In **Information Technology in Environmental Engineering (ITEE 2007)**, (Marx Gómez, J., Sonnenschein, M., Muller, M., Welsch, H., and Rautenstrauch, C, Eds.), Oldenburg, Germany, Springer-Verlag, March 2007, pp.417-432.

# Semantic modelling in farming systems research

## The case of the Agricultural Management Definition Module

Ioannis N. Athanasiadis[1], Sander Janssen[2], David Huber[3], Andrea Emilio Rizzoli[1], Martin van Ittersum[2]

[1] Dalle Molle Institute for Artificial Intelligence (IDSIA), USI/SUPSI, Galleria 2, CH-6928 Manno, Lugano, Switzerland
{ioannis,andrea}@idsia.ch
http://www.idsia.ch
[2] Plant Production Systems, Wageningen University, Haarweg 333, 6700 AK Wageningen, The Netherlands
{sander.janssen, martin.vanittersum}@wur.nl
http://www.pps.wur.nl
[3] AntOptima SA, Fusoni 4, CH-6900 Lugano, Switzerland
david.huber@antiptima.ch
http://www.antoptima.ch

**Abstract.** Farming Systems Research studies agricultural systems and their interaction with the natural environment and ecosystems. Agro-ecosystems are highly complex due to the many feedbacks between natural processes, high geographical diversity and human factors involved both as the farmer's decisions at farm household level and as the policy implementations at regional, national or European levels. This paper presents a novel approach for developing an Agricultural Management Definition Module (AMDM), by exploiting ontologies and semantic modeling. Specifically, a declarative approach has been utilized for conceptualizing farming systems and the management alternatives of a farm household. This conceptual model has been implemented as an ontology that ultimately has been used as the basis for software development and integration. This paper presents in detail the methodology used for developing AMDM and a real-world installation, part of the SEAMLESS integrated project.

## 1 Introduction

### 1.1 Managing the complexity of farming systems

Farming Systems Research studies the agro-ecosystems, that is agriculture and its interaction with other ecosystems and society, at the farm level. The agro-ecosystems are highly complex (Kropff et al., 2001) due to the many feedbacks between natural processes, human factors, high geographical diversity in agro-ecosystems and the limited knowledge on some of the processes. Frequently, models are utilized to simulate the agro-ecosystems and further the understanding of agro-ecosystems. This has resulted in an abundant number of field, cropping system or farm level models, each developed for specific purposes. These models are realized as software components that are hardly re-usable and it is difficult if not impossible to integrate them with other models, in order to perform integrated analyses (Rizzoli et al 2005, Athanasiadis et al. 2006). One of the reasons for this, is the poor semantics that usually characterize farm model implementations. In this paper, we demonstrate how ontologies can help to formalize the knowledge captured in these models; in order to subsequently facilitate model knowledge re-usability and exchangeability.

## 1.2 The SEAMLESS integrated project

European agriculture and rural areas continuously change as a result of an enlarging EU, WTO agreements, introduction of novel agro-technologies, changing societal demands and climate change. Efficient and effective agricultural and environmental policies are needed to support sustainability of European agriculture and its contribution to sustainable development of society at large. Assessing the strengths and weaknesses of new policies and innovations *prior to* their introduction, i.e., 'ex-ante integrated assessment', is vital to target policy development for sustainable development. The SEAMLESS integrated project (http://seamless-ip.org) aims at developing a computerized, integrated and working framework (SEAMLESS-IF) to assess and compare, ex-ante, alternative agricultural and environmental policy options, allowing analysis across different scales (from field, farm to region and EU), dimensions of sustainability (economic, social, environmental and institutional) and for a broad range of issues and agents of change.

SEAMLESS-IF will be an open and modular framework and will offer the flexibility to analyze a wide range of issues (Van Ittersum et al. 2006). For specific questions to be analysed, a subset of models and tools out of the broad range available within SEAMLESS-IF can be used. It will require scientific and technical breakthroughs to enable model integration across scales, disciplines and issues. The *SEAMLESS integrated modelling framework* is aimed to be a platform for the development of integrated applications and used by researchers and scientists to produce applications and outputs for policy makers. Research groups will be able to develop agricultural, environmental, economic and social models, at different scales. These models will be seamlessly integrated, while maintaining the logical independence of data, models and simulation/optimisation algorithms. This will be achievable through the development of *independent components* and a *declarative approach to modelling*. Such an approach will permit the clear delegation of tasks within a multidisciplinary research team, strengthening collaboration and improving integration.

