Design and Implementation of a Multi-agent Framework for the Selection of Partners in Dynamic VEs[1]

Pedro Sanz Angulo[1], Juan José de Benito Martín[1]

[1] University of Valladolid, Industrial Engineering School.
Paseo del Cauce, 59. 47011 Valladolid, Spain.
{psangulo, debenito}@eis.uva.es

Abstract. Many researchers agree that a key element in the VE succeed is the search, identification and selection of partners. This task is complex and is deeply influenced by the negotiation needs. This paper presents a simple but operational model of the selection process, in which partners are selected from a VE Breeding Environment. This model is the core of the multiagent platform that we have developed specifically for the selection of partners in the DVE context, which we also present in this paper.

Keywords: Dynamic Virtual Enterprise, VE Breeding Environments, Selection of Partners, Multiagent Technology, Expert System.

1 Introduction

In order to survive in the present context, companies must be able to permanently satisfy the desires of their customers, improving productive efficiency and adapting continuously to a global, competitive and dynamic environment [1]. However, added value creation for customers has become an increasingly complex process which requires the mix of different kinds of resources and expertise that companies do not necessarily have [2]. Companies are forced to cooperate, sometimes even with their direct competitors, which has led to the introduction of many organizational concepts and structures, highlighting among them the Dynamic Virtual Enterprise model.

There are many benefits associated to this business model: a faster access to new markets and new business opportunities; partners can overcome challenges, achieve business goals, access to resources (skills, know-how ,...), etc., which usually are outside the scope of a single firm; increases the utilization of assets, improves customer service and product quality, reduces risks, costs, etc., allows to achieve economies of scale, SMEs achieve international presence, etc.

However, to achive its large-scale implementation we need to overcome several obstacles, especially those arising from the selection of partners. This is undoubtedly the main reason why many researchers in different areas of knowledge have focused their efforts in this direction in last years. Therefore, it is necessary to offer innovative solutions that meet the demands of the DVE in an efficient way.

Based on these considerations, this paper presents an operational and innovative model of the selection process. This model is the core of the multiagent platform we have developed, and which we also present here, for this purpose.

2   Basic Considerations

DVE is an organizational model where a set of organizations, institutions or individuals (legally independent and geographically dispersed) develop a cooperation environment aimed at achieving a specific objective [1]. This environment allows the manufacture of products of higher quality and tailored to the needs of the market, incurring in a lower cost, a risk-sharing and a lower time to market, resulting in an improved response to the customers' requirements. In short, the goal is to create a best-of-everything organization [3] through the coalition of the complementary strengths of each member.

A DVE often evolves through four stages [4]: creation, operation, evolution and dissolution. Both in the creation and reconfiguration phases, the key element is the search, identification and selection of partners [4-6, etc.]. This task is complex and it is determined by the negotiation needs. It consumes, in addition, large amounts of time and resources which translates into a flexibility loss. The formation of a DVE requires, among other things, adequate information about potential partners, a certain degree of mutual trust, infrastructure to interoperate, common business practices and operational rules, etc. Ultimately, it depends on the potential partners are prepared to participate in such collaboration.

Guarantee this basic set of requirements could be a difficult challenge to achieve when temporal relations are considered. However, this does not happen with long-term collaborations that are not limited to a single business opportunity. Based on this idea a new organizational form has emerged to combine the advantages of DVE with those derived from long-term relationships. These new structures, known as VE Breeding Environments (VBE), are clubs of organizations prepared to work in long term relations and from which temporary coalitions emerge dynamically able to respond to different business opportunities [7].

3   The Model for the Selection of Partners

Based on the study of the different approaches existing in the literature to face the selection of partners in virtual architectures [5,8-10, etc.], we have created a generic and simple model with a multi-approach orientation which supports the advantages of the existing solutions and mitigates their weaknesses, a model that allows the formation of dynamic VEs could be a more effective and efficient process.

