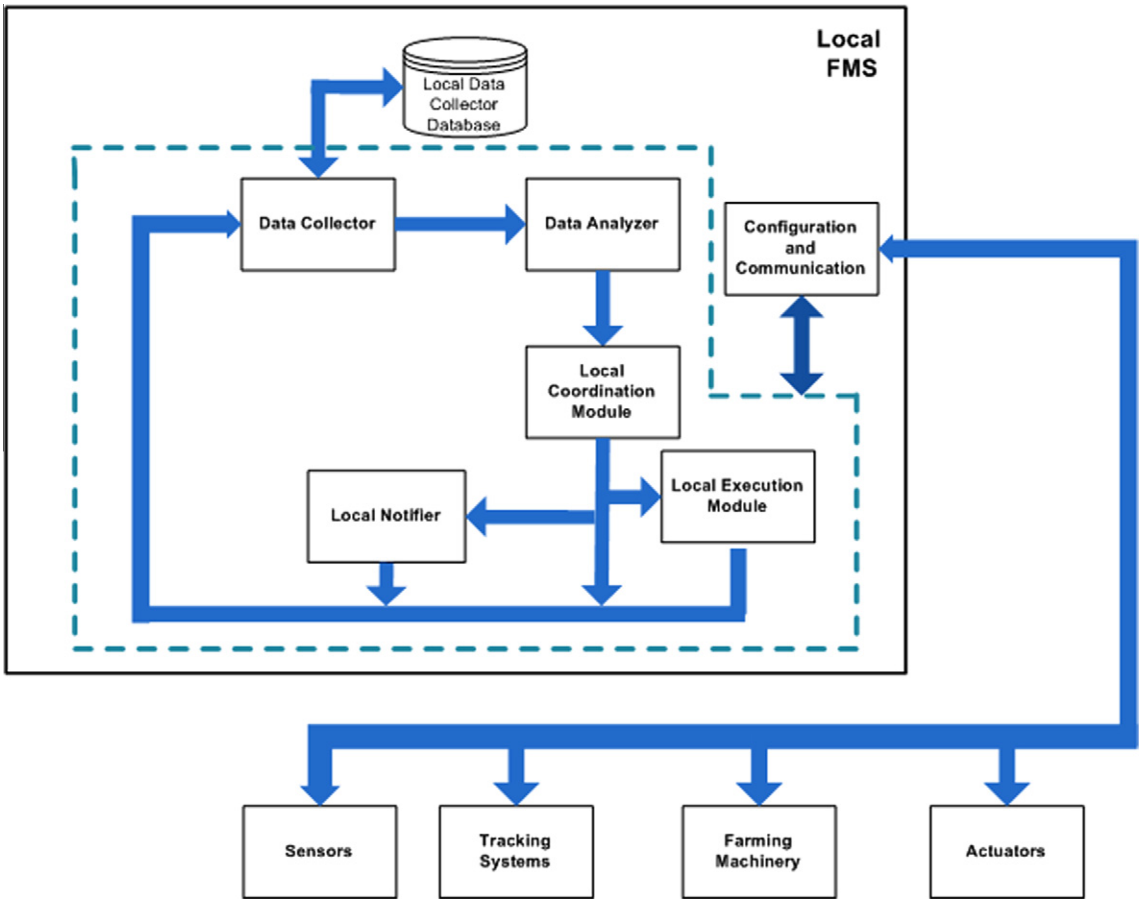

Fig. 7. Structure of the local FMS.

diverse data that come from different sources, effective organization of the huge amount of information are only some of the issues that are considered in the proposed architecture.