### 1.3 Ontologies, Knowledge Bases and Semantic Web

The last few years, ontologies, knowledge bases and the semantic web attract the interest of the research community. An ontology in computer science is considered as a specification of a conceptualization (Gruber 1993). It's a formalization that could be expressed in a machine readable format, i.e. as the Web Ontology Language (McGuinness & van Harmelen, 2004). This provisionally allows a software system to "comprehend" a conceptual schema and makes it possible to reason on it. A knowledge base is the result of expressing the information related to a domain in line with a given domain ontology. Typically this activity involves instantiating data with respect to an ontological definition. Ontologies can be seen as a medium for open software environments (Willmott et al. 2002), where software agents provide semantic web services under strict contracts, as for example in the AgentLink project.

In this background, we envision the development of SEAMLESS-IF as an open modular simulation environment for agricultural, science and policy. In SEAMLESS-IF, modellers are considered as communities of "knowledge workers" in the fashion of Warren (2006). Knowledge captured in environmental models is exposed using ontologies and ultimately using the semantic web can be reused, and combined properly for integrated studies.

The objective of this manuscript is to demonstrate the potential usefulness of semantic modelling in capturing the complexity of the Farming Systems by making knowledge relationships explicit. The development of an ontology formalizing the conceptualization of farming systems considered in SEAMLESS and the use of a Knowledge Manager for linking models and applications together is explained. In this way a Knowledge Manager mediates between various data sources and model algorithms, ensuring data integration based on explicit semantics.

## 2 The Agricultural Management Definition Module

### 2.1 Problem definition

A pivot element of the SEAMLESS modelling framework is farm level modelling. Farm level modelling aims to assess the impact of policy changes and technological innovations on farmer behaviour now and in the future. To achieve these goals it is required to specify the options available to a farmer for using his/her resources and satisfying his/her objectives.

This set of options is the agricultural activities that a farmer can apply. On a purely arable farm, agricultural activities constitute of options related to growing different crops with a range of alternative management practices. In this paper, we define an agricultural activity as a coherent set of crops (a crop rotation) with associated crop management and corresponding inputs, e.g. fertilizer, seed, pesticides, and outputs, e.g. marketable products, production of feedstuffs for on-farm use and environmental effects (Ten Berge et al., 2000; Van Ittersum and Rabbinge, 1997). A rotation is a succession of crops in time (cropping sequence) and space (cropping pattern), where the last crop is the predecessor of the first crop (creating a loop). Crop management is a complete set of agronomic inputs (e.g. management practices) characterized by type, level, timing and application technique (Van Ittersum and Rabbinge, 1997).

A distinction can be made between current and alternative agricultural activities. Alternative agricultural activities are agricultural activities that are not currently used, but are technically feasible alternatives for the future, often technological innovations or newly developed cropping practices, while current activities are agricultural activities that are currently being practiced and can be derived from observed data.

In a farm system model, mathematical programming (usually linear constraints and a non-linear objective function – Janssen and Van Ittersum, 2006) is used to ‘simulate’ the allocation of current and alternative activities to the available resources, while satisfying the objective and meeting the policy constraints.

## 2.2 Abstract architecture and provided services

The main objective of the **Agricultural Management Definition Module** (*AMDM*) is dual. First is to describe, generate and quantify alternative and current agricultural activities that can be evaluated by a dynamic crop simulation model (*Agricultural Production Externalities Simulator: APES*) (cf. Van Ittersum and Donatelli, 2003) in terms of yields and environmental effects. Second is to generate a set of fully quantified agricultural activities that can serve as inputs to a farm level optimization model in which the possible activities are confronted with farm endowments and farmer’s objectives (*Farm System Simulator: FSSIM*). In this respect, AMDM serves both biophysical and optimization models by preparing feasible/possible agricultural activities to be simulated or optimized respectively, as shown in Fig. 1. Specifically, in this paper we present the part of the AMDM that formulates alternative agricultural activities for arable farming systems. In principle, a similar procedure could be carried out for livestock farming systems.