To achieve this goal, each involved actor can select the service providers in two different ways, as shown in Fig. 1.
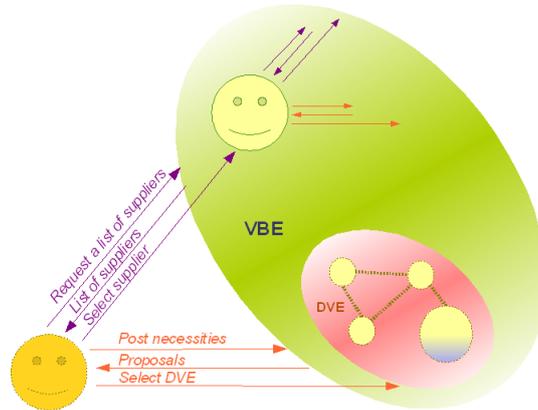

Fig.1. Selection process model.

First, when an entity needs a service it can ask the VBE the list of suppliers, and select one through a negotiation process that takes into account multiple criteria, both objective and subjective. On the other hand, it can also post its needs and expect a DVE (or sub-DVE) to be formed spontaneously in order to provide the requested service in an aggregate way. In both cases, the formation of the DVE is transparent with regard to the service requesting entity.

Allowing this simple choice our model makes possible the formation of DVEs that fit the different models and planning structures found in the literature and, in addition, any other possible combination. The model allows us the creation of any imaginable type of Dynamic VE and with the level of control we need in each moment (Fig. 2).
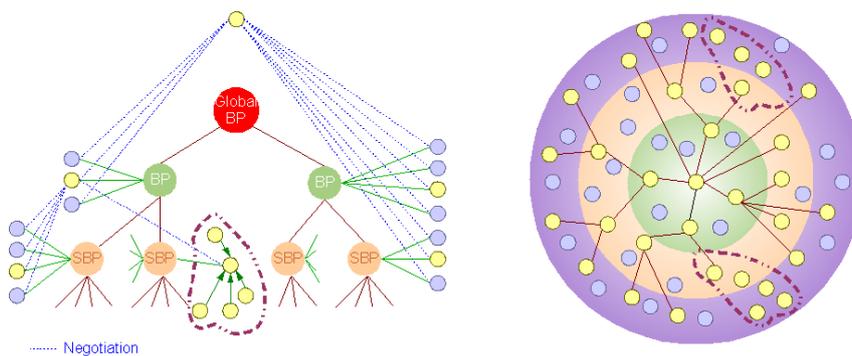

Fig. 2. Possible example of a DVE constituted from our model.

4   The Multi-agent Platform

To support the partner selection process in these virtual environments, any software tool has to permit a flexible network reconfiguration using, in turn, the optimal set of available resources, both human and material, financial, etc. To achieve this purpose, and after an elaborate planning process, we decided to employ a technology that is producing excellent results in different areas of knowledge where it has been applied: the multi-agent systems.

Specifically, we have work with Jade as implementation tool. Jade is a FIPA compliant agent framework, widely used by programmers and researchers, which has a great number of qualities such as flexibility, a good management of the platform and the exchange of messages, extensibility of the code, allows to create distributed applications on different machines easily, extensive documentation, etc. Furthermore, it is free software and the user group is very active.

Starting with this election and following the steps in the methodology proposed by Nikraz, Caire and Bahri [11] specifically for Jade, we have analyzed, designed and finally developed a framework for the multiagent application. This framework will serve as a basis for developing multiagent tools tailor made for each specific application domain.

