Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE)

Saleh AlZahrani, Aladdin Ayesh and Hussein Zedan
De Montfort University
The Gateway, Leicester LE1 9BH, England, UK
Email: salehsz, aayesh, hzedan@dmu.ac.uk

Abstract: Grids are increasingly being used in applications, one of which is e-learning. As most of business and academic institutions (universities) and training centres around the world have adopted this technology in order to create, deliver and manage their learning materials through the web, the subject has become the focus of investigate. Still, collaboration between these institutions and centres is limited. Existing technologies such as grid, web services and agents are promising better results. In this paper we support building our architecture Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE) by combining those technologies via Java Agent DEvelopment Framework (JADE). By describing these agents in details, we prove that agents can be implemented to work well to extend the autonomy and interoperability for learning objects as data grid.

Keywords: Data grid, Regional grid, Multi-Agent System (MAS), JADE, E-learning, Intelligent, Autonomous, Distributed data grid, Learning Objects (LO), Search.

1. Introduction

E-learning has been increasingly used by both academic institutions and businesses for learning and training activities. Various types of e-learning platforms and tools have been introduced in many different education institutions and private training centres. Many technologies include web services (Rodriguez, Anido-Rifon, & Iglesias, 2003), grid computing and data grid technology (Yang & Ho, 2005) and agent technology (Sousa, Silva, Teixeira, & Filho, 2006) have been integrated into e-learning environments to enhance the architecture.

Web services have emerged as a paradigm of distributed computing, and have been proposed as an intermediary framework for the integration of standard compliant e-learning platforms in order to eventually embrace advantage of the benefits offered by their technology (Rodriguez et al., 2003).

Data grid technology is another supporting technology for e-learning services in order to make learning materials such as Learning Objects (LO) sharable by learners in different sites (Yang & Ho, 2005). In data grid, replication services can be used to enhance the performance

in reliability, scalability and fault tolerance (Chervenak et al., n.d.) (Guy, Kunszt, Laure, Stockinger, & Stockinger, Edinburgh, Scotland, July 2002.).
Agents can provide both useful abstraction at data grid environment and very dynamic and robust services. Using the agents' essential powers is strongly recommended in grid environments.
MAG (Mobile Agents Technology for Grid Computing Environments) is developed by Federal University of Maranh˜ao, Brazil. The aim of the project is developing free software infrastructure based on mobile agents technology that allows the resolution of computationally intensive problems in computer grids. MagCat extends MAG to handle applications that manipulates huge amount of data. Although the multi-agent system MagCat has its search agent (known as SearchAgent) which is responsible for performing queries in distributed metadata repositories, this agent does not analyse the result of its search (Sousa et al., 2006).
These technologies have not been adopted cooperatively and collaboratively to support e-learning services. Agent and data grid architectures seem different from each other. As a matter of fact, we can learn from one in order to improve the other (Thompson, 2004). E-learning services are composed of many components which are part of distributed systems. These components and the system as a whole are designed to be cooperative. Grid and agent communities are pursuing the development of such distributed systems (Foster, Jennings, & Kesselman, 2004). In our architecture, we intend to support e-learning services using these technologies.
The organisation of the paper is as follows. First, in section II we present a background which includes grid computing and agent and information management. In section III we present an overview of our architecture Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE). This includes descriptions of architecture components and regional grid structure. In section IV, we introduce agents' specifications. This includes MAS-based e-learning, agent architecture, and agents formalisation. In section V, we introduce the implementation which includes the platform, registries and multi-agent systems, and case study. Finally, in section VI we conclude the paper and future work.

2. Background

The following is background about the technologies mentioned in the previous section in order to present an introduction and definitions. Those technologies are adopted to be embedded in our model to produce a dynamic e-learning environment. Those technologies include grid computing, agent and its role in data management, and finally learning objects.

A. Grid Computing

Grid computing provides an environment where a widely distributed scientific and academic community shares its resources across different administrative and organisational domains. The purpose of grid computing is to solve large-scale computing and data-intensive applications and collaborate in a wide variety of disciplines. Grid computing, therefore, enables the creation of a virtual environment which facilitates physical resources across different administrative domains in order to be beneficial; these resources are then abstracted into computing or storage units that can be transparently accessed and shared by large numbers of remote users.

Data grid is concerned with massive datasets and remotely separated storage units organised in a virtual environment. As a result of the increase of learning materials (Learning objects) and the need for huge masses of information to be archived and shared among academic institutions and training centres, data grids become an indispensable technology in learning fields. E-learning platforms and systems have been adopted, developed and published. These platforms and systems are based on client-server, peer-to-peer and web service architectures (Pankratius & Vossen, 2003).

Based on some papers written in grid fields, there are few extant data grid topologies which have been implemented as case studies or prototypes.

One of these topologies is the tree type (Lamehamedi, Shentu, Szymanski, & Deelman, 2003). The European Organisation for Nuclear Research (CERN) has adopted tree topology in its data grid. The CERN project is implemented as a tree topology consisting of three levels; the root (called Tier 0), intermediate (called Tier 1), and user (called Tier 2) levels (Nuclear Research (CERN), 2006). Another architecture which produced a platform using a Sharable Content Object (SCO) repository is based on data grid technology. This platform integrates the technology of data grids and Shareable Courseware Reference Model (SCORM) (Yang & Ho, 2005). Although this architecture uses Globus Toolkit middleware to accomplish learning processes it, does not exploit agent technology in its architecture.

There are two known Data Grid Management System (DGMS) middleware responsible for controlling and managing data grid within the grid environment. The _rst middleware is Storage Resource Broker (SRB). San Diego Supercomputer Center (SDSC) develops this middleware which supports shared collections that can be distributed across multiple organisations and heterogeneous storage systems (Center, 2005). The second middleware is OGSA-DAI (Open Grid Services Architecture-Data Access and Integration). OGSA-DAI is a middleware product that allows data resources, such as relational or XML databases, to be accessed via web services (Team, 2007).

