We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

5,500
Open access books available

135,000
International authors and editors

165M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Developing a Multi-agent Software to Support the Creation of Dynamic Virtual Organizations aimed at Preventing Child Abuse Cases

Pedro Sanz Angulo and Juan José de Benito Martín
University of Valladolid
Spain

1. Introduction[1]

Child abuse is a serious and worldwide problem that our society must try to completely eradicate. This goal, undoubtedly ambitious, requires an intensive work in the education of every individual and, of course, in the recognition of the children's rights.

Although in recent years significant progresses have been made in this sense, we must do everything in our power to adequately protect children, responding promptly and in a personalized way to each abuse situation. For this purpose, we must employ all the available resources and, furthermore, search for new solutions and tools.

In order to provide an adequate response, all those stakeholders (both entities and people) that are involved in the prevention, detection and intervention activities should work together creating collaborative networks in which they provide what they really do best, i.e., their core business.

However, network collaboration must face numerous challenges in order to become an effective instrument. Every abuse case is unique, so networks must involve different components in each new situation, and even these components can change during the performance: it is necessary, therefore, a dynamic and flexible networking model that allows the continuous evolution of resources and services, as well as the incorporation of new ones. These inter-collaborative problems in child abuse domain are difficult to solve without considering the new organizational models and, of course, the Information and Communication Technologies that make them possible. In this way, the *Dynamic Virtual Organization* (DVO) is probably the best organizational response to the problem.

In a DVO a set of business partners come together dynamically, on demand, and in accordance with the requirements of a specific problem, disappearing when those needs have been met. DVOs are rapid creation and fast dissolution organizations, constructed *ad hoc* the opportunity of collaboration.

But the effective creation of a Dynamic VO only is possible in the context of the *Virtual Organization Breeding Environments* (VBE). The VBEs are clubs of organizations prepared to work in long term relations and from which temporary coalitions emerge able to respond dynamically to the different business opportunities.

Based on these ideas and concepts, our work is currently focused both on the definition of a VO Breeding Environment for our community (Castile and León - Spain) and on the development of a multi-agent expert system able to support the partner selection process in an agile and efficient way.

The introduction of the VBE concept in conjunction with the use of software agents whose behaviour is guided by expert system modules is a novelty in the field of child abuse prevention. This combination will allow us optimizing the use of resources and it also will facilitate the communication between institutions and professionals, making possible a more agile response of the DVOs.

According to these ideas, the present document aims at justifying the use of DVO-VBE concepts for the child abuse problem and, secondly, at presenting the tool we are developing for the selection of partners in these scenarios, the VCAP (*Virtual Child Abuse Prevention*) platform.

About the first part, we present the problem of child abuse, its typologies and the different kinds of prevention, in order to establish the context of our job. We also present some initiatives, both domestic and international, that consider networking as the best way to prevent the child maltreatment. In addition, we analyze the DVO and VBE concepts and discuss their correspondence with the networking model in the child abuse domain.

With regard to the implementation of the multi-agent VCAP platform in which we are currently working, first we describe the partner selection model that we are using, which has been defined based on the study of the different existing approaches associated to the selection of partners. From this model, and following the methodology proposed for JADE platform, we describe the main elements and artefacts defined during the analysis and design of our platform. Finally, we present the multi-agent expert software we are developing, the different tools we are employing and its capabilities.

All this work would not be possible without our previous experience in the development of the DVEBreeder tool (Sanz & de Benito, 2009), a multi-agent expert system which has been designed to support the selection of partners in Dynamic Virtual Enterprise context. In particular, this tool allows to select the best combination of partners for the installation of photovoltaic solar panels.

2. Child abuse problem and networking

The United Nations Convention on the Rights of the Child 1989 defines child abuse as (United Nations, 1989) "all forms of physical or mental violence, injury or abuse, neglect or negligent treatment, maltreatment or exploitation, including sexual abuse, while in the care of parent(s), legal guardian(s) or any other person who has the care of the child." In short, we can say that there is child abuse if children's rights are not respected and there is not a response to their needs (Sanz et al., 2008); we cannot forget that abuse is, primarily, the lack of good treatment.

There is a great variety of abuse types depending on its forms (physical, sexual, psychological, neglect, etc.), the place, the involved actors, the degree of intensity, and so on. Each abuse situation is unique and requires, of course, a different approach, so an

appropriate response must be designed to each particular case. In this sense, prevention is the best possible approximation to the problem.

Prevention may take place at three different levels: primary, secondary and tertiary.

a. *Primary prevention* aims to decrease the incidence or onset of abuse cases, through the child advocacy and a special attention to their needs. It is applied to the general population, to make the community becomes aware about the problem and acquires positive habits and behaviours that prevent the appearance of the child abuse.

b. *Secondary prevention* focuses, in turn, on the concept of risk. This prevention is addressed to social groups, families or individuals classified as "high risk" to avoid those situations that end up in abuse.

c. Finally, *tertiary prevention* tries to reduce the duration and severity of the consequences of the problem; namely, it aims at decreasing the stage of rehabilitation or cure. It also requires the intervention in the context, family, etc., to prevent its recurrence.

As we will see later, our work can be perfectly included in the latter two levels of prevention, and especially in the tertiary one.

With regard to the roles of the different actors involved in the situation of maltreatment, it is necessary to establish a typology since obviously the institutional respond designed to punish the abuser has to be different to the necessary answer to attend or to cure the victim. From a systemic consideration, the number of stakeholders includes the aggressors, either adult or minor, the direct victims and those minors who are involved in an indirect way, in a situation of observation, being passive subjects.

Obviously, all citizens have, or can have some type of connection with the child maltreatment and this one can take place in different contexts although it tends to concentrate at school, at home and in the street (Lila et al., 2008).

2.1 Networking in the child abuse domain

Both in the Spanish Public Administration and in most of the governments of the western countries, the institutional answer to the child abuse is characterized by a frequent interaction between organizational systems, but also by the difficulty to coordinate all their functions. This difficulty is mainly due to the own nature of the problem of child abuse, which includes phenomena and situations as diverse and has such unique characteristics that make difficult to develop an integrated action.

Networking is, clearly, the best way to create a collaborative workspace where to achieve the pursued goals. It is based on the communication of the different agents and institutions, so an optimum exchange of information, both qualitative and quantitative, must exist between them.

The originators of the sociotherapeutic networking were Speck and Attneave (1974). They created therapeutic teams to intervene in families in crisis, in the United States, in order to break destructive patterns of family relationships and provide support for alternative options. Several years later, in the eighties, Elkaim (1989) carried out the first practice at European level in deprived areas of Belgium. Since then, the networking experiences have evolved and multiplied.

In our country, Spain, we find the networking pilot program of Burlada (Pamplona). Its promoters recognize that one of the challenges of any network program is (De Miguel & Fernández, 2002) to provide a process able to organize the different institutional levels and professional resources, in order to ensure the creativity and competence of each one of these instances.

These and other works, experiences and projects emphasize the need of networking as a means to achieve the ultimate goal of prevention and resolution of child abuse, establishing guidelines and procedures that normally are constructed *ad hoc*. However, they do not possess software tools capable of solving the different problems of such relationships, e.g. the need for a rapid intervention.

Our job will serve, precisely, to overcome this deficiency. We pursue the interconnection, in a quick and efficient way, of those professionals and institutions that must be part of the answer. Starting from a simple notification in the system, a dynamic consortium will be configured in order to respond to the specific abuse case in a personalized way and with a little human intervention.