The analysis phase aims to clarify the problem without any concerns about the solution [11]. Based on both the functionality of the platform and the main use cases of the platform, and following an iterative process of refinement, we identified the main types of agents (and their responsibilities), users, resources and relationships, (Fig. 3) as well as the physical layout of each agent.
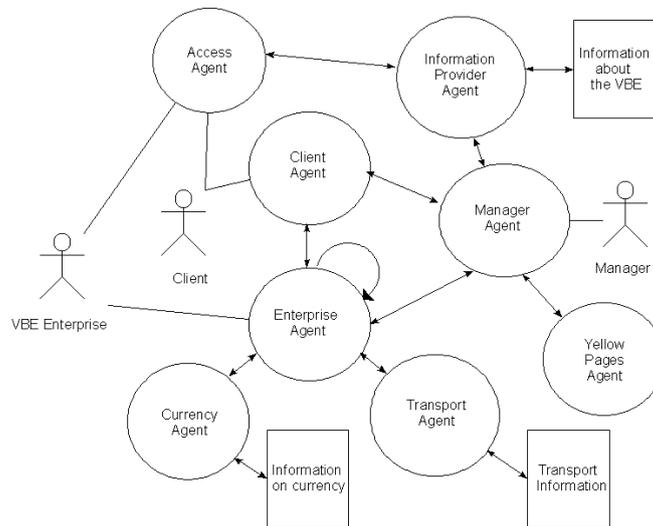

Fig. 3. Agent diagram for the multiagent platform.

After analysis it comes the design phase which aims to specify the solution. The adopted methodology considers Jade [12] as implementation platform, focusing specifically on the constructs provided by it (classes, concepts, behaviors, etc.), which is an important advantage. This allows us to reach a level of detail that is enough to achieve a direct transition to the development phase without having to adapt the results of the design phase to the selected agent platform, and with the possibility of a significant amount of code being generated by an automatic tool. Obviously, this decision provides a significant improvement in time and also a clearer vision of how to move forward on implementation.

Now, in this phase, we bear in mind all the responsibilities related to an acquaintance relation with another agent, leading to an interaction table for each agent type (all roles of the identified interaction protocol are implemented as JADE behaviors). Besides, the `MessageTemplate` objects are also specified to be used in these behaviors to receive incoming messages (Table 1).

Table 1. Interaction table extract for the enterprise agent.

Interaction	Resp	IP	Rol	With	Template
Negotiate with the provider agent	15	Contract net	I	Enterprise Agent	Conv-ID, Perf=CFP Prot=CFP, InReplyTo
Negotiate with the requesting agent	16	Contract net	R	Enterprise or Client Agent	Conv-ID, Perf=CFP Prot=CFP, InReplyTo

As regards the agent interactions with external resources, there are four active resources: these are the XML documents[2] with information about VBE members, the business processes, currency exchange and finally the necessary information to calculate the transportation cost or the distance between two locations. Based on the transducer approach the methodology suggests, we have used four different transducer elements with the necessary methods to have access and manage the information through DOM code. This action results in numerous advantages for the future: modularity, easy modification, reuse, etc.

Moreover, in our scenario there are four agents who need to interact with the users of the system: the enterprise agent, the client agent, the access agent and the manager agent. To make this user-agent interaction easier we have created a local graphical user interface (GUI) based on Swing for each one of these agents. Moreover, and given that agents and graphic elements are operating in different threads, we have used the `jade.core.GuiAgent` class specifically built for this purpose.

About ontologies, it is necessary to define an ontology for the application domain, the photovoltaic modules installation, which we have called the "main ontology": this ontology describes the business process architecture for the considered domain. However, we also use another set of ontologies associated with the different types of information we need for the operation of the platform. Specifically, we have created an ontology for the management which considers both users and their characteristics,

---

[2] Since the necessary resources are XML documents, we have used an *ad hoc* content language through the XMLCodec codec developed by the Team JADE [12].

their relations, the requested information, etc., and one for each auxiliary service we have defined (an ontology for the currency exchange and one for the transportation calculations).

After analysis and design we developed the framework of our multiagent tool, the DVEBreeder platform. For this work we have used different software applications and plugins which are shown in Fig. 4.
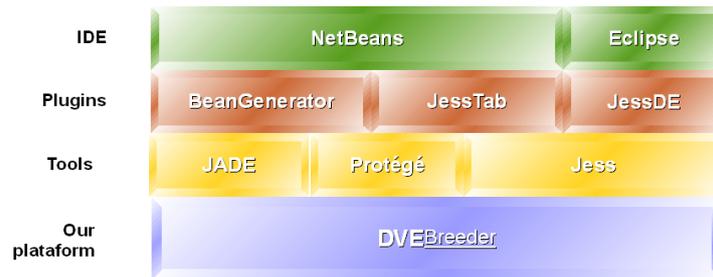

Fig. 4. Development tools overview.