B. *Agent and Information Management*

Agents have many different definitions, one of which is that they are any entities that perceive their environment through sensors and act on that environment based on their own reasoning capability. Examples include human, robotic and software agents (Russell & Norvig, 1995). Agent technology has been exploited heavily on the web through applications in many fields including industry, business, education and training. One of the most famous agent applications is video games, which have become a large part of many people's lives. Applying agent technology in video games has many aspects. One of the obvious benefits of video games is the elimination of risk to human life involved in any real-world application. They also make an excellent testbed for techniques in artificial intelligence (Laird & Lent, 2000).

Information management has an important role in any enterprise system. The Internet and World Wide Web (WWW) refer to massive distributed datasets. If we want these datasets and data to be beneficial, we have to organise them in such way that they can be managed. One of the most well-known methods of managing information is by using agent technology. In the context of information management, the types of agents depend on their purposes and functions. There are three types of agent (Stathis, Bruijn, & Macedo, 2002). The first type is a personal service agent whose purpose is to model the interactions of community members with personal devices, the second type is a location service agent whose purpose is to models the interactions of community members with shared devices placed in specific community locations, and the third type is a memory service agent which models how the information content is stored and disseminated to people or locations. Agent technology has been exploited in e-learning environments for many reasons, especially for integration with other technologies to build more robust, scalable and e_cient systems. Many e-learning architectures have been proposed using integration between agent technologies and web services (Hussain, 27-28 Aug. 2005). Although this architecture integrates agent and web services technologies, it does not exploit data grid technology.

C. Agent-based Grid

There are many projects which have been built based on grid, web service, and agent technologies. Those projects were built according to the user and organisation needs. At the same time those projects were produced based on different approaches and techniques. Yet there are some issues not resolved in those projects and at the same time the aim of using agents technology in our model is different from them. These differences will be detailed in the next chapter. The projects include ChinaGrid and ESESGrid (Engineering Structure Experiment and Simulation Grid).

ChinaGrid (Hai Jin and Li Qi, 2005) project is based on Open Grid System Architecture (OGSA) ChinaGrid is designed as four levels architecture. The levels are physical resources level, common support platform level, application support platform level, and applications level. The first level is physical resources level which takes care of all kinds of physical resources, such as computational resources, storage resources. The second level is common support platform level, which its kernel component is ChinaGrid Support Platform (CGSP). It provides the virtualization for the physical resources level and expands from the core of Globus Toolkits. The third level is application support platform level, which provides the common Problem Solving Environment (PSE) to support the different requirements in various research domains. Application professionals and developers use this level to deploy grid-enabled applications. Finally, the fourth level is applications level, which is the general user interfaces for applications over ChinaGrid, so all requests to ChinaGrid are sent through this level. This level is designed for application users and researchers who need know their application specific knowledge.

ESESGrid (Wang Li, Wang Cong, Long Hao, and Di Rui-Hua, 2008) is designed to share resources for engineering structure researchers in and collaborative work environment. ESESGrid data acquisition system is based on open, common standards and compatible with the relevant norms of WSRF and WS-Notification to build distributed heterogeneous and dynamic resource management in engineering structure research. The structure is composed of four layers. There are local site, agent layer, grid core service layer, and client side. The first layer local site gets real-time experiment data from the experiment centre. The second layer is agent layer, which all agents are responsible for the real-time experimental data management and access. The third layer is grid core service layer. It supports the integration and share of heterogeneous resources. The forth layer is client side which is a means for administrators and common users to use the system.

3. RDADeLE

RDADeLE architecture was designed to achieve and fulfil objectives and aims designed for agent-based e-learning environment needs. These aims include:

1- Dynamic e-learning environment. This aim is achieved by using active components, agents here, which are responsible for most activities in the environment.
2- More coherent e-learning environment by using multi-agent system technology. Agents within multi-agent system have infrastructure specifying communication and interaction protocols. These protocols support the coherence in RDADeLE.
3- Scalable environment: the number of students, members of learning centres, grid services, and other RDADeLE components are increasing. This why scalable environment is required in order to accommodate all components (agents in our architecture).
4- Interoperable environment: environments in general and e-learning environments in particular are not using the same standards to communicate. Interoperability in e-learning environments plays a big role in communication between components in the same environment or between different environments.

RDADeLE is our architecture, which is different from other and above architectures mentioned in the previous section in many ways. The following is the main difference:

1- RDADeLE is divided into a number of regional grids in which each region has its own components as shown in Figure 3 (AlZahrani, Ayesh, & Zedan, 2008). These regional grids have been adopted in RDADeLE for many reasons. It is recommended that massive number of data (data grid) divided into smaller number of segments in order to reduce problems or congestions may occur. Another reason is that such a divided regional grid gives members of each regional grid flexible way to customise the regional grid according to the characteristics of members and environment. Yet another reason is that for any reason if a single regional grid is disconnected from the RDADeLE that will not affect other regional grid in particular and the whole system in general.
2- The multi-agent system is embedded in the RDADeLE. Mulit-agent system has many features which strengthen of RDADeLE. These features include coherence, interoperability, and scalability.
3- There are three layers in RDADeLE: learner (user) interface layer, agents' layer, grid service layer. The first layer is the learner interface which is considered as a means for entities outside RDADeLE to communicate to it. The second layer is the agent layer which is the core part of the RDADeLE. There are three types of agent in this layer administrative agent (AA), regional agent (RA), and sensor agent (SA). More details will be provided later in this chapter. Finally the third layer is the grid layer which is data grid management system. This layer is responsible for connecting between learning objects (LOs) repositories in the same regional grid.