The AMDM for alternative agricultural activities consists of three cooperative components:

(i) The Production Enterprise Generator component, that given a set of production orientations and crops generates all feasible crop rotations,

(ii) The Production Technique Generator component, that is responsible for generating the crop management options of the rotations, and

(iii) The Technical Coefficient Generator component, that quantifies, collects and formats the input data for the farm model.

An abstract architecture of the AMDM is presented in Fig. 1.

The aim is to develop AMDM components with open interfaces that adhere to a shared ontology. In this respect, all three components were developed on top of a Knowledge Manager shell (*KM*), capable to: (a) register domain ontologies, (b)load data from external sources, (c),realize links between components, and (d) provide interfaces with external applications (in our case APES and FSSIM).

The use of a Knowledge Manager maximizes the substitutability of AMDM components, by expressing all the knowledge related to component interfaces in a declarative way, using an ontology (as discussed for a similar application in Athanasiadis et al 2006). Also, based on the AMDM ontology we developed: (a) a database registration facility, that enables different data sources to be used directly as system inputs, and (b) the generation of data types, reflecting the ontology structure, based on which model algorithms can be directly programmed.

In Section 3 is shown how the use of ontologies in conjunction with the KM was realized for enabling all AMDM components to access a shared knowledge base for accessing existing databases or sharing data generated the system.

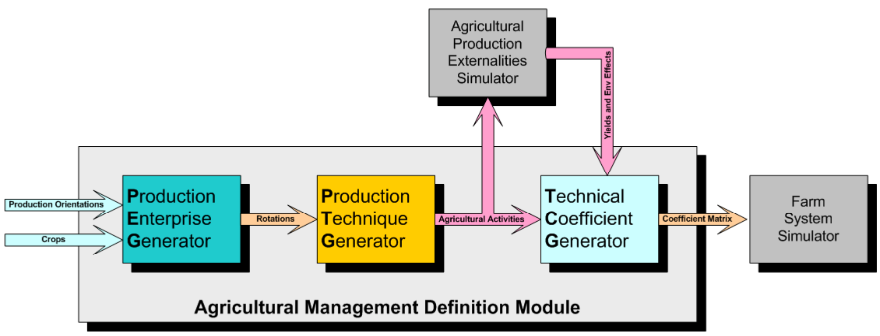



**Fig. 1.** The abstract architecture of the AMDM. Its three components along with the external applications are shown.

## 2.3 Detailed architecture

The features of each of the AMDM components are detailed below.

The **Production Enterprise Generator** (PEG) is the component that generates a set of feasible rotations of the farm based on suitability filters and rotational filters. In principle, all crops that may be grown in a given biophysical environment can be combined into rotations. However, not all of these combinations are agronomically feasible or desirable. The first part of the PEG contains 10 suitability filters that determine which crops can be grown in a certain biophysical environment, given the biophysical environment the farm is in and given a list of crops. The second part is based on a component called ROTAT, developed by Dogliotti et al. (2003). From the list of suitable crops, it generates all possible crop sequences and in a subsequent step eliminates all rotations that are not feasible according to 9 rotational filters. Suitability and rotational filters can be switched on or off as desired by the user as suitability filter factory and a rotational filter factory is used to create suitability filter and rotational filter objects at run time. The production orientation limits the length of the rotation and the amount of different crops in a rotation.

The **Production Technique Generator** (PTG) is a component to generate alternative agricultural activities on the basis of the feasible set of rotations by attaching crop management information to each crop in the rotation. A crop management is a complete set of agronomic inputs characterized by type, level, timing and application technique (Van Ittersum and Rabbinge, 1997). Crop management exists of five management practices: water management, general management (sowing, harvesting and field inspection), nutrient management, conservation management and weed, pest, disease management. For each of these management practices the PTG has one management generator, which generates a set of events for an aspect of crop management. An event is one operation that takes place during the growing season of the crop, for example sowing, fertilization, irrigation, harvesting, field inspection, etc. The five management generators generate the events based on the implements and inputs a farm uses, the specification of the management practices as part of the production orientation and the rotations that were an output of the PEG. Each of the management generators can be switched on or off independently as a management factory is used to create management generator objects at run time. The output of the PTG is fed into the dynamic crop simulation model APES. Note that APES simulates yields and environmental effects for each crop with associated management in a rotation.