Taking into account all these considerations and the Jade peculiarities, we have created a distributed multiagent platform (Fig. 5). This platform consists of a main container in which the manager agent and the service agents reside (in addition to the AMS and DF agents of Jade) and several remote containers where we can find the access agents and those agents representing the platform users (client or enterprise).
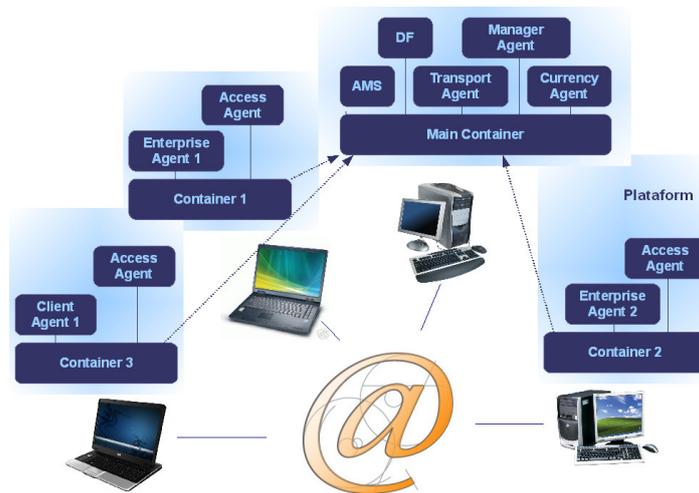

Fig. 5. The platform distributed architecture.

Finally, we have built an open and modular GUI for each of these agents. This allows us a quick and easy customization to any business domain. In Fig. 6 some agent GUIs of the DVEBreeder platform are shown.
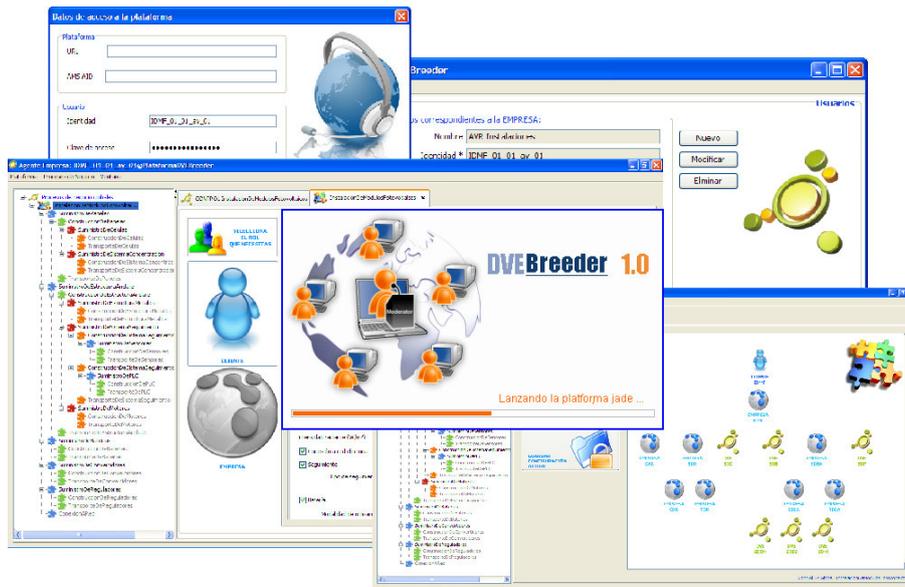

Fig. 6. Some DVEBreeder platform snapshots.