A. Architectural Components

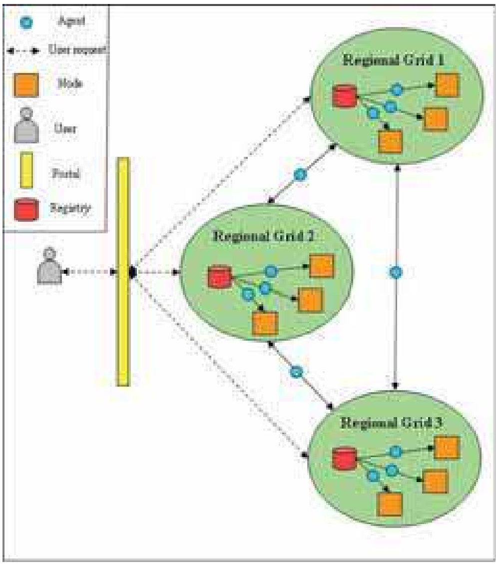

Fig. 3. RDADeLE Overview

Figure 3 shows an overview of our architecture (RDADeLE) which consists of agents providing the internal structure. These agents are triggered when the state of e-learning environment is changed and when a request is initiated by a user. The user in our architecture refers to an end user or a requester representative. Dotted lines show the request path which travels to regional grids in order to search for LOs. The path also shows the results returned to the user. The portal is a thin client which provides a user interface. The portal is a means to enable the user to send a request and receive a response to and from the system. Using data grid and thin client (portal) enables learners, employees and the general public to search for and collect information about (LOs), learning units (LUs), courses and degree plans from nodes all over e-learning environment. The solid lines show the interaction between system components within each regional grid and between grids themselves. On the one hand, the role of agents within each regional grid is both to help users to search and retrieve LOs and update each regional registry. On the other hand, agents traveling between regional grids are responsible for controlling the data passing between them according to assigned constrains which are part of the regional grids' properties.

The square shapes represent nodes. Nodes represent the locations of servers which provide grid services. Grid services include providing educational materials (i.e. Learning Objects LOs). Each regional grid has one or more server which represents one or more academic institution or training centre.

B. Regional Grid Structure

Regional grids each represent one or more educational institutions and training centres. What concerns us is the grid services provided by these sites. Grid services in our architecture will make repositories of LOs and their metadata available to requesters. Each repository represents an institution or training centre as shown in Figure 4.

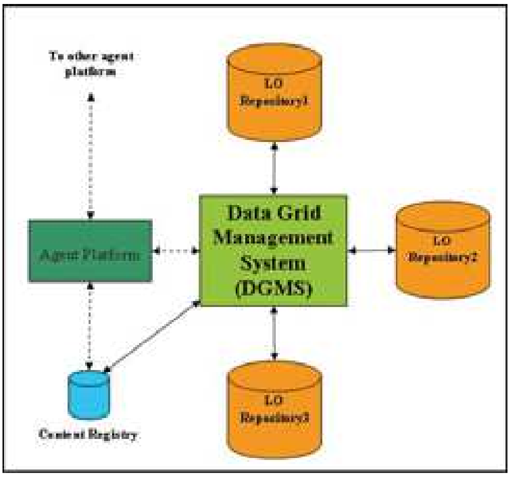

Fig. 4. Architecture of Regional Grid Node of RDADeLE

Reasons for using grid computing in our architecture (particularly data grid) include the following. Data grid provides an optimal organised platform for massive and remote separated storage units (LOs) in a virtual environment. These LOs are needed to be archived and shared among academic institutions and training centres.

Our purpose in distributing the registries in our architecture is based on an approach which will be explained in this section. There are two approaches which could be adopted in our architecture. The first is to create a central global registry. The content of this global registry is the contents of all registries in each regional grid. Users can discover all LOs in all regional grids via the global registry. At the same time, registries in each regional grid are used to discover only that grid's LOs. Users can access global and regional registries to discover LOs. The second approach is based on assigning one regional registry for each regional grid. In this approach there is no central registry with which to discover LOs. Instead, regional registries are used to discover regional LOs. If any regional grid has been disconnected from the global grid other active regional grids will not be affected. In RDADeLE, members (users) of academic institutions or training centres represent the active entities, while the regional grids support the infrastructure:

- A member may dynamically connect (join) and disconnect (leave) from RDADeLE.
- Regional grid is supposed always to be connected to RDADeLE.
- Each regional grid has one registry which publishes all LOs of institutions connected to it.
- A registry of a particular regional grid is updated according to the number of LOs repositories which connect or disconnect from the regional grid.

- Each regional grid has its own constraints which are considered part of its properties. Information is passing between regional grids according to these constrains.

Reasons for dividing our architecture into regional grids (regional segments) include the following. Such a division helps produce a sound structure. This gives the designer a flexible approach of constraint-based segments. The other reason is that regional grids could include one or more countries which have common cultural or demographic properties. This will ease the way for any institution or training centre to connect to the regional grid with the similar properties. Yet another reason that dividing our architecture provides a exible constraint setting for particular regional grids, which is useful in utilising each regional grid in particular and the whole global grid in general.

4. Agents' Specification

We intend to create dynamic e-learning environment which depends on data grid and multi-agent technologies. In our architecture, services are distributed since each regional data grid has its own registry. However, requests from portal could be massive which affects services to be a bottleneck. Using distributed agents in this context will help in resolving part of this problem. Agents in our architecture have been designed to be intelligent and autonomous. In order to make agents autonomous and have flexible behaviours, they have to be reactive and social. Reactivity means that agents can perceive their environment, and respond in a timely fashion. Agents perceive the learning environment through the requesters (learners or students) and service providers (institutions). On the other hand, social means that agents are capable of interacting and communicating with other agents and humans. Agents in our architecture from social context satisfy these features in interaction with requesters and in communication with each other. In this section we will present some details of three titles: MAS-based E-learning, Agent Architecture, and Agents Formalisation.