Finally, the **Technical Coefficient Generator** (TCG) links the alternative agricultural activities generated by PEG, PTG and APES to socio-economic inputs and outputs by simple calculations and prepares the inputs for the farm model in an input-output matrix (see Fig.1). The TCG can produce an input-output matrix for the farm model on different scales: on a daily, yearly or seasonal basis, and on a rotational or individual crop in the rotation basis, as is dependent on the request from the farm model. The simple calculations carried out by the TCG are on variable costs and labour requirements and it has six different variable cost calculators.

### 2.4 An Ontology as a mediator for the Agricultural Management Definition Module

In order to systematically formalize our knowledge on Farming Systems Research, we developed an ontology for specifying the interfaces of AMDM to external applications (Databases, APES, FSSIM), along with the inter-component communication. In this way, AMDM components behave similarly to software agents providing information services, that are explicitly defined using ontologies. The AMDM ontology links to the core SEAMLESS ontologies (discussed in brief in Rizzoli et al. 2005, and will be available at http://seamless.idsia.ch/ontologies).

A small example of the developed ontology is presented in the following (Fig. 3(a)), where the concept of a Production Orientation is presented in the form of a conceptual map. A Production Orientation (PO; Section 2.3) limits the length of the rotation and the amount of different crops in a rotation and directs the management practices associated with the different crops. So, a PO is defined by three data type properties (the minimum and maximum rotation lengths, and the maximum number of crops), and an object property (*hasManagementPractices*) that associates a PO to a set of Management Practices. In Fig. 3(a), we present an example instance of a PO, called “Conventional”. The Management Practices of a PO on their turn are defined similarly as presented in Fig. 3(b), where the characteristics of and the relations between the PO and Management Practices are presented.

Following an iterative development procedure, all intrinsic concepts that AMDM deals with and their properties have been defined in the system ontology. This process involved several iterative reviews of the ontology among the domain experts and the knowledge engineers. The result of this activity was a declarative formalization of concepts that AMDM deals with. In this respect, we do not restrain the components developed by their current implementation languages or internal structures. In the contrary future extension or substitution could be easily and soundly supported if new components are developed with respect to the same (or an equivalent) ontology.

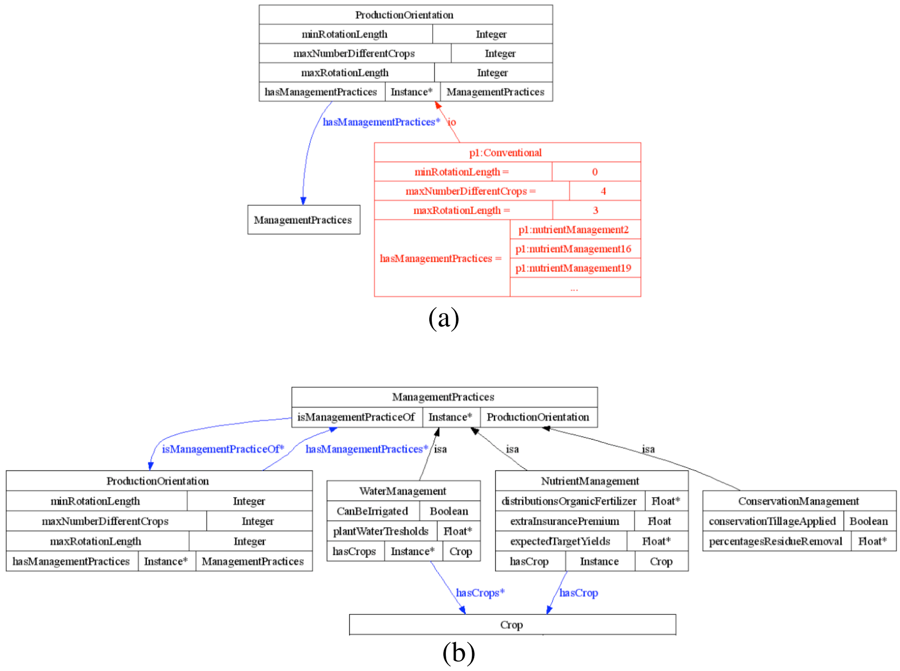



**Fig. 2.** Snapshots of the developed ontology: (a) The Production Orientation class and an example instance “Conventional”, (b) The Management Practices class and their characteristics.

