Multi-agent Systems and Ontologies Applied to New Industrial Domains
Case Study: Ornamental Plants

Lorena Otero-Cerdeira, Francisco J. Rodríguez-Martínez,
Tito Valencia-Requejo and Loxo Lueiro-Astray
Laboratorio de Informática Aplicada 2, University of Vigo, Campus As Lagoas, Ourense, Spain

Keywords: Multi-agent System, Heterogeneity, Information Integration, Ontology.

Abstract: This paper describes a real solution applied to an enterprise of ornamental plant selling and distribution. The platform that we propose uses intelligent agent technologies and ontologies to meet the special needs of an enterprise of this kind. We present the architecture defined with the agents involved in both parties, the plant wholesaler and the plant producers. A description of the ontologies that these agents use to interact is also provided. In the final section some relevant issues detected and conclusions will be presented.

1 INTRODUCTION

The market of ornamental plants has always been very important in northwestern Spain and yet it is one of the sectors where fewer technological change has been incorporated in recent years. This paper describes a real solution applied to an enterprise of ornamental plant selling and distribution. This enterprise, a plant wholesaler, needs to interact daily with the producers of the plants they sell and with the customers that buy them. Our platform uses intelligent agent technologies and ontologies to meet the special needs of an enterprise of this kind.

In business context it is essential to track all information and business processes, even more, since this enterprise is working with a perishable product as plants are, making an efficient and just-in-time control is crucial.

The platform that has been designed allows the enterprise to carry out tasks such as, *Production planning*, *Order management logistics* and *Catalog processing*.

When taking into account all the requirements and issues of the enterprise, no software product or platform was found that integrated all of them in a fulfilling way. So there was the need to develop a completely new platform from scratch to satisfy all the requirements of such a system without the help or expertise that the existence of other software products or platforms would provide. Here lies the strength and the importance, but also the main difficulty of this project.

The development of such a platform was addressed as a multi-agent system (MAS), which we chose to divide as well in different smaller MAS for the sake of the design and categorization, obtaining this way a layered representation of the MAS.

MAS have usually been used in simulation applications, real-world interactions or adaptive structured information systems (Valckenaers et al., 2006). Within these contexts MAS have been used for fire accident detection(Gowri et al., 2010), emergency evacuation simulation (Sharma, 2009)(Murakami et al., 2002), meetings planning (Macho et al., 2000), etc., but never to develop a platform as the described in this paper.

It is important to state that not the whole system was addressed as a MAS, other pieces of software were developed to cover aspects of the system where the features of the MAS were not needed, such as human resources management or accounting. For the purposes of this paper only the part of the system regarding the MAS will be described

The platform is divided into three different MAS. The *Plant Wholesaler MAS* is composed by the agents that take care of the enterprise's tasks mentioned above. The *Plant Nursery MAS* is composed by the agents that take care of all the processes related to plant producing, such as catalog processing or production following up. Both of these MAS are tightly related to the legacy systems existing in both types of organizations. The *Communication MAS* holds the agents that undertake the actions related to mutual exclusion preservation and information integration.

Otero Cerdeira L., J. Rodríguez Martínez F., Valencia Requejo T. and Lueiro Astray L..
Multi-agent Systems and Ontologies Applied to New Industrial Domains - Case Study: Ornamental Plants.
DOI: 10.5220/0004136303570364
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2012), pages 357-364
ISBN: 978-989-8565-30-3

The rest of this paper is structured as follows, in Section 2 a detailed view of the system's goals will be provided. In Section 3 the builded platform will be fully analyzed. In Section 4 we will describe some issues that were discovered during the development of the platform and finally in Section 5 we will summarize the main conclusions of the development as well as further improvements that will be undertaken.

2 SYSTEM'S GOALS

The project's main goal was to obtain a platform that could improve the production management, cost control in the producers and marketing support in foreign countries by using MAS and Business Intelligence algorithms.