A. MAS-based E-learning

The primary concern of the MAS-based e-learning environment is the interaction of agents themselves and their relationships with their environment. These relationships are created in order to introduce dynamic e-learning environments. Controlling and organising agents' behaviours are another feature in producing a dynamic e-learning environment. The multi-agent platform in Figure 5 (Lee, 2006) plays a major role in relationships between regional grids themselves and between components within each regional grid. The main role of multi-agent systems in regional grids is to build a dynamic, intelligent and collaborative environment.

The multi-agent structure used here is hierarchical (i.e. layered). This approach is ideal for solving large-scale, complex problems. Hierarchical structures have been adopted and studied in many fields of research including scientific computing and business processing. The basic idea of hierarchical structure is that a complex system can be divided into subsystems; the overall behaviour of the system is figured out by its subsystems which perform sub-functions (Simon, 1960). Hierarchical layered MAS has three layers, the upper, middle and bottom levels, as shown in Figure5.

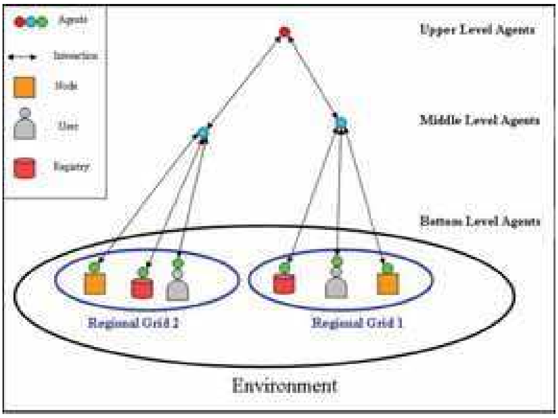

Fig. 5. Hierarchical Agent Organisation of MAS-based E-learning

Agents need to get behavioural instructions, whether from their higher level agents in the hierarchical structure or from other environment components (e.g. learners), and get support from agents in lower-level agents to perform their tasks. In our model, RDADeLE corresponds to the upper level agents (administrative agents). These agents are responsible for controlling and managing other agents at lower levels, who in turn are responsible for regional grids communicating with each other. Regional grids correspond to middle level agents, which are responsible for supporting activities within regional grids. Regional grid components correspond to bottom level agents which are considered to be sensors of the environment. The following table presents Agents Category:

TABLE 1
AGENT CATAGORY

Agent Name	Agent Category	Agent Level
Sensor	Sensor	Bottom Level
Regional	Regional	Middle Level
Administrative	Administrative	Top Level

B. Agent Architecture

Agent architecture is essentially a map of the internals of an agent. Agent architecture includes agent data structure, the operations that may be performed on these data structures, and control flow between these data structures. There are many agents' architectures which have been adopted in many applications. These architectures include Logic-Based architecture, Reactive architecture, Belief-Desire-Intention architecture (BDI), Layered architecture, and Deliberative architecture. Subsumption architecture is arguably the best-known reactive agent architecture in which agent decision making is achieved through the interaction of a number of behaviours. Most of the agents in the e-learning environment are subsumption agents. This type of agents is behaviour-based architecture which decomposes complicated intelligent behaviours into many simple behaviours, which in turn organised into layers (Weiss, 1999).

There are two models of agents which could be compared. The first one is that there is only one type of layered architecture which substitutes to do different functions. This type of agent is suitable for homogenous systems. The other type of agent is behaviour-based agents. In our architecture, we intend to adopt behaviour-based agents which are capable of achieve intelligent behaviour. This type of agent suitable for heterogeneous systems.

C. Agents Formalisation

Based on what we have mentioned, there are three types of agents. The first one is the administrative agents, the second one is the regional agents, and the third one is the sensors agents.

The administrative agents are described as a tuple as follows:

$$AA = < cr, e, a, c > \tag{1}$$

where:

- cr is a control request.
- e is an entity.
- a is an action.
- c is a constraint (policy).

Administrative agent: This kind of agent works as an information service for the system. It collects and preserves information about the system which includes registered regional grids, registered sensors (learners or users). Also it is responsible for authentication and registering of regional grids. This information helps in searching for grid services (learning objects). The agent is described as <cr, e, a> where cr: is control request, e: entity, and a: action. The control request (cr) includes creating and terminating other agents (regional agents and sensor agents). The entity (e) is either regional agents or sensor agents. Actions (a) which are performed include authentication services, registering and deregistering regional grids, creating grid model.

The second type is the regional agents which again can be described as a tuple as follows:

$$RA = < Rg, l, s, a > \tag{2}$$

where:

- Rg is a regional grid.
- l is a learning object.
- s is a service.
- a is an action.

Regional agent: This agent type works on behalf of regional grid. Grid services through this agent are presented to help learners to search for desired learning objects. According the constraints the desired learning objects are delivered the learners. The agent is described as <rg, l, s, a, c> where rg: is regional grid, l: learning objects, s: service (grid), a: action, and c: is constraint. The regional grid (rg) is a unique name and the learning object is the description of the desired learning object. The service (s) here is a special agent represents a grid service which is published in regional registry (yellow pages) to be discovered in order to retrieve desired learning object (l). Some actions are performed by this type of agent with cooperation with other agents. Those actions include searching for desired learning objects, updating regional grid data by authorised staff, and applying constraints to deliver learning objects.