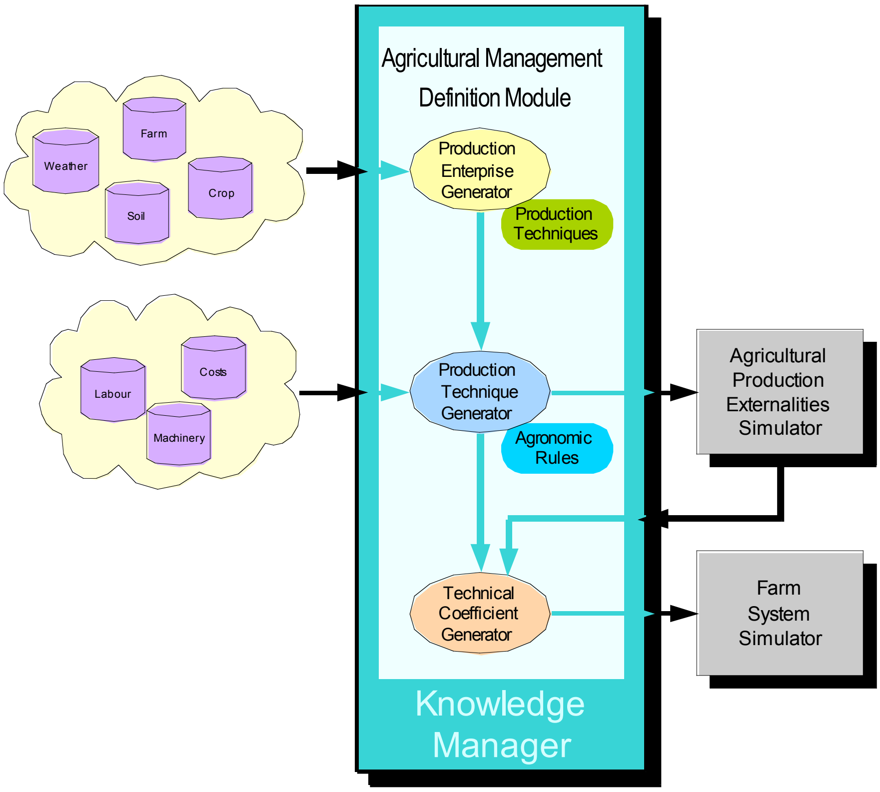
## 3. System realization and implementation

### 3.1 AMDM Implementation

The three components Production Enterprise Generator, Production Technique Generator and the Technical Coefficient Generator of the AMDM for alternative activities are developed in JAVA, while the Knowledge Manager was implemented as an extension of the Java Protégé-OWL API (Horridge et al., 2004).

The AMDM implementation is illustrated in Fig.3. AMDM was implemented on a KM shell that provides interfaces to external sources or applications and facilitates the linking of AMDM components. The added value of such an approach is three-fold:

a. completely separates algorithms from data and user interfaces,
b. facilitates easy linkage to external database sources and user interfaces, and




c. makes algorithms easily extensible and comprehendible.

**Legend**

Algorithm — Interface communication

Applications — Communication via the KM

Data sources

**Fig. 3.** AMDM for alternative activities and its components: algorithms, databases and connections.

These objectives were achieved by (i) linking the algorithms to databases through the ontology, (ii) developing the user interface at the very end, (iii) using design patterns, especially factory and strategy pattern where possible in the algorithms.