Nowadays there are some commercial applications that centralize all the information related to plant nursing and plant selling processes allowing queries over this information. These applications are simply aggregative or storage systems and do not provide support to the enterprise's manager in relevant tasks such as production management, real-time order management, stock following up and production planning. Some of these solutions do have modules that take care of tasks such as field control, commercial invoices management and, customers and providers tracing.

From plant producers point of view, such applications do not achieve the requirements established, since they don't take into account the special characteristics of plants as products. Plants are a perishable item and so its production planning is not as straight and pre-established as other product's planning would be. Plants grow, die, bloom, etc., these kind of changes and others plants experience turn products into new ones. The same plant in the same flowerpot can be different products as it changes. Besides in these applications costs are stablished following financial and accounting parameters, which works fine in other types of productive areas but not on this one.

From the plant wholesaler point of view, there are several systems that do marketing management but that are simply Enterprise Resource Planning systems (ERP) which have been adapted from other productive areas and that do not consider the special circumstances of such an heterogeneous market as plant selling is.

The following needs have been identified as project goals as they don't coexist in a single software application in this environment: *Dynamic catalog processing*, *Dynamic price estimation*, *Dynamic route planning*, *Order processing*, *Production cost management*, *Production planning*, *Reservation of future products*, *Sales forecasting* and *Stock following up*.

Besides the functional requirements identified above, there were also other non-functional requirements that were tagged as mandatory for the final platform. Among them, the most remarkable are: *Efficiency*, *Expandability*, *Flexibility*, *Maintainability*, *Performance*, *Reliability*, *Robustness*, *Scalability*, *Security* and *Usability*.

After analyzing all the requirements listed above, both functional and non-functional, the most suitable approach seemed to develop a *multi-agent system*.

At this point it is important to state that each plant nursery has its own information structure and confers different semantic meaning to the concepts of the plant nursing field. Information integration was one of the major problems that the legacy system had. People that first developed the legacy system were not experts in the field and so they used concepts without fully understanding its meaning. Without a common ontology over the concepts, the use and maintenance of the legacy system has turned incomprehensible. It is important to remark that the MAS itself will not cover for the information integration and so, means to overcome this information heterogeneity needed to be studied.

In order to achieve all the requirements listed above a MAS by itself could not replace the previously existing legacy system, it was also necessary a deep reengineering over the information system.

After considering several alternatives to rebuild the information system, the chosen one involves rebuilding the database structure for both types of entities. To perform the communication and data exchange between them the solution was found in ontologies.

By using this approach, agents defined for the MAS would use an ontology to describe the meaning of concepts in the communication processes (Wiesman et al., 2002), avoiding this way the heterogeneity that exists between the involved knowledge sources.

3 SOLUTION'S DESCRIPTION

The platform that we propose consists in the definition of a layered MAS whose agents use ontologies to provide interoperability between them. As mentioned before, the global MAS is divided into three MAS that will now be described.

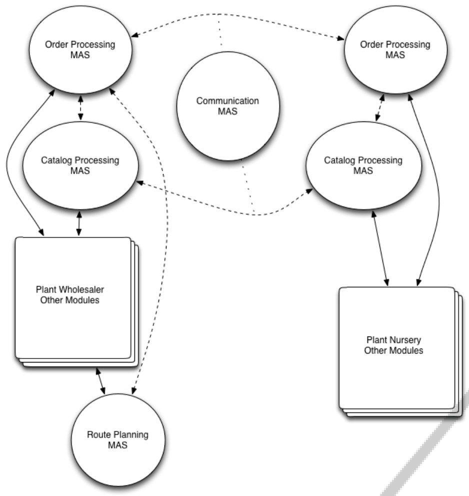

Figure 1: MAS and their relations.

3.1 MAS Description

The plant wholesaler MAS consists of a group of agents performing tasks for which a traditional solution would not provide the appropriate functionality. Accordingly, intelligent behavior will accomplish tasks such as order processing, catalog processing or route planning.