3. The organizational basis of our work

There are a lot of similarities between the networking that a child abuse situation requires and the networking of a dynamic virtual organization. Below, we will summarize the more interesting ideas to understand our work, although previously it is necessary to describe this paradigm as well as the VO Breeding Environments concept.

3.1 The dynamic virtual organization

In order to survive in the present context, companies must be able to constantly meet their customers' needs, while improving productive efficiency and adapting continuously to a global, competitive and dynamic environment.

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 (Beer et al., 1990). Companies are forced to cooperate, sometimes even with their direct competitors, what has led to the introduction of multiple organizational concepts based on collaboration. In this sense, the paradigm of Virtual Enterprise (VE) represents one of the most relevant examples of collaborative networks.

A Virtual Enterprise is an organizational model that allows a number of organizations, institutions or individuals (legally independent and geographically dispersed) to develop a cooperation environment aimed at achieving a specific objective (Sanz & de Benito, 2009). This environment allows the manufacture of products or the provision of services of higher quality and tailored to the needs of the market, incurring in a lower cost, a risk-sharing and in a reduction of the time to market, resulting in a better response to the customer requirements. In short, the goal is to create a *best-of-everything* organization (Adams et al., 2001) through the coalition of the complementary strengths of each member.

There are a lot of 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, materials, know-how, expertise,...), 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, although a company can be capable of supplying a quality product with reduced costs, it does not mean it has the ability to adapt efficiently to those changes in customers` demand. This ability, known as agility, is really provided by the dynamic models of VE (DVE, *Dynamic Virtual Enterprise*), also known as *Agile Virtual Enterprises* (AVE).

The same idea is also very appealing in other non-business oriented contexts (Camarinha-Matos & Afsarmanesh, 2006), what leads us towards the paradigm of *Dynamic Virtual Organization* (DVO), a most suitable concept for our current work and which encompasses the former.

In a Dynamic Virtual Organization a set of business partners come together dynamically, on demand, and in accordance with the requirements and needs (Ouzounis, 2001) of the problem, disappearing when those needs have been met. DVOs are rapid creation and fast dissolution organizations (Browne & Zhang, 1999), constructed *ad hoc* the opportunity of collaboration.

A Dynamic VO often evolves through four stages (Camarinha-Matos, 2003): creation, operation, evolution and dissolution. To summarize, the phase of creation includes identifying the business opportunity, the selection of partners, the design of the company and its constitution. After that, the DVO reaches the operation stage which covers all activities of the current mission (Do et al., 2000). During this phase the structure of the DVO can need to be changed and a phase of reconfiguration will begin (Dang, 2004). In any case, when DVO is considered no longer effective it is dissolved.

As shown in Fig. 1., both in the phases of creation and reconfiguration, the key element is the search, identification and selection of partners (Camarinha-Matos & Afsarmanesh, 2001; Petersen, 2003; Fischer et al., 2004; 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 loss of flexibility.
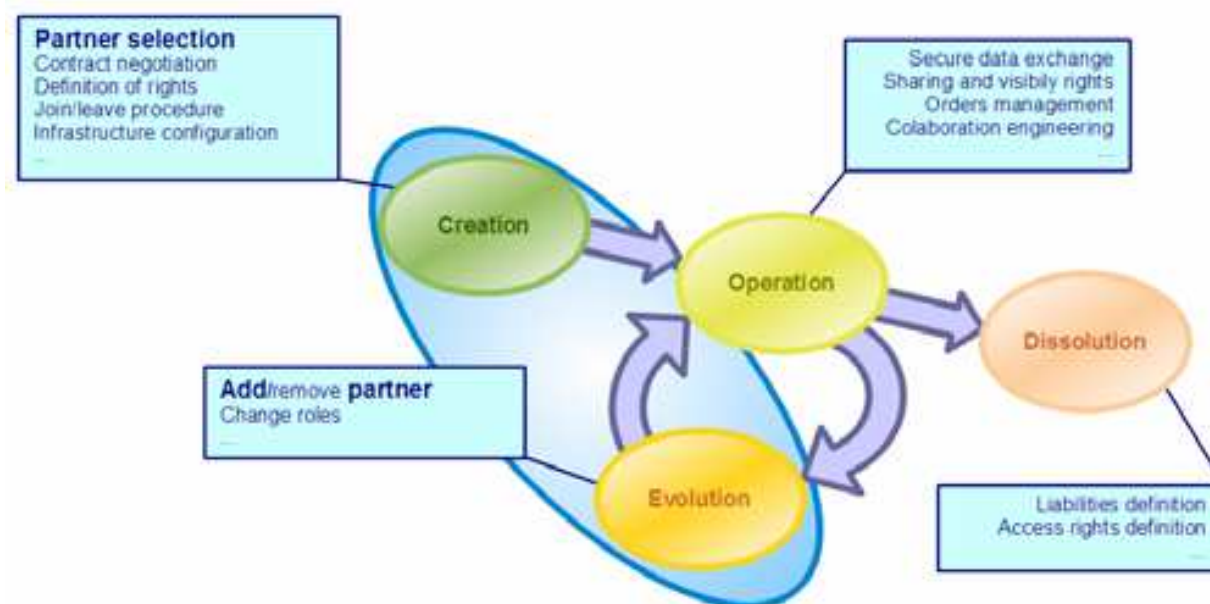

Fig. 1. The life cycle of a DVO and the scope of the partner selection process in it. Adapted from (Camarinha-Matos, 2003)

3.2 The DVO in the context of the virtual organization breeding environments

As mentioned before, the creation of a DVO whenever a new collaboration opportunity emerges requires large amounts of time and resources, causing a reduction of the agility associated with this business model. The effectiveness of the process depends largely on the

availability of adequate information about potential partners, their level of preparedness to engage in a Dynamic VO, the existence of trust, etc.

Most of all these problems have an easy solution when long term collaborations are considered. This is the reason why different authors have begun to consider that the formation of such organizations has to take place in the context of *Virtual Organization Breeding Environments* (Afsarmanesh & Camarinha-Matos, 2005), *pool communities* (Do et al., 2000), *virtual industrial parks* (Nayak, 2001), *virtual industry clusters* (Siqueira & Bremer, 2001; Rabelo et al., 2000), etc.,

The VO Breeding Environments, like the other equivalent concepts, are composed of organizations that are prepared to collaborate in long term relations (Camarinha-Matos et al., 2005). When a collaboration opportunity is identified, a subset of the VBE members can be rapidly selected to form a virtual organization. VBEs emerge as an evolution of *clusters* and *industrial districts* (Bremer et al., 1999; Mejía y Molina, 2002).

