VECIMS 2007 – IEEE International Conference on
Virtual Environments, Human-Computer Interfaces, and Measurement Systems
Ostuni – Italy , 25-27 June 2007

Distributed Database for Environmental Data Integration

A. Amato[1], V. Di Lecce[2], and V. Piuri [3]
[1] II Engineering Faculty of Politecnico di Bari - Italy
[2] DIASS, Politecnico di Bari, Italy
[3]Dept Information Technologies, University of Milan, Italy
[e-mail: a.amato@poliba.it, dilecce@poliba.it (corresponding author), piuri@dti.unimi.it]

Abstract – ***An information system supporting environmental applications must be reliable, scalable and able to acquire and integrate data from a lot of monitoring stations distributed in different places. This paper proposes a system integrating the data acquired by a distributed network of sensors for air quality monitoring. The monitoring system is based on a well-tested Multi Agent System architecture based on functions layering. Aim of this paper is to highlight the behavior of the MAS while it is integrating data from multiple information sources and present an innovative method for web based information source integration.***

I. INTRODUCTION

The air quality monitoring systems are increasing in number and complexity all over the world, due to an expanding consciousness about the value of a high-quality environment and the specific Laws and Acts. The interest and the attention devoted to the environmental thematic, monitoring and controlling activities about air quality, are, in fact, growing quickly.

Many Government Agencies for environmental protection (i.e., the Italian ANPA) are imposing implementation of environmental monitoring stations, located in a wide geographic area, in order to:

– supply significant real data;
– document the air quality in time;
– characterize the causes determining the atmospheric pollution;
– foresee the biological effects due to long term pollutant exposure

Each monitoring station can be equipped with different kinds of sensors each one measuring different environmental parameters. Typically the acquired data are stored in a local database characterized by a schema related to the specific sensors. Nowadays, many monitoring stations are equipped with a web-based interface. This feature has led to interesting application as the one in [1]

These systems are typically based on many distributed processing units, devoted to acquire and store data from sensors [2]. The high-level processing and analysis tasks are often centralized in a single unit. In classical approach, each monitoring station sends data to the central unit that carries out information parameters (such as the trend curve of a specific substance or pollution values distributed in a particular geographic area). These parameters are necessary to analyze and validate other values acquired by instrumentations [3-4].

A natural evolution of this approach is to minimize data transfer, using the distributed artificial intelligence, also based on M. Minsky intuitions. Multi Agent Systems (MAS) represent one of the most promising approaches for creating open information systems using their ability to handle multi-agent coordination protocols to dynamically self-organize themselves when their problems and constituent agents change [5].

This work shows a Multi Agent System architecture that can be seen as an Open System suitable for environmental applications based on data integration. Databases integration is an active research field and an open question. There are many different approaches to this problem. There are some software houses trying to obtain at least integration among some Database Management Systems (for example [6]). This kind of solution is often used by companies to leverage their existing database investments. A more critical task in data management is extending database integration technologies to include a vastly greater range of data sources and data structures. The main approaches to this problem are: collection types, schema mapping and partial information [7]. In this work two different database integration methods are implemented. The first method is the schema mapping and it is used to integrate databases for classic monitoring stations. The second method is based on intelligent agent technology and it is used to integrate databases for monitoring stations equipped with web interface.

In this work, schema mappings are specified using high-level declarative formalisms that describe the correspondences between different schemata at a logical level following the approach proposed in [8].

The proposed system is able to validate and foresee data collected by monitoring stations. Using this system, user sees disparate computing sources and databases in a uniform manner, so that these resources can be accessed remotely without a priori knowledge about used data structures and of the possible queries. This gives a great and innovative methodological perspective in the design of distributed systems for environmental applications. A layered approach

* *This work is partially supported by the "New. Ton." Project, co-financed by the ERDF through the INTERREG IIIB ARCHIMED Programme A.1.180 (http://www.newton.interreg.net)*

is proposed for realizing an Open System able to host different type of agents. In this way the system is flexible, reliable and able to allow for the continuity of operation while the environment is dynamically changing.