In figure 1 the whole structure of the Plant Wholesaler MAS, the Plant Nurseries MAS and the interactions among all these MAS are depicted.

3.1.1 Order Processing

Order processing involves several groups of agents. Those from the plant wholesaler compose its Order Processing MAS. The same happens in the plant nurseries, every agent involved in order processing integrates the Order Processing MAS. The agents defined for the plant wholesaler are of two different types, *Order Agent* and *Negotiator*.

Order Agent. *An order agent is responsible for splitting an order between its negotiators and composing the shipping with the results of each negotiator.* When a new order arrives to the plant wholesaler, a new order agent is created to manage it. This agent reports its existence to a mediator agent, which acts as an intermediary between the plant wholesaler's agents and the plant nurseries's agents. For every item in the order list, the order agent activates a negotiator agent to manage it. Once every negotiator is done, the order agent composes the final order with the information that each one of its negotiators provides.

Negotiator. *A negotiator agent is responsible for dealing with a provider to get a product.* A negotiator agent is responsible for getting the best offer for a product. The best offer might not only be determined by the price of the product. Indeed, it is necessary allowing that other constraints could be applied, such as provider's proximity or even constraints in the product.

For order processing, in the plant nurseries side two types of agents were defined, *Provider* and *Broker*.

Provider. *A provider agent supplies products from the plant nursery.* When a new provider is created it communicates its existence to a mediator agent. Since all mediator agents are federated, the knowledge about providers and negotiators is shared among all of them. A provider agent has a group of broker agents that negotiate the different products. A provider has at most a broker agent for each product that is sold in the plant nursery. The provider regularly reports its state to a mediator, so the mediators can optimize the distribution of requests from the negotiators among the different brokers. The provider reports on the products that their brokers are currently selling and on those that they have in stock for selling.

Broker or Delegate. *A broker agent is responsible for dealing with plant wholesaler's negotiators.* A broker agent sells to negotiator agents its product, respecting the boundaries that the provider agent may have established. If the broker belongs to an internal provider, it deals the conditions of the sale with the negotiator agent specified by a mediator. In this case we refer to it as *delegate*. If the broker agent belongs to an external provider, the selling process works as an auction between the negotiator agent and all the broker agents that sell the product.

The last type of agent involved in order processing is the *Mediator*. This type of agent does not belong to the plant wholesaler's MAS or to the plant nurseries's MAS, it belongs to the Communication MAS.

Mediator. *A mediator agent is responsible for linking negotiators to brokers.* In the communication MAS there are several federated mediators, which means that the knowledge that one of them has is immediately shared with the other mediators. A mediator agent is responsible for discovering the providers and for linking the negotiators to the brokers.

In figure 2 the whole structure of the order processing procedure is reflected. In figure 3 a detailed view of all the communications established among the agents is specified. In this figure the communication between *Negotiator - K* and *Delegate - Q* is performed inside the Communication MAS, as reflects the line labeled as *4*. The ontologies necessary for communication to be successful will be described in section 3.2.