The third type is the sensors agents which can also be described as a tuple as follows:

$$SA =< e, r, st, p > \quad (3)$$

where:

- e is an entity.
- r is a request.
- st is a state.
- p is a perception.

Sensor agent: This agent works on behalf of learners (users). The agent is described as <e, r, st, p> where e: is entity which is here a learner, r: is request from learner which include searching for learning objects. st: is state of learner which include some properties like unique name, number, institution belongs to, and regional grid belongs to. p: is perception.

In our model, the highest level agents are responsible for monitoring the environment. This means that the primary role of these agents is controlling both the behaviours of a whole environment and the data passing between regional grids according to assigned constrains. The middle level agents are intermediate ones which are responsible for controlling the behaviour of a regional zone that includes a number of low-level agents.
In a regional grid, possible requests can be represented as a set

$$R = \{r0, r1, r2,\ldots\} \quad (4)$$

For each user request, there exists constraints which forms a set of constrains:

$$C = \{c0, c1, c2,\ldots\} \quad (5)$$

Once the agent perceives the request, it executes an action. Agent capabilities can be represented by a set of actions:

$$A = \{a0, a1, a2,\ldots\} \quad (6)$$

This process of agent action generation can be represented as a function:

$$action : R \times C \rightarrow A \text{ (or } a_n = action\ (r_n, c_n)) \quad (7)$$

5. Implementation

The implementation is performed under the JADE platform to produce required agents. These agents collaborate with each other and collaborate with data grid which is considered as the infrastructure of the RDADeLE.

A. The Platform

JADE is the middleware developed by TILAB[1] for the development of distributed multi-agent applications. Both the intelligence, the initiative, the information, the resources and the control can be fully distributed on mobile terminals as well as on computers in the fixed network (Bellifemine, Caire, Poggi, & Rimassa, 2003). This middleware has some features within its platform. These features include heterogeneous entities communication, security, and Interoperability with other agents. JADE is an ideal platform to implement our model in order to present concepts and objectives of our research. In (Chmiel, Gawinecki, Kaczmarek, Szymczak, & Paprzycki, 2005), authors have proven some features of JADE platform. These features include efficiency, effective, and scalability. The features are limited by standard limitations of Java programming language and other factors which include processor speed, amount of available memory and speed of network connection. Experiments with thousands of agents and thousands of ACL messages had been implemented effectively. Hence, these features of JADE platform are needed in our model in order to present concepts and objectives of our research, and we see this platform is suitable to implement part of our model.

B. Registries and Multi-Agent Systems

As we have mentioned before in this paper, we have adopted the second approach of registries distribution. This approach is based on assigning one regional registry for each regional grid. In this approach there is no central registry with which to discover learning objects (LOs). Instead, regional registries are used to discover regional LOs. In this approach, all regional registries should be connected to each other in order to discover all LOs across the global grid. Agents in MAS update these registries and help to communicate between registries. However, this approach provides information service in order to handle the search process more efficient. This means that the search process will search for required information only in the registries which have it. This is accomplished by using information service the Administrative agent provides in the system. A broker middleware is responsible for connection between regional registries and universal Description, Discovery and Integration (UDDI) in order.

C. Implementation Scenarios

The implementation is limited to the role of agents in our architecture. As we mentioned earlier, the following includes three types of agents and their roles in the implementation:

1. Sensor (user): This agent represents a learner or a user who sends a query. The query includes LOs, universities or institution connected to the global grid, and a degree (major).

---

[1] Telecom Italia Lab is the R&D branch of the Telecom Italia Group and is responsible for promoting technological innovation by scouting new technologies, carrying out and assessing feasibility studies, and developing prototypes and emulators of new services and products.Telecom Italia has conceived and developed JADE, and originated the Open Source Community in February 2000.

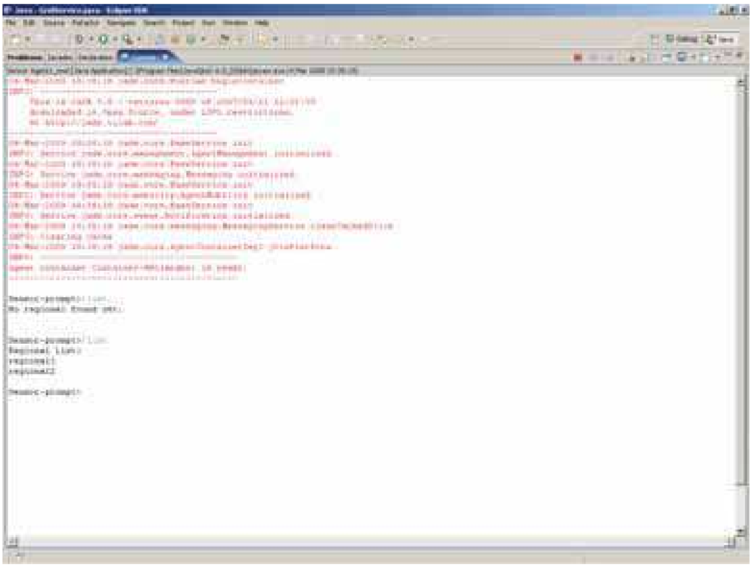

Fig. 6. Screen shot of Sensor agent.

2. Regional: This agent represents a regional grid. The role of this type of agents is to connect LOs registry of a particular regional grid to the global grid in order to make it available to be queried. Regional grids need to be registered using Administrative to be authenticated and connected to the global grid.