Fig. 2 shows how the creation of VO Breeding Environments and the genesis of Dynamic Virtual Organizations are different processes that are triggered by very different reasons.
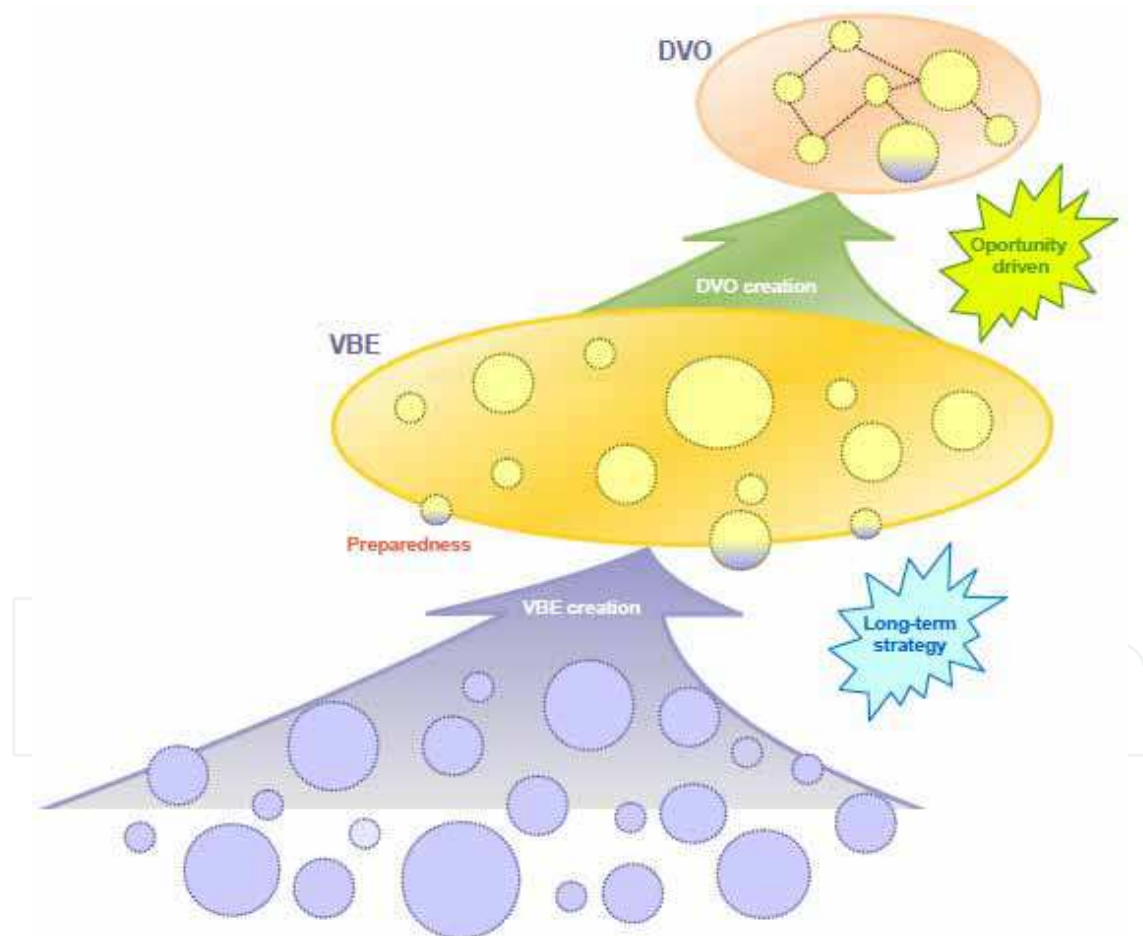

Fig. 2. Creation of DVOs in the VBE context. Adapted from (Camarinha-Matos et al., 2005)

A VBE is created as a long term association and its members are recruited from the "*open universe*" of organizations according to the criteria defined by the VBE administrators. By contrast, a VO is a temporary organization triggered by a specific collaboration opportunity, and its partners are primarily selected from the VBE members.

The VBEs simplify the configuration and establishment of DVOs since they solve, or reduce, many of the obstacles associated with the temporality of such organizations, contributing, thus, to their creation in a more efficient way. Partners of the DVO can, this way, benefit fully from unexpected changes in their context, providing an agile response to the problems or the opportunities. These entities are, in short, prepared to operate in an agile way following the definition provided by Goranson (1999).

The introduction of VBEs to solve the selection of partners' problem is an important organizational innovation. However, to make this concept operational and to ensure that the DVO really reacts swiftly and appropriately to the changes in demand, new tools and innovations of technological nature are required. After all, the concept of DVO absolutely depends on innovation to achieve its full potential and the agility that characterizes it.

3.3 Equivalence between DVO and networking in child abuse domain

As can be easily intuited, there are several similarities between networking that a child abuse situation requires and the collaborative networking of the DVOs.

We can start, for example, talking about the VO Breeding Environment. In the child abuse domain it is obvious that the VBE would be constituted by any entity or person involved in the solution: schools, social services, health centres, youth services, law enforcement agencies, state security bodies, and so on. All of them bring to the partnership their core business, what they do best, so the VBE contains all necessary processes that the abuse situation requires. In addition, these entities are required to share certain culture of work, ICT infrastructure, etc., to achieve the highest possible performance.

After identifying a new case of abuse (either in the school, or in a health centre, or through a complaint at the police station or in the child line, etc.), it is the time to go to the breeding environment and select the set of institutions and/ or persons that are the best prepared to respond to the particular problem.

The union of the selected entities configures, ultimately, the DVO responsible for providing the personalized answer to the child abuse situation. Naturally, different DVOs can arise from the breeding environment to provide this response, although the selection of the best one depends both on the requirements and needs of the particular situation of abuse and on the features and capacities of the partners.

After the selection process, partners need to work together and coordinate their actions in order to find an efficient and agile solution to the problem. This operation phase lasts until an adequate response to the initial problem has been provided. That moment coincides, precisely, with the dissolution of the DVO.

In addition, whenever it is necessary, more partners can be searched, or even their number can be reduced, the roles or the structure can change, etc., according to the evolution of the abuse case. In short, during the operation of the DVO an evolution-reconfiguration phase can take place.

Finally, we must also notice that those entities which form part of a particular dynamic virtual organization can be simultaneously partners of another consortium created in response to a different abuse case.

4. The VCAP multi-agent expert platform

Our fundamental objective is to enhance and develop a networking model through the creation of DVOs capable of facing child abuse problems in an agile and efficient way. For

this purpose, it is necessary to align the VBE and DVO concepts to the child abuse domain and provide efficient tools to support its formation and operation.

In this paper we propose a multi-agent approach to support the selection process in this context. This technology provides several advantages over other proposed methods, especially for its ability to manage complex and distributed problems. But in addition, our agents have an intelligent behaviour through a rule-based Expert System (ES) implemented in their decision module.

Through this platform the best combination of partners is defined dynamically, in real time, based on the system state, the predefined objectives and the features both of the problem and of the stakeholders. The identification and assignment of partners emerge from the interaction between agents, through different negotiation mechanisms. In this sense, we can introduce the Virtual Child Abuse Prevention (VCAP) concept.

4.1 The selection model

From the review of the multiple R&D projects and studies related to the identification and selection of partners in virtual organizations, the coexistence of two opposing positions can be easily observed.

A common solution is to follow a *top-down* approach, known as *planning* approach, where the planner designs the network and selects the partners that best fit his plan (Camarinha-Matos & Afsarmanesh, 2006). The opposite alternative is based on the use of a *bottom-up* (competition or emerging) approach; in this case, the client or broker announces the collaboration opportunity, waits till some consortia are formed spontaneously (by the initiative of some members), analyzes the received bids and selects the most appropriate according to his needs.

Intermediate solutions can be defined between these two extreme alternatives. Thus, for example, we found what we call *hierarchical* approach, where the broker or coordinator only defines the highest level business processes (abstract definition) and selects the appropriate partners for each one of them. Each one of these new members of the Dynamic VO is responsible for refining the design in accordance with its local capacities, so planning and detail are alternating in the process.

The planning approach is the most used solution in the research literature (Rocha & Oliveira, 1999; Petersen, 2003; etc.), including its hierarchical variant (Ouzounis, 2001), since it is easier to implement and allows greater control over the resulting DVO. The button-up approach (Rabelo et al., 2000), in turn, leads to forms of Dynamic VOs more efficient and agile, but its control is often far more complex.