In our next steps we plan to implement and test this architecture for at least two pilots. The first one involves the management of a greenhouse while the second one is deals with the automatic

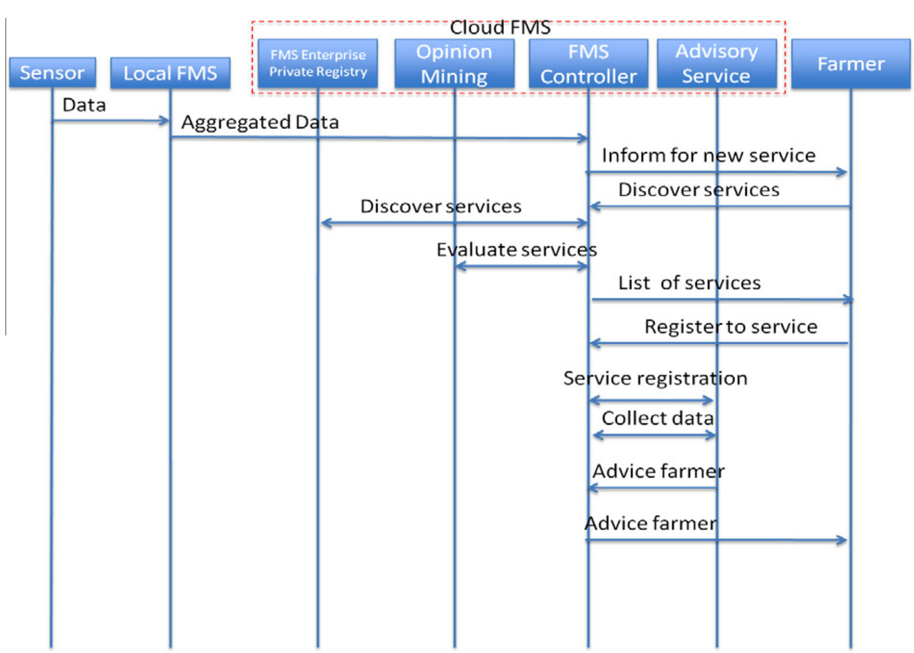

Fig. 8. A simplified operation for the FMS system.

Table 3
Functional requirements.

Functional requirement	Functional block/FI functionality
Gathering multimedia information for further analysis should be possible	Data collector
Collection of sensed data should be performed	
Collection of news related to a farm's operation should be performed	
Tracing/tracking capabilities should be enabled	
Data aggregation should be dynamically performed	
Data coming from neighboring infrastructures should be gathered for further use	
Collection of information that refer to farm products should be possible	
Identification of extraneous and foreign bodies should be performed	Data analyzer
Multimedia analysis should be performed	
Efficient indexing techniques should be used to have access to stored data	
Data processing for animal or data collected data should be possible	
All transactions between different services and modules will be used for making the system learn	Statistical analyzer
Statistical process of data should be done	
Data mining techniques should be enabled	
Faulty sensors should be detected	
Fault operation of agricultural machines should be detected	
Predictions about estimated yield should be possible	Coordination module
Recognize if products are developed properly should be performed	
Planning the daily tasks should be supported	
Plan for cooperative harvest should be produced	
Reorganization of cooperative harvest when a problem occurs should be performed	
Disease forecast as well as recommendations should be available	
Decision for recommending cancelation of scheduled tasks inside the farm when weather is not proper should be given	
Recommendations should be given for handling alarms	
Suggestion should be available according to statistical data	
Cultivation plans should be produced	Notifier
Farmers notify the authorities about the production (e.g., milk quota)	
Receive notifications from the authorities about the national quota	
The spraying system should inform the machine operator for existing alarms	
Notifications should be sent when it is predicted that an event could take place (e.g., plants infected by a disease)	
Notifications about farming processes (e.g., spraying) should be sent to interested parties	
Notifications with high priority should be sent using appropriate protocols	
Notifications should be sent to every stakeholder when an abnormal state takes place in a farm	
A farmer should be informed about news he is interested in	
Automatic end – terminal selections should be available	
Removal of extraneous and foreign bodies should be performed	Execution module
Remote control of agricultural machines or equipment should be possible	
Automatic firmware updates should be supported	
Automatic reaction of the system should be present	
Actions should be performed in order to isolate malfunctioning equipment (e.g., sensors)	
Automatic firmware updates should be supported	Configuration and communication
Configuration of parameters for existing methods should be available at any time	
Only authenticated and authorized users would have access to information	
The system must be re-configurable based on new data received over the Internet	
Subscriptions should be provided for stakeholders to all the available services	
Registrations could be performed for every service	
External services should be able to access farmer's information	
Self-configuration mechanisms should take place	FI – Resource management GE (IoT)
Mechanisms should be developed for managing and controlling all up – coming services and applications	FI – GEs for Composition and Mashup
Cooperation between different FMSs should be possible	FI – Service, Capability, Connectivity and Control GE (S3C)
Geo – located users activity data and mobility profiles should be available	FI – Mobility analysis GE
A stakeholder may be able to switch end – terminals on the fly	FI – GEs for Multi-channel and multi-device access
A stakeholder would like to give his opinion about another stakeholder or the overall system	FI – Opinion making GE
Proper notifications should be sent to neighboring stakeholders for any emergencies	FI – Social network analysis GE
Periodically updates of different profile should be possible depending on different types of feeds.	FI – Behavioral: and web profiling GE
Devices linked to the system announce their capabilities	FI – Connected Device Interfaces GE (CDI)
Adaptability of content for different devices should be possible	FI – Connected Device Interfaces GE (CDI)
Access to a stakeholder's information should be made in a secure way	FI – Security GEs
Local system has to take control when internet connection fails	FI – Cloud Proxy
Real-time recommendations should be sent according to stakeholder's behavior	FI – Real-time Recommendations GE
Collecting and processing multimedia information for identifying significant events should be available.	FI – Multimedia analysis GE
Simplify/automate the management of the FMS	FI – Service, Capability, Connectivity and Control GE (S3C), Network Information and Control GE (NetIC) Connected Device Interface – CDI GE
Finding other players and link to them should be available	Market place and store GE