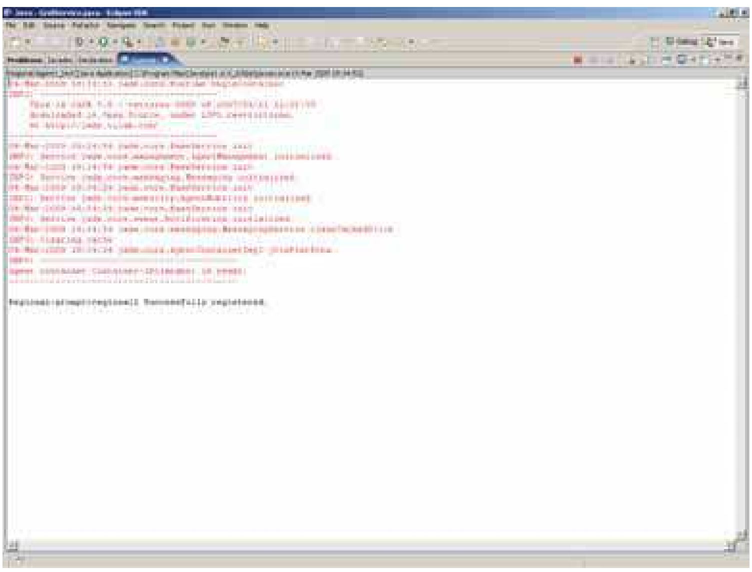

Fig. 7. Screen shot of regional agent.

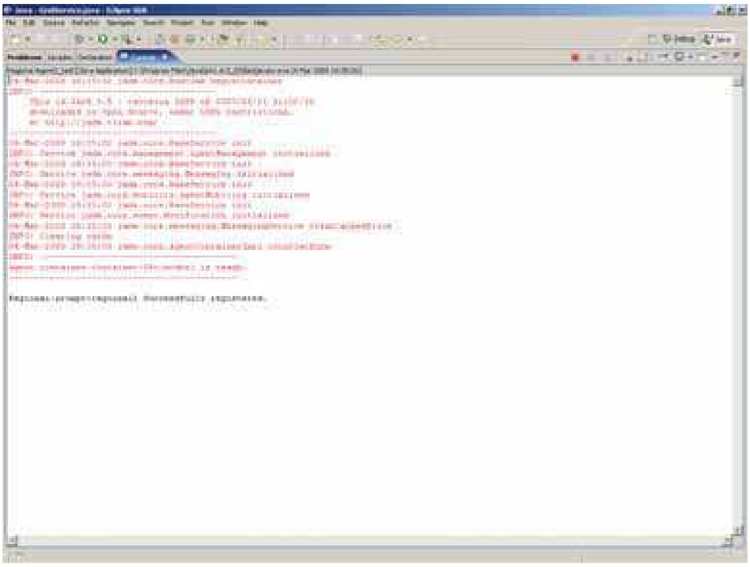

Fig. 8. Screen shot of another regional agent.

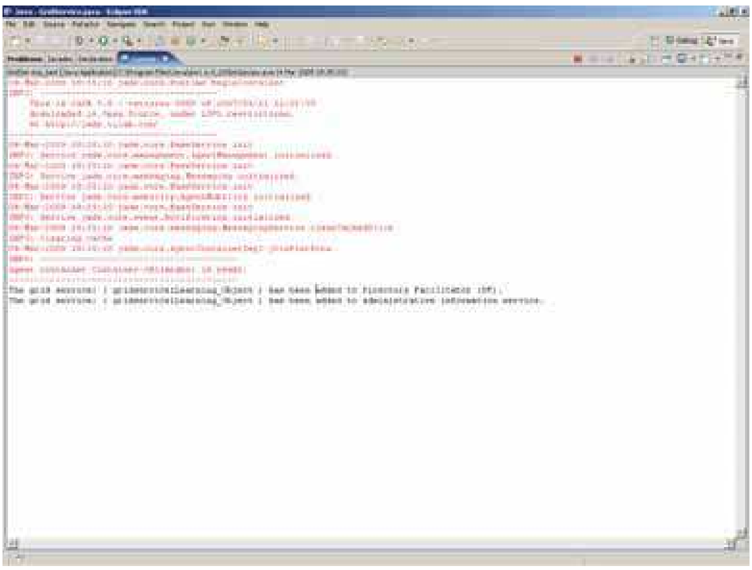

Fig. 9. Screen shot of registration of grid service.

3. Administrative: This type of agent has two roles. The first one is to supply authentication and authorisation services for different regional grids to be connected to the global grid. The second role is to function as an information service for the whole global grid to ease the search process and make it more efficient.

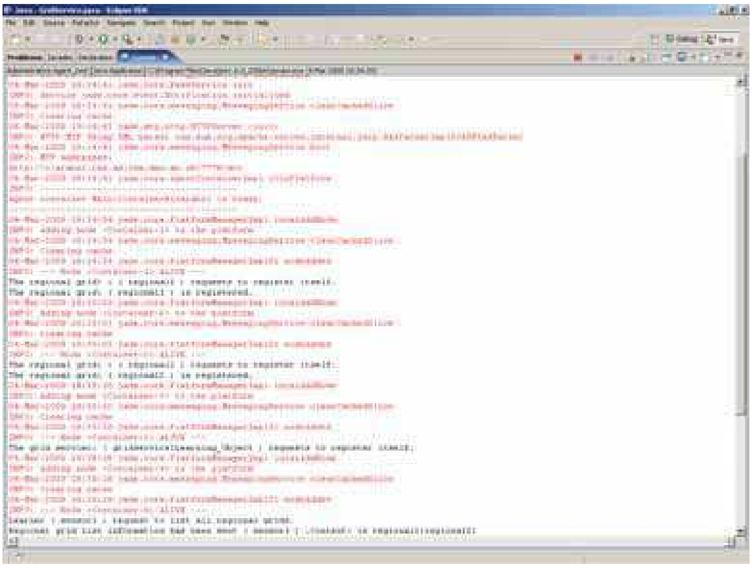

Fig. 10. Screen shot of Administrative agent.