Following the workflow of Fig.3, the AMDM process starts with the invocation of PEG. All required inputs related to crops, farm, soils, etc are

fetched from external sources via the KM layer. In this respect PEG is not tightened to a specific database(s) structure, rather it relies on the KM to acquire all required information. Then the PEG is executed, creating the rotations that apply to the suitability and rotational filters that the user selected. These are transferred to PTG using the KM as a common repository for sharing results. Then, PTG calculates all possible Agricultural Activities, based on the appropriate management practices for each crop selected by the user, and on data related to machinery, labour and costs. Once again the data sources are decoupled from the algorithm implementation. Next,
the Agricultural Activities resulting from the PTG are communicated to APES (an external application), which simulates each one of those, calculating the yields and environmental effects. APES results are captured by the KM interface and fed to the TCG. Finally, TCG is executed for generating the Coefficient Matrix for the FSSIM optimizer. In this respect, the AMDM facilitates the linking of APES with FSSIM, in an open, "loose" coupling based on ontology-specified interfaces.

In the following paragraphs we present how the ontology was utilized for data type code generation, to facilitate on semantically aware development, and how the KM was utilized for connecting to external data sources.

### 3.2 Use of the ontology for code generation and semantic-rich development

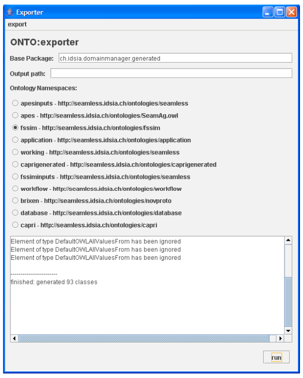
The ontology structure of the AMDM (Section 2.3) was used directly for developing the interfaces of the AMDM by generating the source code of the data types exchanged among components and applications. Although Protégé already includes a plug-in for code generation, a new code generator was developed. This was for two reasons. Firstly, because the Protégé code generator is outdated, and takes no advantage of powerful implementation practices, as code annotations and generics. Secondly, Protégé used a class implementation for each interface. In contrast the generator we built uses only interfaces and a common proxy class for accessing the knowledge base. This ensures that the developer will be accessing the Protégé Knowledge Base only via the interfaces, and will either have direct access or duplicate information. In this way, model exchanged information flows via a Knowledge Base, which allows performing semantic checks at runtime for ensuring the soundness of the link among components and applications.

The GUI of the ONTO:Exporter application for generating code interfaces is shown in Fig. 4 along with a code segment from the generated interface of the Production Orientation ontology class example, discussed in Section 2.4.

### 3.3 System execution and linking to databases with D2R

The AMDM system developed has been tested on existing data coming from the region of Flevoland, The Netherlands. As mentioned above database sources and schemas are entirely decoupled from the developed system. The latter operates on the Knowledge Manager generated datatypes, which are generic and could "hide" behind various implementations. For software testing purposes and experimentation, we employed the D2R language and library (Bizer 2003) for defining a mapping between any SQL Database (accessed via ODBC) and the AMDM ontology. This functionality is integrated in the Knowledge Manager for accessing external sources that are transformed to instances of the domain ontology.

```
package ch.idsia.domainmanager.generated.fs;

import java.util.Collection;
...

@ClassURI("http://seamless.idsia.ch/ontologies/fssim#ProductionOrientation")
public interface ProductionOrientation extends PEGDatatype {

    // Datatype Functional Property fs:minRotationLength
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#minRotationLength")
    public Integer getMinRotationLength();
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#minRotationLength")
    public void setMinRotationLength(Integer var);

// (Properties maxNumberDifferentCrops and maxRotationLength omitted
for simplicity)

    // Object type non-Functional Property fs:hasManagementPractices
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#hasManagementPracti
ces")
    public Collection< ManagementPractices> getHasManagementPractices();
    // Object type non-Functional Propertyfs:hasManagementPractices
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#hasManagementPracti
ces")
    public void setHasManagementPractices(Collection<ManagementPractices>
val);
    // Object type Propertyfs:hasManagementPractices
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#hasManagementPracti
ces")
    public void addHasManagementPractices(ManagementPractices val);
    // Object type Propertyfs:hasManagementPractices
    @PropertyURI("http://seamless.idsia.ch/ontologies/fssim#hasManagementPracti
ces")
    public void removeHasManagementPractices(ManagementPractices val);
}
```




**(a)** **(b)**

**Fig. 4.** (a) Code segment of the generated java interface ‘ProductionOrientation’ that is exported from the ontology. (b) The Onto:Exporter GUI through which the user may access more than one ontologies.