Based on the study of the different approaches, we have created a generic and simple model with a multi-approach orientation which pools the advantages of the existing solutions and mitigates their weaknesses, making possible the formation of dynamic virtual organizations in a more effective and efficient way.

To achieve this goal, each involved actor in the selection process can select the service providers in two different ways (see Fig. 3).

First, when an entity needs a service it can ask the VBE for the list of suppliers and selects 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 the formation of DVOs (or sub-DVOs) and then selects one. In both cases, the formation of the resulting DVO is transparent with respect to the entity that is requesting the service.
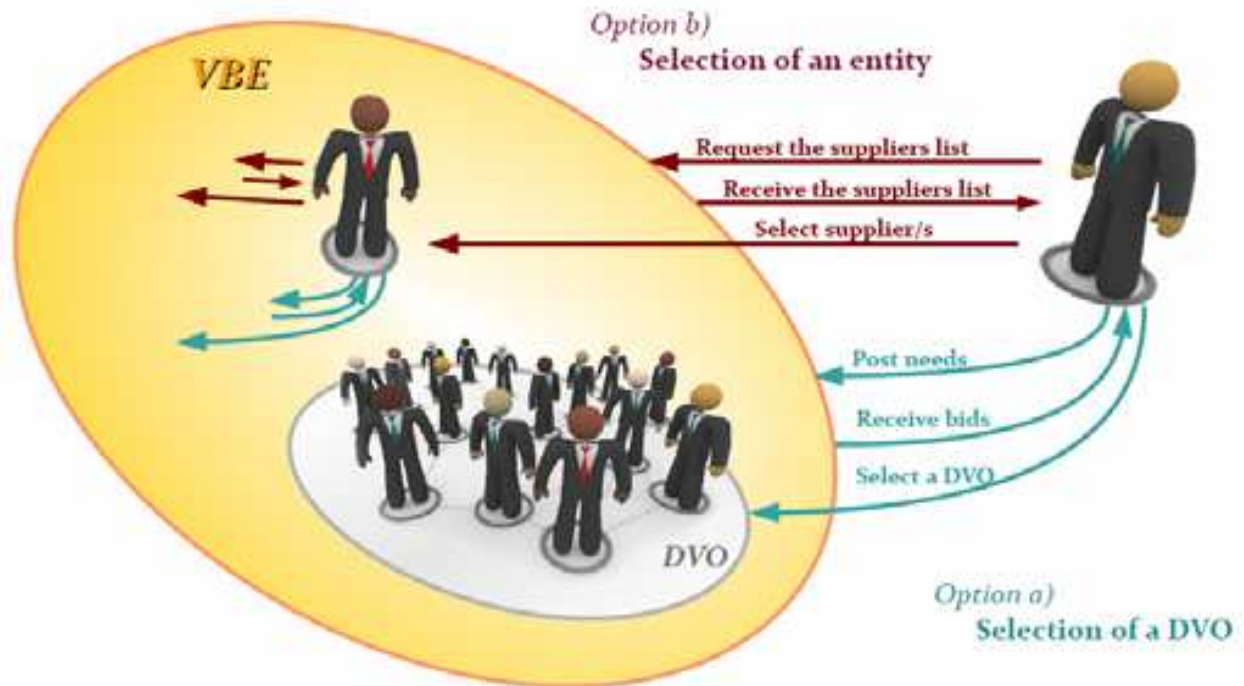
Fig. 3. Our model for the partner selection process

Allowing this simple choice our model makes possible the formation of DVOs that fit the different models and planning structures found in the literature and, in addition, any other possible combination, as can be seen in Fig. 4.
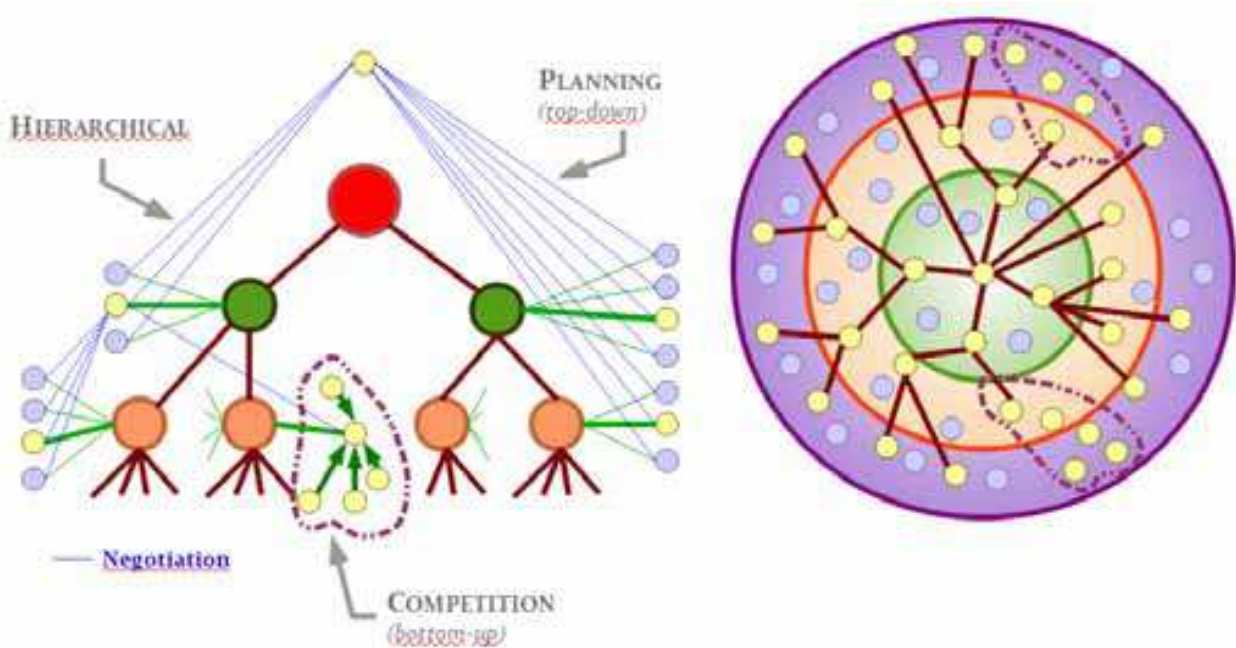

Fig. 4. A possible example of DVO created from our selection model

This model constitutes the selection conceptual basis for the VCAP multiagent platform. In the development of this platform we are following an iterative process of planning, analysis, design, construction, testing and validation. The following describes the most important aspects of this process.

4.2 Planning

Planning is the early stage of the software development cycle, where it is decided which technology-tool combination is the most appropriate solution for the problem.

As we know, the selection of partners is a key activity in the success of any collaborative network, particularly in the dynamic ones. This is, undoubtedly, a complex problem with no easy solution because involves a lot of elements, variables and factors of both objective and subjective nature. To address the resolution of this problem, over recent years different approaches have been used (computational automation - integer programming, AHP, etc. - Web services, multi-agent technology, etc.) what has led to a variety of solutions and R & D projects aimed at providing both the technological bases and the operating practices necessary to support the selection process in this kind of consortia.

Among the different existing solutions, the multi-agent technology highlights significantly. In fact, there are many features in the domain of DVOs that make these organizational models an appropriate application area for *Multi-Agent Systems* (MAS), as evidenced by several authors (Ambroszkiewicz et al., 1998; Camarinha-Matos and Afsarmanesh, 2001; Dignum & Dignum, 2002; Petersen, 2003; etc.).