The cooperation of the developed multi-agent system begins when Administrative agent has been initiated. First, a regional agent is initiated to be authenticated and authorised using initiated Administrative agent in order to connect regional grid to the global grid. This means that a particular regional grid has been connected to the global grid which implies that registry of that regional grid has been connected to the whole system in order to be available for learners to be queried. The agent Administrative is responsible for registering and authentication of regional grids. It the same time, Administrative agent plays as information service which includes information about all connected regional grids all over global grid. Second, a learner agent is initiated to represent a learner. The learner should be authenticated by authentication and authorisation agent in order to access to (LOs) via OGSA. At this time, learner may search for LO, connected institutions, and degree. On the other hand, authorised persons (staff) should be authenticated by Administrative agent in order to update and amend regional and institution registries. According to a learner request and the portal, a query could be:

- About LOs, institutions, and degree.
- Request a LO.

In the first case, a Sensor agent send a query about a LO. The search process will be initiated to search for the required LO. The required LO is determined by a field and subject from learner. The search process looks for all LO and their locations from all registries of connected regional grid and bring back search result to the learner. The same process will occur if the query about institutions. But process will be different if the query is about degree. In this case the Administrative agent will play as information service for the whole grid. This agent will help the learner to find different years courses in all connected regional grids. The agent will ease the search by providing the learner regional grids (institutions) that have required courses. After that the learner will connect to these regional grids (registries) to obtain information about degree courses. The second case (Request a LO) will be in the future work. In this case, learner requests the desired LO in order to access related materials. This could be available via Learning Management System (LMS). Policies of a particular regional grid are applied on components of that regional grid for both cases (e.g. LOs, learners, institutions).

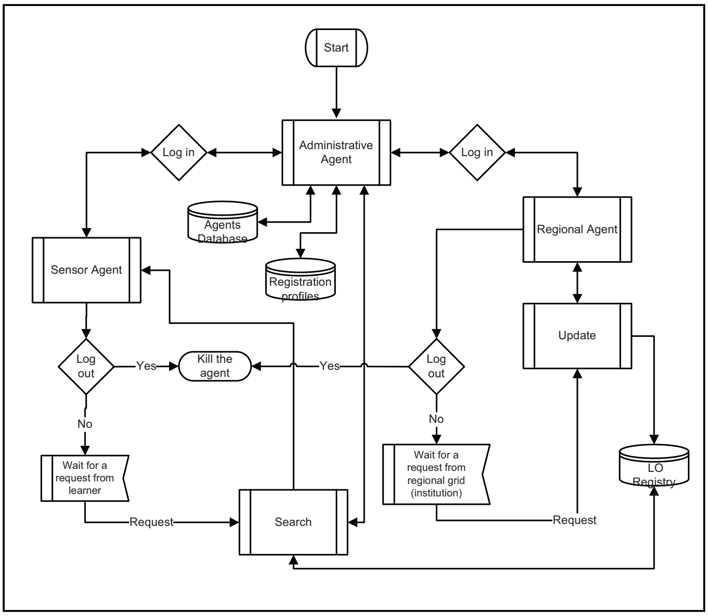

Fig. 11. illustrates the chart of processes of agents in our model.

6. Conclusion and Future Work

In this paper we have presented the role of multi-agent system in Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE). This paper, mainly describes in details the role of agents in regional data grids; the agent role is based on multi-agent systems using JADE to guarantee autonomy, interoperability, and reliability in the whole system. We began the paper by highlighting two factors: grid computing and agent technology. Secondly, we presented an overview of our architecture Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE), its components, and regional grid structure. Then, we presented agents' specifications. Finally, we introduced the implementation. Implementation performance was analysed, and some practical applications and further experiments remain to be performed in the future. The next steps of our project are incorporating OGSA-DI in order to produce grid services, ameliorating the performance of the whole project in general and refining search result in particular, developing an ontology agent to ameliorate the heterogeneity of regional grid, and developing appropriate policies for regional grids.

7.References

AlZahrani, S., Ayesh, A., & Zedan, H. (2008, May 25{27). Regionally Distributed Architecture for Dynamic e-Learning Environment (RDADeLE). *In Conference on human system interaction HSI08*. Krakow, Poland: IEEE Industrial Electronics Society.

Bellifemine, F., Caire, G., Poggi, A., & Rimassa, G. (2003, September). JADE: A white paper. EXP *in search of innovation, 4* .

Canada Secretariat, T. B. of. (2004). *Imrc - e-learning metadata sub-group: Guidelines for the government of canada apprentissage / training (gcat) metadata application profile* (Tech. Rep.). Author.

Center, S. D. S. (2005). *SRB the sdsc Storage Resource Broker* (Tech. Rep.). University of California, San Diego.

Chervenak, A. L., Deelman, E., Foster, I. T., Guy, L., Hoschek, W., Iamnitchi, A., et al. (n.d.). Giggle: a framework for constructing scalable replica location services. In *Proceedings of ieee supercomputing conference, 2002* (pp. 1-17). IEEE Computer Society.

Chmiel, K., Gawinecki, M., Kaczmarek, P., Szymczak, M., & Paprzycki, M. (2005). Efficiency of JADE agent platform. *Scientific Programming, 13* (2), 159-172.

Educational Communications, A. for, & Technology, B. (2000). *Connecting learning objects to instructional design theory: A definition, metaphor and taxonomy* (Tech. Rep.). Author.

Foster, I. T., Jennings, N. R., & Kesselman, C. (2004). Brain meets brawn: Why grid and agents need each other. In *Aamas* (pp. 8-15). IEEE Computer Society. Available from http://csdl.computer.org/comp/proceedings/aamas/2004/2092/01/20920008abs.htm

Guy, L., Kunszt, P., Laure, E., Stockinger, H., & Stockinger, K. (Edinburgh, Scotland, July 2002.). "*replica management in data grids*" (Tech. Rep.). Global Grid Forum Informational Document, GGF5.