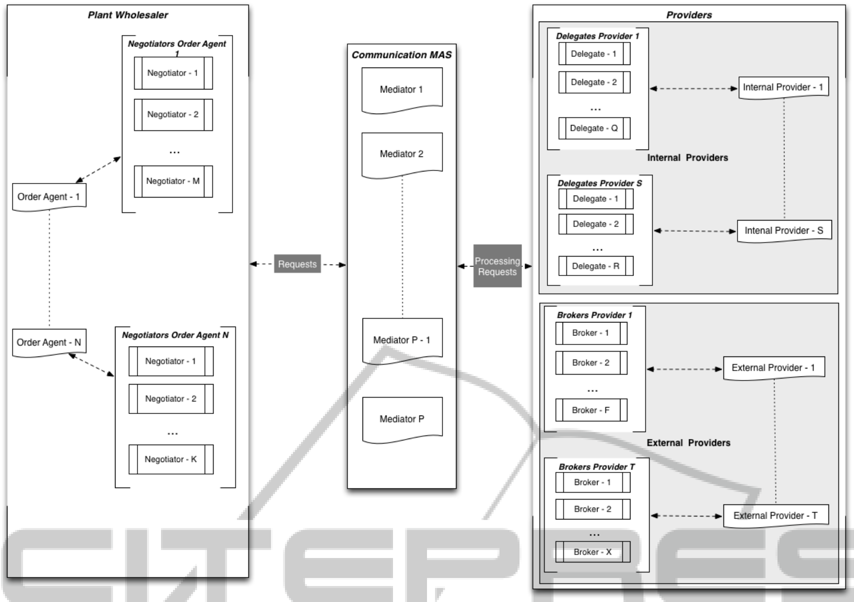

Figure 2: Order Processing.

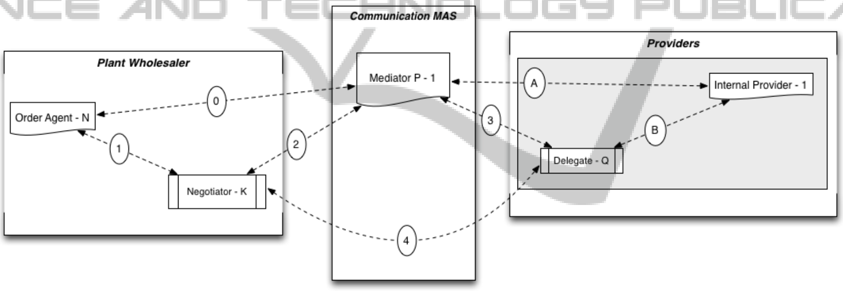

Figure 3: Communication during Order Processing.

3.1.2 Catalog Processing

Catalog processing involves agents in the plant wholesaler and in the plant nurseries, additionally agents in the Communication MAS are necessary for mediation purposes.

The types of agents involved in catalog processing are *Catalog Combiner* and *Seeker* for the plant wholesaler, and *Catalog Manager* for each plant nursery.

Catalog Combiner. *A catalog combiner is responsible for obtaining an updated catalog for the plant wholesaler.* The product catalog of the plant wholesaler must be daily revised and modified to provide the customers with an updated version. The catalog combiner agent builds the catalog from the existing one, taking into account both internal and external information. As internal information, the sales of the day and the reservations must be considered, and as external information, the updates on every plant nurseries's catalog that is a provider for the plant wholesaler must be introduced.

Seeker. *A seeker agent is responsible for gathering the updated information about a provider's catalog.* A seeker agent is responsible for getting the updated version of a provider's catalog. Once a seeker is created it already knows the provider that it must deal with. By default the seeker gets the plant nursery's whole catalog, but it can also be configured to query the provider's *Catalog Manager* for an specific product o for a product that fulfills a set of requirements.

Catalog Manager. *A catalog manager agent is responsible for updating a plant nursery's catalog.* The catalog manager is responsible for updating a plant nursery's catalog by taking the existing one and modifying it with the sales of the day and the modifications that the products may have experienced. The catalog manager reports its existence to a mediator agent, so when a catalog combiner agent queries the mediator for the list of plant nurseries, it is included. From that moment on, the catalog manager is available for taking the requests from the seekers.

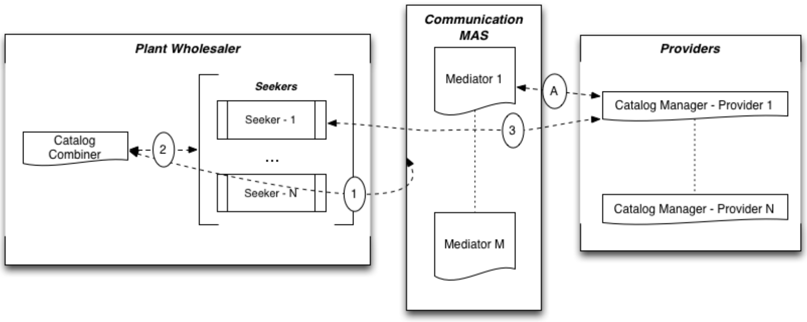

Figure 4: Catalog Processing. Communication and Structure.