As mentioned above, we have worked on the formation of DVE oriented on the installation of photovoltaic modules. In this platform agents make their decisions based on a system of rules and facts implemented by Jess, a rule engine and scripting environment written entirely in Sun's Java language by Ernest Friedman-Hill [13]. By studying different scenarios we have validated both the platform and the initial operational model. Furthermore, we have proved that the coalition of multiagent technology and expert systems is a correct approach to addressing the selection of partners in the DVE context.

Conclusions

Companies must begin to adapt their structures to a new business model, the Dynamic Virtual Enterprise paradigm, if they want to ensure their survival in the present complex context. However, there are still some challenges to be overcome for the DVE becomes a reality on a large scale. One of such challenges is the right selection of partners because of its high relevance in the final product or service and, consequently, in the success of the DVE.

In an attempt to overcome these difficulties, in this paper we have presented a simple but innovative and comprehensive model to select the most appropriate partners regardless of the control level. Based on this model we have developed the framework of a distributed multiagent application capable of responding to the selection process needs.

Through the personalization of this platform we can deal with the creation process of any kind of Dynamic VE. Indeed, we have developed a multiagent platform for the formation of DVE aimed at the installation of photovoltaic modules in which agents make their decisions based on a set of rules and facts implemented by Jess.

Our work does not stop here and now we are working on customizing this platform to the real and specific case of the DVO creation aimed at child abuse prevention. For this work we have the collaboration of REA [14], a spanish regional association for the Defense of Children and Youth.

References

1. Sanz, P. "Partner Selection in Dynamic Virtual Enterprises". Ph.D thesis. University of Valladolid (2008)
2. Beer, M., Eisenstat, R.A., Spector, B.: Why Change Programs don't Produce Change. Harvard Business Review, 68 (6), 158--166 (1990)
3. Adams W.M., Wallas, R.M., Sengupta, A.: Collaborative Commerce: The Agile Virtual Enterprise Model. Pushing the Digital Frontier: Insights into the Changing Lands of E-Business. Penn State eBusiness Research Center, 242--262 (2001).
4. Camarinha-Matos, L.M.: Virtual Enterprise Modeling and Support Infrastructures: Applying Multi-Agent Systems Approaches. In: L.M. Camarinha-Matos, H. Afsarmanesh, (by invitation) in Lecture Notes in Artificial Intelligence LNAI Nº 2086, pp. 335--364, Springer, issue on Multi-Agent Systems and Applications (2001)
5. Petersen, S.A.: An Agent-based Approach to Support the Formation of Virtual Enterprises. Ph.D thesis. Norwegian University of Science & Technology (2003).
6. Fischer, M., Jähna, H., Teich, T.: Optimizing the Selection of Partners in Production Networks. Robotics and Computer-Integrated Manufacturing, 20, 593--601 (2004).
7. Afsarmanesh, H. (coord.). Key Components, Features, and Operating Principles of the Virtual Breeding Environment". ECOLEAD Deliverables (2005).
8. Rocha, A., Oliveira, E.: An electronic market architecture for the formation of virtual enterprises. In: Infrastructures for Virtual Enterprises, Kluwer, B.V., 421--432 (1999).
9. Ouzounis, E. K.: An Agent-Based Platform for the Management of Dynamic Virtual Enterprises. Ph.D thesis. Universität Berlin (2001).
10. Rabelo, R., Camarinha-Matos, L.M., Vallejos, R.: Agent-based Brokerage for Virtual Enterprise Creation in the Moulds Industry. In: E-business and Virtual Enterprises, Kluwer Academic Publishers, 281--290 (2000).
11. Nikraz, M., Caire, G., Bahri, P.A.: A Methodology for the Analysis and Design of Multi-Agent Systems Using JADE. International Journal of Computer Systems Science and Engineering, 21(2) (2006).
12. Jade – "Java Agent Development Framework", http://jade.tilab.com/index.html
13. Jess – "the Rule Engine for the JavaTM Platform", http://www.jessrules.com/
14. REA – "Asociación Castellano Leonesa para la Defensa de la Infancia y la Juventud", http://www.asociacionrea.org/