Hussain, M., N. ; Khan. (27-28 Aug. 2005). Service-oriented e-learning architecture using web service-based intelligent agents. *First International Conference on Information and Communication Technologies, 2005*. ICICT 2005., 137-143.

Koper, R. (2001). *Modeling units of study from a pedagogical metamodel behind eml* (Tech. Rep.). Open University of the Netherlands.

Laird, J., & Lent, M. V. (2000). Human-level AI's killer application: Interactive computer games. *The AI Magazine, 22* (2), pp.15-25.

Lamehamedi, H., Shentu, Z., Szymanski, B. K., & Deelman, E. (2003, April 22{26). Simulation of dynamic data replication strategies in data grids. In *Proceedings of 17th international parallel and distributed processing symposium (IPDPS-2003)* (pp. 100{100). Los Alamitos, CA: IEEE Computer Society.

Lee, J. (2006). *Design collaboration as a framework for building intelligent environments*. Unpublished doctoral dissertation, University of Illinois.

McClelland, M. (2003). Metadata standards for educational resources. , 36 (11), 107{109. Available from http://dx.doi.org/10.1109/MC.2003.1244540

Nuclear Research (CERN), E. O. for. (2006). *Global grid service for lhc computing succeeds in gigabyte-per-second challenge* (Tech. Rep.). Author.

Pankratius, V., & Vossen, G. (2003, October). Towards E-learning grids: Using grid computing in electronic learning. In *Proc. ieee workshop on knowledge grid and grid intelligence, ieee/wic international conference on web intelligence / intelligent agent* *technology* (pp. 4-15). Halifax, Nova Scotia, Canada: Saint Mary's University. Available from citeseer.ist.psu.edu/pankratius03towards.html

Polsani, P. (2003, February). Use and abuse of reusable learning objects. *Journal of Digital Information, 3 (4)*. Available from http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani

Project, T. U. (2005). *Open elearning standards. how learning design can be used* (Tech. Rep.). Author.

Rodriguez, J. S., Anido-Rifon, L. E., & Iglesias, M. J. F. (2003). How can the web services paradigm improve the E-learning? In *The 3rd ieee international conference on advanced learning technologies, 2003. proceedings. icalt '03* (p. 479). IEEE Computer Society. Available from http://csdl.computer.org/comp/proceedings/icalt/2003/1967/00/19670479abs.htm

Russell, S., & Norvig, P. (1995). *Artificial intelligence: A modern approach*. Prentice Hall.

Simon, H. A. (1960). *The new science of management decision*. New York: Harper and Brothers.

Sousa, B. B. de, Silva, F. J. da Silva e, Teixeira, M. M., & Filho, G. C. (2006). Magcat: An agent-based metadata service for data grids. In *Ccgrid* (p. 6). IEEE Computer Society. Available from http://doi.ieeecomputersociety.org/10.1109/CCGRID.2006.153

Stathis, K., Bruijn, O. de, & Macedo, S. (2002). Living memory: agent-based information management for connected local communities. *Interacting with Computers, 14* (6), pp. 663{688. Available from http://dx.doi.org/10.1016/S0953-5438(02)00014-0

Steinacker, A., Ghavam, A., & Steinmetz, R. (2001, January). Standards: Metadata standards for Web-based resources. *IEEE MultiMedia*, 8 (1), 70-76. Available from http://dlib.computer.org/mu/books/mu2001/pdf/u1070.pdf

Team, T. O.-D. P. (2007). *The ogsa-dai project* (Tech. Rep.). The OGSA-DAI Project.

Thompson, C. W. (2004). Agents, grids, and middleware. *IEEE Internet Computing, 8* (5), 97-99. Available from http://doi.ieeecomputersociety.org/10.1109/MIC.2004.32

Weiss, G. (Ed.). (1999). *Multiagent systems - A modern approach to distributed artificial intelligence. Cambridge, Massachussetts*: The MIT Press.

Yang, C.-T., & Ho, H.-C. (2005, 29 March-1 April). A shareable e-learning platform using data grid technology. In *The 2005 ieee international conference on e-technology, e-commerce and e-service, 2005. eee '05. proceedings.* (pp. 592-595). IEEE Computer Society.

Hai Jin; Li Qi, "ChinaGrid and its impact to science and education in China," *Collaborative Computing: Networking, Applications and Worksharing, 2005 International Conference on*, vol., no., pp.9 pp.-, 0-0 0 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1651208&isnumber=34623

Wang Li; Wang Cong; Long Hao; Di Rui-Hua, "An Adaptive MAS-Based Data Acquisition Model in ESESGrid," *ChinaGrid Annual Conference, 2008*. ChinaGrid '08. The Third , vol., no., pp.218-222, 20-22 Aug. 2008.

E-learning

Edited by Maja Jakobovic

ISBN 978-953-7619-95-4
Hard cover, 312 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

E-Learning is a vast and complex research topic that poses many challenges in every aspect: educational and pedagogical strategies and techniques and the tools for achieving them; usability, accessibility and user interface design; knowledge sharing and collaborative environments; technologies, architectures, and protocols; user activity monitoring, assessment and evaluation; experiences, case studies and more. This book's authors come from all over the world; their ideas, studies, findings and experiences are a valuable contribution to enriching our knowledge in the field of eLearning. The book is divided into three sections. The first covers architectures and environments for eLearning, while the second part presents research on user interaction and technologies for building usable eLearning environments, which are the basis for realizing educational and pedagogical aims, and the final last part illustrates applications, laboratories, and experiences.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821