As line labeled as 3 in figure 4 reflects, the communication between a seeker and a catalog manager takes place inside the *communication MAS*. This communication is be successful since both agents involved are using same ontology. This will be detailed in section 3.2.

Mediator. *A mediator agent is responsible for providing indexing information.* A mediator agent holds the information over which catalog managers are available to be queried from the seekers. Since in the communication MAS several mediators exist, they must be federated so there is no difference in the information that they hold.

3.1.3 Route Planning

Another important task of the system is the dynamic route planning. It was also addressed as a *Multi-Agent Resource Allocation (MARA)* problem. In this type of problems a group of agents share a common resource which requires a coordination mechanism that will manage its usage (Cicortas and Iordan, 2011). Dynamic route planning can be addressed as an independent task that lies outside the scope of this paper, and that will be developed in further papers.

3.2 Ontologies Defined

The existing information structure that the different entities had, was very different, causing the interaction between them to be very hard. In addition we have identified several cases where the same term was used with different meanings, not only between the different entities but also within the same entity. The MAS defined in this project would not succeed if the problem of the semantic gap were not addressed. To accomplish this task, two different ontologies were defined, one for the plant wholesaler and the other one for the plant nurseries, this two ontologies share the semantic meaning over the concepts and so, agents are able to interact. In multi-agent environments ontologies are expected to complement mutual understanding and interactive behavior between such agents (Laera et al., 2007).

The ontologies used in the MAS were defined with the *Ontology Web Language (OWL2)* (Grau et al., 2008) which is the standard recommended by the W3C. This language is used to formalize the domain, defining the concepts as classes, and the properties that these classes have (Roussey et al., 2011). Ontologies were identified as the most accurate technique to share common knowledge among a group of software agents.

3.2.1 Plant Wholesaler Ontology

In figure 5 the global structure of the Plant Wholesaler Ontology is depicted.This ontology reflects the specific situation of the enterprise that hosted the project and it may not be applied to every plant wholesaler due to the peculiarities of the enterprise.

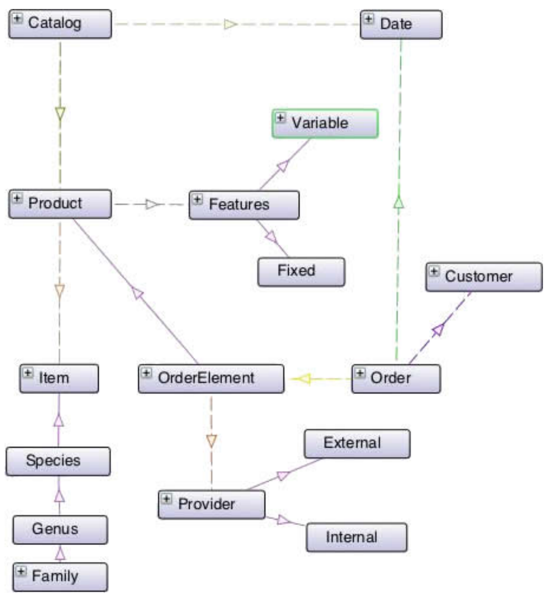

Figure 5: Plant Wholesaler Ontology.

The plant wholesaler ontology is composed by the following classes and properties.

Class: *wholesaler:Catalog.* It represents a collection of products that are available. The catalog is composed by integrating the list of available products that each provider of the plant wholesaler has. The property *wholesaler:generatedIn* links a catalog to its creation date.

Class: *wholesaler:Product.* A product is the smaller entity that composes a catalog. Products are usually plants, but they could also be goods related to the business, such as fertilizers or flowerpots. Products are identified by a plant variety and the defining features of that plant. The property *wholesaler: integratedBy* defines the products that are part of a catalog.