coordination and management of tractors in open fields under different scenarios. These cases will allow us to test in real environments the suitability of existing technologies as well as the need for new approaches and improvements especially under the scope of the Future Internet.

Acknowledgments

This work was performed in project SmartAgriFood that has received research funding from the Community's Seventh Framework program. The SmartAgriFood project is part of the Future Internet Public–Private Partnership (FI-PPP) program. This paper reflects only the authors' views and the Community is not liable for any use that may be made of the information contained therein. The contributions from colleagues of the SmartAgriFood consortium are hereby acknowledged.

Appendix A

An example of functional requirements and their mapping to functional Blocks or FI generic enablers (see Table 3).

References

Alexander, I., 2004. Introduction: scenarios in system development. In: Alexander, Maiden (Eds.), Scenarios, Stories, Use cases: Through the System Development Life-Cycle. John Wiley & Sons, Chichester, UK.

Allen, J., Wolfert, J., 2011. Farming for the Future: towards better information-based decision-making and communication – Phase I: Australasian stocktake of farm management tools used by farmers and rural professionals. New Zealand Centre of Excellence in Farm Business Management, Palmerston North. <http://edepot.wur.nl/194811>.

Batte, M., 2005. Changing computer use in agriculture: evidence from Ohio. Comput. Electron. Agric. 47 (1), 1–13.

Beck, H., 2001. Agricultural enterprise information management using object databases, Java, and COBRA. Comput. Electron. Agric. 32 (2), 119–147.

Carroll, J.M., 1995. Introduction: the scenario perspective on the system development. In: Carroll, J.M. (Ed.), Scenario-based Design: Envisioning Work and Technology in System Development. Wiley, New York.

Chakrabarti, S., 2002. Mining the Web: Discovering Knowledge from Hypertext Data. Morgan-Kauffman Publishers, India.

Csótó, M., 2010. Information flow in agriculture-through new channels for improved effectiveness. J. Agric. Inform. 1 (2), 25–34.

Duncan, M., Harshbarger, E., 1979. Agricultural productivity: trends and implications for the future economic. Econ. Rev. (September), 1–12.

Evans, D. 2011. The Internet of Things – How the Next Evolution of the Internet is Changing Everything. CISCO White Paper.

FI-WARE MediaWiki: Materializing the FI-WARE Vision. <http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/Materializing_the_FI-WARE_Vision>.

Fountas, S., Wulfsohn, D., Blackmore, S., Jacobsen, H.L., Pedersen, S.M., 2006. A model of decision making and information flows for information-intensive agriculture. Agric. Syst. 87, 192–210.