Finally, we demonstrated the use of the system with data from Flevoland. Specifically, PEG was run for 10 crops, a conventional production orientation with a maximum rotation length of 3 years and a farm type in Flevoland. The crops were Carrot, Onion, Pea, Springbarley, Springwheat, Tulip, Lucerne, Fibrehemp, Grassseed and Sunflower. All Suitability Filters and Rotational Filters in the PEG were used, and this ultimately led to 2 rotations as a result of the PEG execution. These rotations are ‘`GrassSeed-SpringBarley`’ and ‘`FibreHemp-GrassSeed-SpringBarley`.’ Next, the two rotations were fed into the PTG, which used data obtained from the database and specification of management practices on the production orientation, and ultimately generated 252 alternative agricultural activities. Every agricultural activity contains a set of crops (rotation), each of which is associated with a certain year in the rotation and with a set of management events, which is all the information required by APES. The grass seed crop in agriculturalActivity1 with ‘`GrassSeed-SpringBarley`’ has 5 different events, e.g. 1 sowing event, 1 harvest event, 3 nutrient events, and no irrigation events. The execution of APES associated yields and environmental effects to each agricultural activity, that ultimately was directed to the TCG. TCG execution ultimately resulted 720 production coefficients that were finally forwarded to the FSSIM to select the optimal set of production activities given the farmer objectives. For the creation of production coefficients again the data was retrieved from the database via the KM, as in the TCG information on variable costs and labour requirements were attached to each production coefficient. For example, a grass seed crop in productionCoefficient1 with rotation ‘`GrassSeed-SpringBarley`’ has associated a labour requirement of 20 hours per hectare per year and variable costs of 450 euros per hectare per year.

## 4. Discussion

### 4.1 Benefits of the approach

Through the case of the AMDM it was explained how semantic modelling, ontologies and the knowledge management practices can be used in modelling agricultural systems. The ontology structure of AMDM helped to capture the complexity of the AMDM by making explicit the knowledge that the agricultural scientist holds. By the use of ontologies, the agricultural scientist is *forced* to define concepts he/she commonly refers to by specifying their properties (datatype properties) and relationships to other concepts (object properties) in a detailed formalization. Also, the concepts in the ontology could subsequently be made available for modelling by allowing the agricultural scientist to write the algorithms. As farming systems approaches require interdisciplinary studies (Kropff et al., 2001) and therefore require the use of several different models and techniques, the different techniques and models need to be able to communicate with meaningful objects. In the AMDM the developed ontology played a vital role in clarifying exchanged information between AMDM and its external peers (i.e. APES, FSSIM, and the databases), as well as among the components of AMDM (PEG, PTG and TCG). This allowed the AMDM to operate on two different scales, both in space in time. Firstly, AMDM is able to exchange meaningful objects with a point scale model, operating on a daily basis, as APES is a dynamic crop simulation model. Secondly, it is able to exchange meaningful with a farm scale model, using annual data, as FSSIM is a usually a static, mathematical programming farm model.

Using ontologies in farming systems research requires a close cooperation between disciplines, in this case agronomy, agricultural economy and information technology. A close cooperation can be achieved by frequent iterations and discussion on the concepts used in both disciplines. Using ontologies implies an additional layer to the modelling exercise, which enforces the other layers (databases, algorithms and model structure), while at the same time making the modelling exercise more distributed.

In particular, in this paper contributed with a modular architecture and implementation of the Agricultural Management Definition Module, by exploiting ontologies and semantic modelling. Also, we examined the performance of existing knowledge engineering tools, particularly related to linking ontologies with legacy database sources and generating programming interfaces. Although effective the use of D2R language and library for Object-Relational Mapping can be significantly improved, if the process becomes semi-automated, i.e. through a formal data registration process. Therefore our future developments will drive towards this direction.