These features have led to the emergence of multiple and diverse software solutions that use this technology to carry out the selection of partners in virtual organizations. However, the developed systems have a high degree of automation based primarily on quantitative issues, what prevents reflecting the full reality of companies and their decision-making mechanisms. Therefore, these systems do not lead to a fully realistic and satisfactory solution.

In order to provide an adequate solution to the problem that concerns us, it is first necessary to examine and understand how organizations make decisions, particularly those decisions aimed at building partnerships with other entities in order to resolve a case of child abuse. In this sense, we can see that such decisions are not made based on complex mathematical functions or statistical hypotheses; rather, they often rely on their "experience", "knowledge" and "intuition".

Therefore, to give an effective response, software agents should behave, think and act like a human expert in the considered domain would do, but how can we do this? For this purpose we have used the *Expert Systems* (ES). By definition, an expert system is a computer program that simulates the thought process (learning, memorization, reasoning, communication, etc.) of a human expert to solve complex decision problems in a specific domain. By this way, ES can (Castillo & Alvarez, 1991) store data and knowledge, draw logical conclusions, make decisions, communicate with human experts, explain their decisions, etc., and, as a consequence of all above, take actions.

Thanks to these systems, and in particular to a rule-based ES, we obtain agents with a rational behavior based on the knowledge and expertise of human experts in the child abuse area. Thus, without resorting to complex mathematical procedures, the agents are able to make the best decisions based on the needs of the problem and on the structured knowledge that has been supplied to them.

The introduction of the expert system is a clear innovation, from the technological point of view, over other existing multi-agent solutions.

After justifying the technology to use, the next step is to select the development tools and methodologies to employ in the building of the software application. In particular, we have analyzed and compared the major multi-agent standards, platforms and development methodologies. Similarly, we have analyzed the different tools for the building of the expert system, taking into account previous decisions related to the MAS. It should be noted that in the choice of these tools we have tried to make use of free software or, alternatively, software under the Academic Free License.

FIPA (*IEEE Foundation for Intelligent Physical Agents*) is, without any doubt, the multi-agent standard which has currently been accepted by a larger number of developers, since it incorporates all aspects of this paradigm described so far by both the research community and the industry. For this reason, in this work we have followed the FIPA specifications and JADE (*Java Agent Development Environment*) as the implementation tool. Jade is a fairly commonly used development environment, not only for its strict conformity with the standard, but also by other qualities such as flexibility, good management of both the platform and the message exchange, code extensibility, easiness of both debugging and development of distributed applications on different machines, extensive documentation, etc. It is also written in Java, it is free software and the user group is very active.

In short, we believe that the selection of FIPA and Jade is a fairly accurate solution, given the nature of our problem and the wide acceptance of both the standard and the tool. This election, however, significantly constrains the choice of the development methodology, because since 2006 there is a methodology (Nikraz et al., 2006) explicitly defined to address the analysis and design of Jade-based applications, which we have followed.

Another key element in the MAS building, especially to achieve efficient information exchange between agents, are the ontologies. An ontology is simply a model of the real world, and for the specific case of an agent, an ontology defines its environment. In this work we have chosen to follow the methodology proposed by Noy and McGuinness (2001), which faces the ontology construction from an iterative approach in which the ontology has to be assessed and streamlined to throughout its life cycle. As ontology development tool we have chosen Protégé, a Java-based graphical tool developed by the Stanford University.

Finally, we have addressed the issues associated with the construction of the expert system with which we are going to build the decision module of the most of the agents of our multi-agent platform. In the market there are a lot of programming languages (LISP, Prolog, OPS5 …), shells (EMYCIN, Crystal, Leonardo, XiPlus, Exsys, VP-Expert ...) and development environments (CLIPS, JESS, KEE, ART, Egeria , Kappa, etc.) which can be used for this purpose. In our case, and based on both our objectives and decisions summarized in the preceding paragraphs, we have finally decided to use JESS (*Java Expert System Shell*). Jess is a development environment with the same functionality, and even higher, than CLIPS but also it is based on Java what allows an easy integration with the Jade platform.

4.3 Analysis and design of the VCAP multi-agent expert system

The analysis of the platform has been made following the steps defined by the Jade methodology. Thus, we have identified the different types of users, agents and external resources in the system, along with their responsibilities and the acquaintance relationships that exist between them.

One of the major artefacts defined in this stage is the agent diagram. Fig. 5 shows the agent diagram for our platform.

In this diagram four types of elements can be distinguished:

- First, we identify the *users*, those individuals who must interact with the system: these elements are represented in the diagram by the UML actor symbol. In our case, we identify two different actors: the administrator of the platform and those entities that form part of the breeding environment. It should be noted that the same entity can have multiple roles in the platform: for example, VBE member-manager, VBE member-client (when a member of the breeding environment needs a service that other member from the own environment provides), or even client-VBE partner-manager.
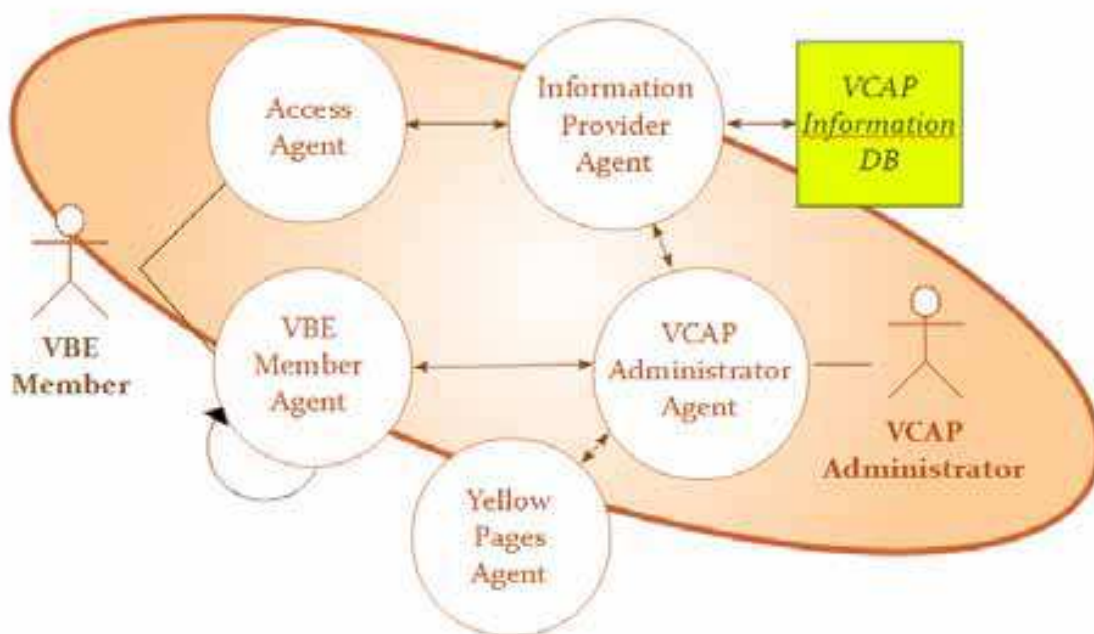

Fig. 5. Agent diagram for the VCAP platform

- In addition to humans, we also find those external systems that must interact with the system under development, i.e., the *resources*, which are represented in the diagram by rectangles. As can be seen, and contrary to what happens in UML case diagrams, the agent diagram provides an explicit distinction between humans and resources: this is due to that the interaction with people through a user interface presents some additional problems with respect to the interaction with an external system. In our case, external resources are simple data bases that have been built as valid XML documents.