Class: *wholesaler:Item.* It identifies the actual plant variety of a Product. In the botanical hierarchy the item should be equivalent to the plant species **(Class: *wholesaler:Species*)**, however in most cases this does not happen, and the equivalence is set at the genus level **(Class: *wholesaler:Genus*)** or even at the family level **(Class: *wholesaler:Family*)**. Fixed attributes would be: *Flowerpot, Size and Price*, which means that for every product the values of this attributes must always be set. The property *wholesaler: basedIn* indicates that the item is the base of a product.

Class: *wholesaler:Features.* It represents a collection of attributes that distinguish the different products. There are some attributes that every product must enclose which are included in the **Class: *wholesaler:Fixed***, and there are other features that depend on the family, genus and species of the product, which are encompassed in **Class: *wholesaler:Variable***. This subclasses are defined as disjoint classes. The property *wholesaler:definedBy* indicates that a product is modified by a collection of features.

Class: *wholesaler:Order.* It represents the collection of elements that a customer requests from the plant wholesaler. The property *wholesaler:orderedIn* links an order to its creation date.

Class: *wholesaler:Customer.* It encloses the knowledge about the plant wholesaler's clients. The property *wholesaler:orderedBy* identifies the customer has requested an order.

Class: *wholesaler:OrderElement.* It represents each one of the elements in an order. Each OrderElement is only linked to a product in the catalog and to a provider, that would be the nursery that provides the product. The property *wholesaler:composedOf* denotes that an order is integrated by a collection of order elements.

Class: *wholesaler:Provider.* It identifies the nurseries that interact with the plant wholesaler. The providers must be split into two different and disjoint subclasses, internal providers **(Class: *wholesaler:Internal*)** and external providers **(Class: *wholesaler:External*)**. The property *wholesaler:providedBy* indicates the provider that supplies a certain product.

3.2.2 Plant Nursery Ontology

In figure 6 the global structure of a nursery's ontology is depicted. This is the ontology that every agent in a nursery's MAS will use. This ontology is composed by the following classes and properties.

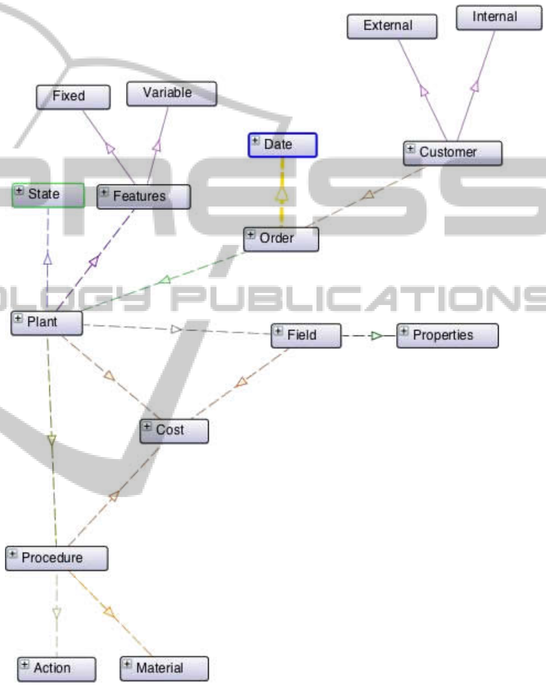

Figure 6: Plant Nursery Ontology.

Class: *nursery:Customer.* It represents the clients that the plant nursery may have. There are some customers that are considered as internal, and therefore defined in **(Class: *nursery:Internal*)**. This class is disjoint to the subclass **(Class: *nursery:External*)** that represents the external clients of the nursery. To relate a customer with the orders, there is the property *nursery:makes*, defined to be asymmetric and irreflexive.

Class: *nursery:Order.* It represents the collection of elements that a customer requests from the plant nursery. The property *nursery:involves* indicates that an order groups a collection of plants.The property *nursery:orderedIn* relates an order to the date it was made.