Future work will concentrate on expanding the current implementation. We plan to exploit further the developed ontology by expressing both production techniques filters and agronomic rules in a declarative fashion. Due to the nature of the rules, it is very hard, if not impossible to express them using description logics in OWL-DL. Thus extended frameworks like RuleML and SWRL will be considered for incorporating reasoning capabilities. This will further advance the benefits of using ontologies and semantic modelling in the agricultural modelling domain. Also, alternative implementations of the same system design using web services or software agents will be investigated for deploying the system across a distributed network. The same ontology could be used for describing model interfaces, which in such a case will be implemented as web services or agent communicative acts. Finally, we consider within the context of SEAMLESS project to promote further the use of semantic modelling, and the development of ontologies for the agricultural sector in order to maximize the reusability and the extensibility of the systems developed. Parallel efforts are adopting declarative approaches for the APES and FSSIM applications. The adoption of a set of shared ontologies within these modularised applications will lead us to a semantic-aware modelling and simulation framework for the agriculture sector.

**Acknowledgement:** This publication has been partially funded under the SEAMLESS integrated project, EU 6th Framework Programme for Research, Technological Development and Demonstration, (contract no. 010036-2). For further reference, the SEAMLESS project website is: http://www.seamless-ip.org.

## References

1. Athanasiadis, I.N., Rizzoli, A.E., Donatelli, M. & Carlini, L., Enriching software model interfaces using ontology-based tools, *in* '3rd Biennial meeting of the International Environmental Modelling and Software Society', iEMSs, 2006.
2. Bizer, C. 2003. D2R MAP - A Database to RDF Mapping Language. In: Proceedings of the Twelfth International World Wide Web Conference (WWW2003), Budapest, Hungary.
3. Dogliotti, S.; Rossing, W.A.H. & van Ittersum, M.K.,(2003) 'ROTAT, a tool for systematically generating crop rotations. European Journal of Agronomy 19, 239-250
4. Ewert, F.; van Ittersum, M.; Bezlepkina, I.; Oude Lansink, A.; Brouwer, F.; Andersen, E.; Flichman, G.; Heckelei, T.; Olsson, J.; Olsson, L.; Rizzoli, A.; van der Wal, T. & Wery, J. (2005),'Development of a conceptual framework for integrated analysis and assessment of agricultural systems in SEAMLESS-IF', Technical report, SEAMLESS-IP, http://www.seamless-ip.org/Reports/Report_01_PD1.2.1.pdf.
5. Gruber, T.R., 'A Translation Approach to Portable Ontology Specifications', Knowledge Acquisition 5(2), 199-220, 1993.
6. Horridge, M.; Knublauch, H.; Rector, A.; Stevens, R. & Wroe, C. A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools, Online Tutorial (http://protege.stanford.edu/plugins/owl/), The University Of Manchester and Stanford University, 2004.
7. McGuinness, D. & van Harmelen, F. (2004),'OWL Web Ontology Language Overview', Technical report, WWW Consortium, www.w3.org/TR/owl-features/.
8. Kropff, M.J.; Bouma, J. & Jones, J.W. (2001), 'Systems approaches for the design of sustainable agro-ecosystems' Agricultural Systems 70 (2-3), 369-393
9. Rizzoli, A., Donatelli, M., Athanasiadis, I., Villa, F., Muetzelfeldt, R. & Huber, D., Semantic links in integrated modelling frameworks, *in* A Zerger & R.M Argent,

ed.,'16th International Congress on Modelling and Simulation (MODSIM 2005)', MSSANZ, 2005, pp. 704-710.

10. Ten Berge, H.F.M; van Ittersum, M.K.; Rossing, W.A.H.; van de Ven, G.W.J.& Schans, J.(2000) ‘Farming Options for the Netherlands explored by multi objective modelling’ European Journal of Agronomy 13, 263-277
11. Van Ittersum, M.K. and Donatelli, M. (eds.) (2003), ‘Modelling Cropping Systems: Science, Software and Applications’. Special Issue of European Journal of Agronomy 18 (3-4)
12. Van Ittersum, M.K. and Rabbinge (1997) ‘Concepts in production ecology for analysis and quantification of agricultural input-output combinations.’ Field Crops Research 52, 197-208
13. Warren, P. (2006), 'Knowledge Management and the Semantic Web: From Scenario to Technology', IEEE Intelligent Systems 21(1), 53-59.
14. Willmott, S.; Dale, J. & Charlton, P. (2002),'Agent Communication Semantics for Open Environments', Technical report f-in-00066, FIPA.