- The *agents* that will form the multi-agent platform are represented by circles. In the picture we can appreciate that an agent has been added for each user (the *VBE Member Agent* and the *Administrator Agent*) and for every resource (the *Information Provider Agent*, which allows accessing to the information about the platform and its operational). Furthermore, an *Access Agent* has been added, whose sole function is to verify that the user-agent who wants to connect to the platform really belongs to it, and the *Yellow Pages Agent*, defined to identify the agents based on the services they provide. Although the yellow pages mechanism can be fully distributed across all

Developing a Multi-agent Software to Support the Creation
of Dynamic Virtual Organizations aimed at Preventing Child Abuse Cases
467

agents in the system, we have decided to adopt a centralized approach that maps completely to the *Directory Facilitator* (DF) agent provided by Jade, in order to save work in the successive phases of the development process.

• Finally, *relations* are represented in the diagram by arrows, which join the instances of the above items: this way we specify that such elements must interact in some way while the system is operational. Although most of the relationships shown in the figure are fairly obvious, it is necessary to comment on a couple of them. On the one hand, we see that the yellow pages agent is only related to the administrator agent: although this behavior may be more inefficient, this decision is quite deliberate because, by allowing that only the administrator agent can carry out both the registration and the search of services, we improve the security and robustness of the system. On the other hand, when the customer is internal (i.e., an entity belonging to the breeding environment), relations-negotiations will occur between the VBE members agents, although with different roles.

Another interesting artifact generated during the analysis is the agent deployment diagram, which indicates how agents are going to be physically deployed between the different hosts/ devices. The deployment diagram for our scenario is shown in Fig. 7.
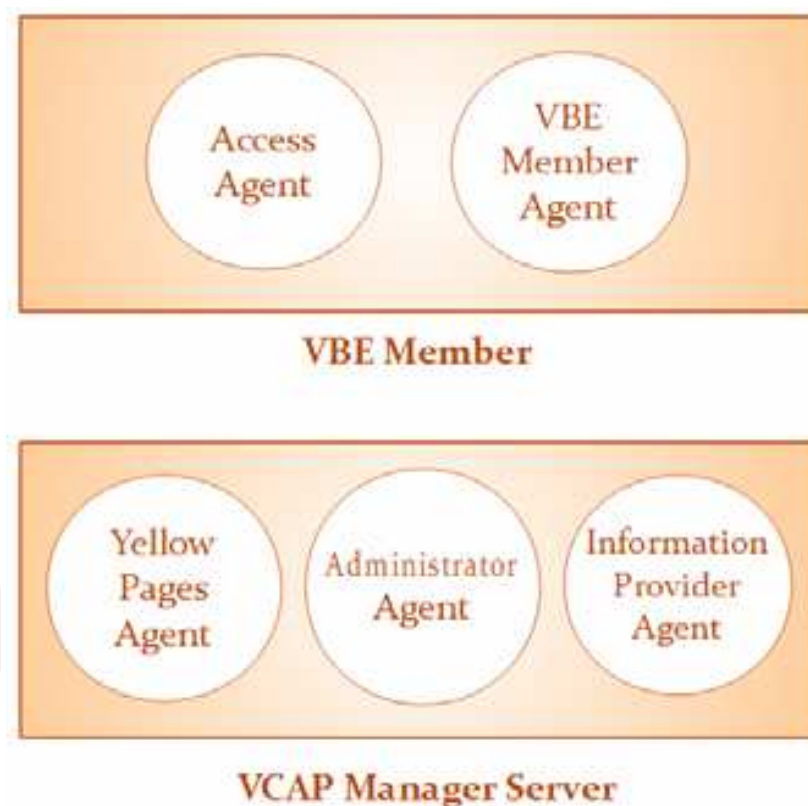

Fig. 7. Agent deployment diagram for our scenario

During the design stage, and prior to the implementation, we are working in the definition of different elements as the interactions between agents and the interaction protocols, message template (see Fig. 8), the description of the services to be registered/ searched in the yellow pages agent, the agent-resource and agent-user interactions, the internal agent behaviors, the content language, the ontologies, etc.

With regard to the yellow pages mechanism, which is maintained by the DF agent of Jade, VBE member agents can register the services they provide in the yellow pages catalog, while others can search for the services they need to carry out their activities in such catalog. In addition, the same agent can register one or more services and, simultaneously, can require any other ones from the other agents.

In the design phase we have also addressed the interaction with the external resources. In our case, we have considered only tree active resources, which are the XML documents that contain information about both the business process, and the activities that comprise it, and VBE members, including their behavior during their membership to the breeding environment.

```
MessageTemplate registroDeProcesos =
    MessageTemplate.and (
        AchieveREResponder.createMessageTemplate
            (FIPANames.InteractionProtocol.FIPA_REQUEST),
        MessageTemplate.and (
            MessageTemplate.MatchPerformative (ACLMessage.REQUEST),
            MessageTemplate.MatchConversationId (
                "registrar-proceso-de-negocio")));
```

Fig. 8. Example of a message template for a FIPA-REQUEST interaction

For these situations, the JADE methodology proposes to employ a transducer approach: this way we avoid embedding code to access data within those agents that need to consult the file with the information. For this reason, we are using two classes that contain the necessary methods to access and manage the information through DOM code. Thus, we get the information provider agent does not include inside the code, resulting in different advantages for the future: modularity, easy modification, reuse, etc.

Another issue to consider is the agent-user interactions. In our scenario we can easily identify three agents that need to interact with the users of the system (those that have an acquaintance relationship with an actor element in the agent diagram), namely the VBE-member agent, the administrator agent and the access agent. Although there are several ways to get the human-software interactions, we have considered using *Graphical User Interfaces* (GUI) which are by far the most common type of interface. Specifically, and given the current requirements of the application, we have only considered necessary to create local Swing-based GUI.

However, there is an interaction problem between agents and graphic elements, since these entities work in different threads. Although Jade has some ways to tackle this problem, we are using the `jade.core.GuiAgent` class, built specifically for this purpose.

In the design stage we are currently defining the ontologies of the system. In particular, we are working in the main ontology, i.e., the ontology associated with the child abuse domain: actors, types, business processes and their main characteristics, etc. We have also defined an ontology for the management of the platform, which considers the information about users, their characteristics, their relations, the last operational, etc.

Based on all these considerations, and bearing in mind the notion of container that Jade incorporates, we get the architecture that is depicted in Fig. 9.

Fig. 9. Distributed structure of the VCAP platform

As seen in this figure, there is a main container on the manager server in which the VCAP Administrator Agent resides, in addition to the Information Provider Agent, the Yellow Pages Agent (where the Resource Agents register their services) and the rest of agents that Jade incorporates for the platform management.

Also, there are multiple remote containers: specifically, so many remote containers as institutions and professionals the breeding environment has. In each of these containers we can identify the VBE Member Agent (associated to the different actors involved in the solution) and an Access Agent for security issues.

We have considered that all of these entities can provide their core business and, at the same time, be the promoters of the DVO formation process. It is also possible that a third entity or person starts the process, but for now we consider that the system is open only to the VBE members. Perhaps in the future the platform can be opened to everyone, but to do that previously it is necessary to solve some security problems.

The union of these remote containers with the main container results into the distributed agent platform shown in the Fig. 9.

4.4 Current development of the VCAP platform
The current development of the VCAP platform is possible thanks to our experience in the creation of the DVEBreeder tool (see Fig. 10), a multi-agent expert platform developed to support the selection of partners in the enterprise domain (Sanz & de Benito, 2009).