Class: *nursery:Plant.* It represents the smaller entity that the plant nursery works with. The property *nursery:locateIn* indicates the location of a plant by relating them to one of the different available fields.

Class: *nursery:State.* It denotes the different situations a plant may have. The plant nurseries do not have a catalog itself but a group of plants that satisfy some quality requirements to be sold. This state has a high frequency change. The property *nursery:hasA* identifies that a plant has a certain state among all the possible ones.
Class: *nursery:Field.* It represents a piece of land where different types of plants may be cultivated. Each field is identified by a group of properties and it has a cost assigned. The features of the field have an impact over its cost, which therefore affects plant's cost.
Class: *nursery:Properties.* It represents a group of attributes that distinguish the different fields. The property that relates a field to its feature is *nursery:identifiedBy*.
Class: *nursery:Features.* It represents a collection of attributes that distinguish the different plants. There are some attributes that every plant must enclose which are included in the **Class: *nursery:Fixed***, disjoint to the features defined in **Class: *nursery:Variable***. The property *nursery:definedBy* indicates that a plant is modified by a collection of features, it is an irreflexive and asymmetric property.
Class: *nursery:Procedures.* It denotes a collection of actions or operations that can be made over the plants. The procedures are composed by atomic instructions defined in **Class: *nursery:Action.*** The property *nursery:composedBy* represents the link between a procedure and its composing actions. The property *nursery:suffers* indicates that procedure is applied to a plant or group of plants. Every procedure consumes resources such as, fertilizers, water, sawdust, that are enclosed in the class **Class: *nursery:Material.*** These concepts are related by the property *nursery:consumes*.
Class: *nursery:Action.* It defines the atomic actions that could be made on a plant or set of plants.
Class: *nursery:Cost.* It denotes the value that plants, procedures or fields have. Its domain is defined as the intersection of these classes. The property *nursery:worthA* relates a class in the domain to its cost. The cost of a plant is determined taking into account the cost of the procedures that it may have suffered and the cost of the field where the plant is being grown. It is important to remark that the cost of a plant is not its price. The price is determined applying a percentage of profit over the cost of production. This margin of benefit is crucial to the agents negotiation processes that take place to accomplish order processing.

4 IDENTIFIED ISSUES AND FUTURE WORK

The platform defined to ensure proper operation and networking between plant nurseries and plant wholesaler fully meets expectations as far as regards inner functioning.

Plant nurseries that adopted the platform described have experienced an improvement in response times, cost estimation and production following up. In turn in the plant wholesaler the effort to keep the catalog updated and to process the costumer's orders has been reduced. Problems showed up when some plant nurseries chose to use the platform but keeping their own information representation. Agents were no longer able to interact correctly since their ontologies where not compatible.

To overcome the situation the most suitable solution was to force an *ontology matching* process before agents started their communication. Imposing a single shared ontology would be, not only impractical because it would force the parties to use a standard communication vocabulary; but also limiting since it would not consider the requirements of agents that could be developed in the future. Ontology matching is a way of guaranteeing the interoperability of the parties involved in a communication process, i.e, to ascribe to each important piece of knowledge the correct interpretation (Euzenat, 2001). In our case, the purpose is to find semantic mappings between the concepts of the different ontologies that the agents use.

The problem of ontology matching has been extensively studied in recent years as stated in the works of (Noy, 2004), (Euzenat and Shvaiko, 2007), (Trojahn et al., 2008) or (Shvaiko and Euzenat, 2012). And also its applications in Multi-Agent Systems (Laera et al., 2007) (Wiesman et al., 2002).For generating the matches between ontologies several frameworks and techniques have been proposed, as those in the works of (Falconer and Noy, 2009) (Klein, 2001) (Shamsfard and Barforoush, 2003) (Köpcke and Rahm, 2009) (David et al., 2010).