Holt, A., Weis, K., Huberty, K., Gelblum, E., Flannery, S., Devgan, S., Malik, A., Rozof, N., Wood, A., Standaer, P., Meunier, F., Lu, J., Chen, G., Lu, B., Han, K., Khare, V., Miyachi, M., 2011. Morgan Stanley Blue Paper: ™Cloud Computing Takes Off. <http://www.morganstanley.com/views/perspectives/cloud_computing.pdf>.

Kephart, J., Chess, D., 2003. The vision of autonomic computing. IEEE Comput. 36 (1), 41–50.

Lewis, T., 1998. Evolution of farm management information systems. Comput. Electron. Agric. 19 (3), 233–248.

Martin-Retortillo, M., Pinilla, V., 2012. Why did agricultural labor productivity not converge in Europe from 1950 to 2006? In: Proceedings of the 2012 Economic History Society Annual Conference, University of Oxford, Friday 30 March-Sunday 1 April. UK.

McCown, R.L., 2012. A cognitive systems framework to inform delivery of analytic support for farmers£ intuitive management under seasonal climatic variability. Agric. Syst. 105 (1), 7–20.

Murakami, E., Saraiva, M., Ribeiro, M., Cugnasca, E., Hirakawa, R., Correa, P., 2007. An infrastructure for the development of distributed service-oriented information systems for precision agriculture. Comput. Electron. Agric. 58 (1), 37–48.

Nikkilä, R., Seilonenen, I., Koskinenen, K., 2010. Software architecture for farm management information systems in precision agriculture. Comput. Electron. Agric. 70 (2), 328–336.

Nuthall, P., 2009. Modelling the origins of managerial ability in agricultural production. Aust. J. Agric. Resour. Econ. 53, 413–436.

Olmstead, L., Rhode, P., 2009. Conceptual issues for the comparative study of agricultural development. In: Lains, P., Pinilla, V. (Eds.), Agriculture and Economic Development in Europe since 1870. Routledge, London, pp. 27–52.

Robertson, S., 2004. Scenarios in requirements discovery. In: Alexander, Maiden (Eds.), Scenarios, Stories, Use Cases: Through the System Development Life-Cycle. John Wiley & Sons, Chichester, UK.

Salami, P., Ahmadi, H., 2010. Review of Farm management Information Systems (FMIS). New York Sci J 3 (5), 87–95.

Samaan, N., Karmouch, A., 2009. Towards autonomic network management: an analysis of current and future research directions. IEEE Commun. Surv. Tutorials 11 (3), 22–36.

Simon, H., 1996. The Sciences of the Artificial, third ed. MIT Press, Massachusetts, USA.

Sørensen, C.G., Fountas, S., Nash, E., Pesonen, L., Bochtis, D., Pedersen, S.M., Basso, B., Blackmore, S.B., 2010. Conceptual model of a future farm management information system. Comput. Electron. Agric. 72 (1), 37–47.

Sørensen, C.G., Pesonen, L., Bochtisc, D.D., Vougioukas, S.G., Suomi, P., 2011. Functional requirements for a future farm management information system. Comput. Electron. Agric. 76 (2011), 266–276.

Subramanian, L., Surana, S., Patra, R., Ho, M., Sheth, A., Brewer, E., 2006. Rethinking Wireless for the Developing World. In: Proc. of the 5th Workshop on Hot Topics in, Networks (HotNets-V).

Thessler, S., Kooistra, L., Teye, F., Huitu, H., Bregt, A., 2011. Geosensors to support crop production: current applications and user requirements. Sensors 11, 6656–6684.

Teye, F., 2011. A Conceptual Model for Collaboration-based Management Information Systems. Master Thesis, Helsinki, Metropolia University of Applied Science.

Wang, N., Zhang, N., Wang, M., 2006. Wireless sensors and food industry: recent Development and future perspective. Comput. Electron. Agric. 50, 1–16.

Wolfert, J. 2002. Sustainable Agriculture: How to Make it Work? A Modeling Approach to Support Management of a Mixed Ecological Farm. PhD-Thesis, Wageningen University, Wageningen, pp. 278.

Wolfert, J., Verdouw, C., Verloop, C., Beulens, A., 2010. Organizing information integration in agri-food – a method based on a service-oriented architecture and living lab approach. Comput. Electron. Agric. 70, 389–405.