Fig. 10. Some snapshots of the GUIs of the DVEBreeder platform

To address the construction of the platform, we have used the tools and *plug-ins* that appear graphically represented in Fig. 11.
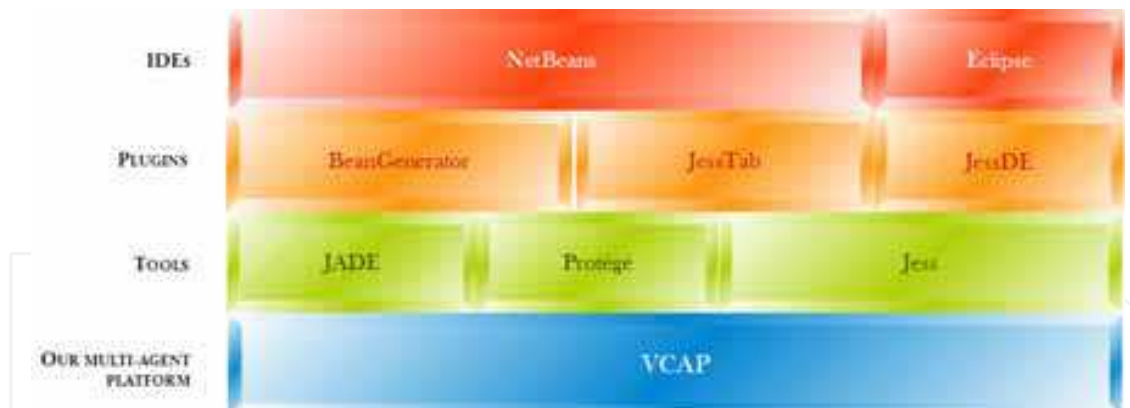

Fig. 11. Overview of the software tools used in the implementation

The VCAP platform inherits most of the features of its predecessor: the integration of multi-agent and expert system technologies; the identification of the intelligent agents is made through an multi-approach model that takes into account the VBE concept; the negotiation between autonomous agents is based on the knowledge embodied in their decision modules; the communication between agents is quite simple thanks to the platform used in the MAS development (Jade); the behaviour of the agents is quite similar to the behaviour that a human expert can have thanks to JESS (see Fig. 12), the rule-based expert system; it is flexible, dynamic and scalable; it is FIPA-compliant agent platform; the interface is intuitive and the code reuse is simple thanks to its open structure, etc.

Fig. 12. Example both of a Jess rule and of the GUIs where the decision modules of agents can be modified

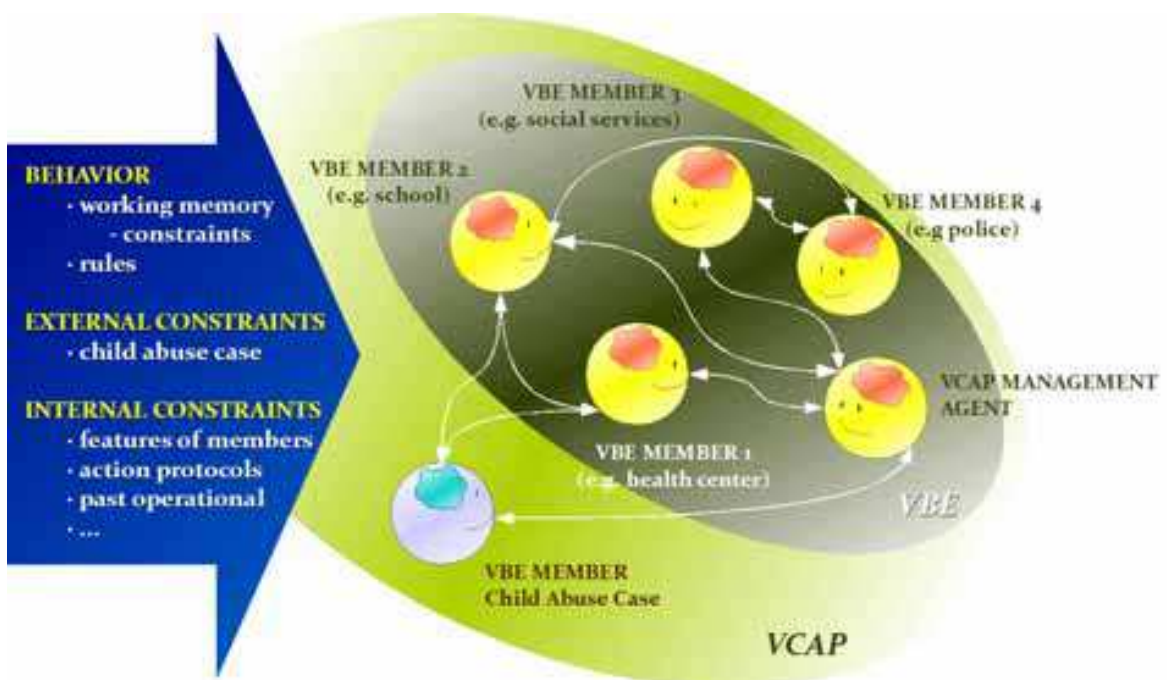Finally, Fig. 13 shows a summary of the platform performance in the application domain.

Fig. 13. Simplified scheme of the VCAP platform performance

5. Conclusions

In this article we have tried to justify the utilization of the DVO paradigm and the VBE concept as a valid way to create collaborative networking in the child abuse domain. Based

on both concepts, we have also presented our current work oriented towards the creation of a multi-agent expert system able to select and join those entities/ persons that can provide the best response for a particular child maltreatment case; in other words, we offer a tool for the creation of DVOs that emerge from the VBE context.

After having done the planning and the analysis, we are currently working in the design and implementation phases. To do this, we are defining some elements of the platform as e.g. the application domain ontology, the service descriptions, the agent decision modules, etc. Others, such as the agent interactions, the message templates, the interaction protocols or the ontologies for the management of the platform, are very similar to those established in the DVEBreeder platform, a tool created for the business domain, what undoubtedly will greatly facilitate the final implementation of the platform.

In summary, we believe that the VCAP software will be an innovative tool able to solve the Child Abuse Prevention problem in an agile and efficient way.

6. Acknowledgments

7. References

Adams W.M.; Wallas, R.M. & Sengupta, A. (2001). Collaborative Commerce: The Agile Virtual Enterprise Model. In: Nirmal, P., Judith, M.R. (eds.): *Pushing the Digital Frontier: Insights into the Changing Lands of E-Business*. Penn State eBusiness Research Center, pp. 242-262.

Afsarmanesh, H. & Camarinha-Matos, L.M. (2005). A Framework for Management of Virtual Organization Breeding Environments. *Collaborative Networks and their Breeding Environments*, Springer, Boston, pp. 35-48.