The next step of the development of this platform will be the integration of ontology matching techniques, to ensure the communication with agents that use an ontology different to the proposed one. The process that we will use to accomplish this challenge is as follows. First, the state of the art of ontology matching (Kalfoglou and Schorlemmer, 2003) will be deeply studied to identify the new trends and research fields. Then a framework will be developed to test different ontology matching algorithms, the purpose is to determine which set or combination works the best for this problem. Finally, once the algorithms are determined, they will be integrated in the system.

5 CONCLUSIONS

In this paper we presented a Multi-Agent System applied to a new field, ornamental plant selling and distribution, where this technologies have never been used before. The MAS defined takes into account the particularities of this environment and it obtains the most profit of MAS features to deal with it. The system allows the different parties to operate not only coordinated but also independently. To allow this coordination two ontologies were defined in order to transmit the knowledge of this domain to the agents involved in the communication processes. This ontologies by themselves were not enough to cover every possible communicative scenario, since other systems, or agents could use ontologies different from the ones defined. A further step of including ontology matching has been identified as necessary and it is the core of the ongoing research.

ACKNOWLEDGEMENTS

This work has been supported by the project 10MRU007E supported by Xunta de Galicia.

REFERENCES

Cicortas, A. and Iordan, V. (2011). Multiagent systems for resource allocation.

David, J., Euzenat, J., Scharffe, F., and dos Santos, C. T. (2010). The Alignment API 4.0. *IOS Press*, 1:1 – 8.

Euzenat, J. (2001). Towards a principled approach to semantic interoperability.

Euzenat, J. and Shvaiko, P. (2007). *Ontology Matching*. Springer.

Falconer, S. M. and Noy, N. F. (2009). Interactive Techniques to Support Ontology Matching. *Data-Centric Systemsand Applications - Springer*.

Gowri, R., Kailas, A., Jeyaprakash, R., and Anirudh, C. (2010). Development of multi-agent system for fire accident detection using gaia methodology. *International Journal of Computer Science and Information Security (IJCSIS)*.

Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., and Sattler, U. (2008). Owl 2: The next step for owl.

Kalfoglou, Y. and Schorlemmer, M. (2003). Ontology mapping: The state of the art. *Science*, 2:3.

Klein, M. (2001). Combining and relating ontologies: An analysis of problems and solutions.

Köpcke, H. and Rahm, E. (2009). Frameworks for entity matching: A comparison. *Data and Knowledge Engineering - Elsevier Science*.

Laera, L., Blacoe, I., and Tamma, V. (2007). Argumentation over ontology correspondences in mas.

Macho, S., Torrens, M., and Faltings, B. (2000). A multi-agent recommender system for planning meetings. In *In Proc. of the Agents'2000 workshop on Agent-based recommender systems (WARS'2000)*.

Murakami, Y., Minami, K., Kawasoe, T., and Ishida, T. (2002). Multi-agent simulation for crisis management.

Noy, N. F. (2004). Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD*, 33(4).

Roussey, C., Pinet, F., Kang, M. A., and Corcho, O. (2011). An Introduction to Ontologies and Ontology Engineering. *Advanced Information9 and Knowledge Processing - Springer*.

Shamsfard, M. and Barforoush, A. A. (2003). The State of the Art in Ontology Learning: A Framework for Comparison.

Sharma, S. (2009). Avatarsim: A multi-agent system for emergency evacuation simulation. *Journal of Computational Methods in Science and Engineering*.

Shvaiko, P. and Euzenat, J. (2012). Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Software Engineering*.

Trojahn, C., Moraes, M., Quaresma, P., and Vieira, R. (2008). A cooperative approach for composite ontology mapping.

Valckenaers, P., Sauter, J., Sierra, C., and Rodriguez-Aguilar, J. A. (2006). Applications and environments for multi-agent systems. *Springer Science Business Media*.

Wiesman, F., Roos, N., and Vogt, P. (2002). Automatic ontology mapping for agent communication. *ACM*.