Ambroszkiewicz, S.; Cetnarowicz, K. & Radko, B. (1998). Enterprise Formation Mechanisms based on Mobile Agents. *Proceedings of Workshop on Intelligent Agents in Information and Process Management, KI'98*, pp. 23-34

Beer, M.; Eisenstat, R.A. & Spector, B. (1990). Why Change Programs don't Produce Change. *Harvard Business Review*, 68 (6), pp. 158-166

Bremer, C.; Mundim, A.; Michilini, F.; Siqueira, J. & Ortega, L. (1999). A Brazilian Case of VE Coordination. In *Networking Industrial Enterprises*, Springer, pp. 377-386

Browne, J. & Zhang, J. (1999). Extended and Virtual Enterprises - Similarities and Differences. *International Journal of Agile Management Systems*, 1 (1), pp. 30-36

Camarinha-Matos, L.M. & Afsarmanesh, H. (2001). Virtual Enterprise Modeling and Support Infrastructures: Applying Multi-Agent System approaches. *Multi-Agent Systems and Applications, Lecture Notes in Artificial Intelligence*, pp. 335-364

Camarinha-Matos, L.M. & Afsarmanesh, H. (2003). Elements of a Base VE Infrastructure. *Journal of Computers in Industry*, vol. 51, nº 2, pp. 139-163

Camarinha-Matos, L.M. & Afsarmanesh, H. (2006). Creation of virtual organizations in a breeding environment. In *Proceedings of INCOM'06*, St. Etienne, France

Camarinha-Matos, L.M. (2003). Infrastructures for Virtual Organizations - Where We Are.
    Emerging Technologies and Factory Automation, 2003. *Proceedings of ETFA '03.*
    IEEE Conference, vol. 2, pp. 405- 414

Camarinha-Matos, L.M.; Silveri, I.; Afsarmanesh, H.; Oliveira, A.I. (2005). Towards a
    Framework for Creation of Dynamic Virtual Organizations. *Collaborative Networks
    and their Breeding Environments* (PRO-VE'05), Springer, pp. 69-81.

Castillo, E. & Alvarez. E. (1991). *Expert systems. Uncertainty and Learning.* Elsevier Applied
    Science and Computational Mechanics Publications, London and New York

Dang, V.D., (2004). *Coalition Formation and Operation in Virtual Organisations.* PhD, School of
    Electronics and Computer Science, University of Southampton

De Miguel, M. & Fernández, M. (2002). Detección precoz del maltrato infantil. Programa
    piloto de trabajo en red. *Anales del Sistema Sanitario de Navarra,* 25(2), pp. 25-34

Dignum, V. & Dignum, F. (2002). Towards an Agent-Based Infrastructure to Support Virtual
    Organisations. In Camarinha-Matos, L.M. (ed.), *Collaborative Business Ecosystems and
    Virtual Enterprises,* Kluwer Academic Publishers, pp. 363-370

Do, V.T.; Halatchev, M. & Neumann, D. (2000). A Contextbased Approach to Support
    Virtual Enterprises. *Proceedings of the 33rd Hawaii International Conference on System
    Sciences,* vol. 1, nº 4-7.

Elkaïm, M. (1989). *Las prácticas de la terapia de red.* Barcelona: Gedisa

Fischer, M.; Jähna, H. & Teich, T. (2004). Optimizing the Selection of Partners in Production
    Networks. *Robotics and Computer-Integrated Manufacturing,* 20, pp. 593-601

Goranson, H.T. (1999). *The Agile Virtual Enterprise: Cases, Metrics, Tools.* Quorum Books

Lila, M.; Herrero, J & Gracia, E. (2008). Multiple victimization of Spanish adolescents: A
    multilevel análisis, *Adolescence,* 43, pp. 333-350

Mejia, R. & Molina, A. (2002). Virtual Enterprise Broker: Processes, Methods and Tools. In
    Collaborative Business Ecosystems and Virtual Enterprises, Springer, pp. 81-90

Nayak, N.; Chao, T.; Li, J; Mihaeli, J; Das, R.; Derebail, A. & Hoo, J.S. (2001). Role of
    Technology in Enabling Dynamic Virtual Enterprises. *Proceedings of the International
    Workshop on Open Enterprise Solutions: Systems, Experiences and Organizations,* Rome,
    Italy

Nikraz, M.; Caire, G. and Bahri, P.A. (2006). A Methodology for the Analysis and Design of
    Multi-Agent Systems Using JADE. *International Journal of Computer Systems Science
    and Engineering,* 21(2)

Noy, N.F. & McGuinness, D.L. (2001). *Ontology Development 101: a Guide to Creating your First
    Ontology.* Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and
    Stanford Medical Informatics Technical Report SMI-2001-0880.

Ouzounis, E. K. (2001). An *Agent-Based Platform for the Management of Dynamic Virtual
    Enterprises.* PhD thesis. University of Berlin.

Petersen, S.A. (2003). *An Agent-based Approach to Support the Formation of Virtual Enterprises.*
    Ph.D thesis. Norwegian University of Science & Technology.

Rabelo, R.; Camarinha-Matos, L.M. & Vallejos, R. (2000). Agent-based Brokerage for Virtual
    Enterprise Creation in the Moulds Industry. In *E-business and Virtual Enterprises,*
    Kluwer Academic Publishers, pp.281-290

Rocha, A. & Oliveira, E. (1999). An electronic market architecture for the formation of virtual
    enterprises. In *Infrastructures for Virtual Enterprises,* Kluwer, 153, pp. 421 - 432

Sanz, P. & De Benito, J.J (2009). Design and Implementation of a Multi-agent Framework for the Selection of Partners in Dynamic VEs. *Leveraging Knowledge for Innovation in Collaborative Networks*, Springer, pp. 341–348.

Sanz, P.; Villa, M.E.; Manso, M.A. & De Benito, J.J. (2008). Aplicación de la Tecnología Multiagente y los Sistemas Expertos al trabajo en Red para la Prevención del Maltrato Infantil. *Proceedings of the IX Congreso Estatal de Infancia Maltratada.* Valladolid. Spain

Siqueira, J. E. M. & Bremer, C. F. (2000). Action research: The formation of a manufacturing virtual industry cluster. *E-Business and Virtual Enterprises: Managing Business-to-Business Cooperation*, Kluwer Academic Publishers, pp. 261– 68

Speck, R. & Attneave, C. (1974). *Redes Familiares.* Amorrortu Editores, Buenos Aires

United Nations - Office of the High Commissioner of Human Rights (1989). *Convention on the Rights of the Child*, resolution 44/ 25 of 20 November 1989. Last visit in February 2010. Available at http:/ / www2.ohchr.org/ english/ law/ crc.htm
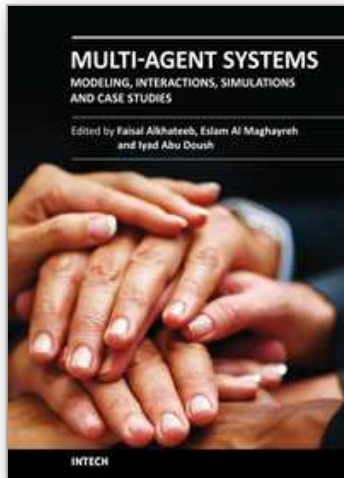
Multi-Agent Systems - Modeling, Interactions, Simulations and Case Studies

Edited by Dr. Faisal Alkhateeb

A multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent or monolithic system to solve. Agent systems are open and extensible systems that allow for the deployment of autonomous and proactive software components. Multi-agent systems have been brought up and used in several application domains.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Pedro Sanz Angulo and Juan José de Benito Martín (2011). Developing a Multi-agent Software to Support the Creation of Dynamic Virtual Organizations aimed at Preventing Child Abuse Cases, Multi-Agent Systems - Modeling, Interactions, Simulations and Case Studies, Dr. Faisal Alkhateeb (Ed.), ISBN: 978-953-307-176-3, InTech, Available from: http://www.intechopen.com/books/multi-agent-systems-modeling-interactions-simulations-and-case-studies/developing-a-multi-agent-software-to-support-the-creation-of-dynamic-virtual-organizations-aimed-at-

INTECH
open science | open minds