This paper is organized as follows: section II briefly shows related works while section III describes the case study and the proposed MAS structure. Section IV gives a schematic overview of inter-agents communications. Section V presents the proposed approach to database integration and in section VI there are the conclusions.

II. RELATED WORKS

Due to the flexible and dynamic characters of intelligent agents, they are being used widely as an interface system between user and WWW for whatever application.

An example of this kind of applications could be an expert system developed for the user assistance to a better understanding of scientific data retrieved from environmental monitoring systems (these collected data are actually translated by expert chemists or biologists).

In practical application, each agent is defined as a software entity, that is capable of flexible autonomous action in order to meet its design objective [9]. A multi agent system can be defined as an organization composed of autonomous and proactive agents that interact with each other to achieve common or private goals [10-11].

According with Qiao and Zhu [12], in conceptual schemes of each agent it is possible to recognize five components:

– *Perception*, a channel for an agent to receive information from the external world.
– *Effector*, an interface for an agent in order to modify or influence the state of MAS.
– *Communication*, a mechanism for an agent to exchange communication with other members of the agent society.
– *Objectives*, list of roles that an agent can play

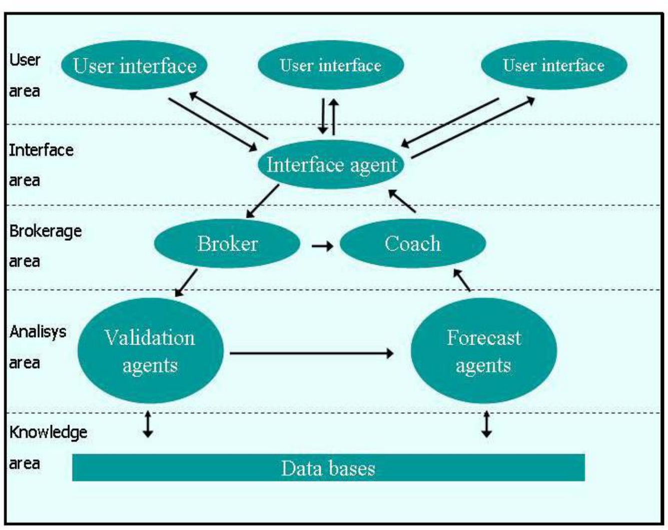

Fig. 1. Dataflow of Multi Agent System architecture for environmental monitoring systems

– *Knowledge processor*, a knowledge base system that stores and processes the necessary knowledge for an agent so that this one can play the role the MAS has assigned to it.

To create a structured, flexible, and scalable MAS we can layer different functions and components needed to solve the problem [13]. A good abstraction can be achieved and a large class of different problems solved by using a layered approach

Each layer represents a single functionality in order to minimize the interlayer communications and localize (by one or more agents) the activity. The number of agents performing functions in each layer is defined according to the goals that layer must satisfy.

III. CASE STUDY AND PROPOSED MAS ARCHITECTURE

This work deals with the analysis of environmental data, recorded in several monitoring stations installed in Taranto, Italy. Basic idea is that users (expert and not expert, human and non human) can formulate a query to the system to obtain environmental information. For this reason and according to the previous chapter the MAS is arranged in five layers: *User Area, Interface Area, Brokerage Area, Analysis Area, and Knowledge Base System Area* (Fig. 1). The proposed MAS was heavy tested in many previous works [2, 13, 21, 22] giving encouraging results.

Users can be both human and different external systems requiring access to the MAS services on the WWW. In *User Area* there are two different kinds of interface: video masks, if a human wants to interact with the MAS; communication ports, if another external MAS requires interactions.

The main role of *Interface Area* is to translate messages from external world (with its own ontology) to the MAS intermediating messages typically based on an ACL (Agent Communication Language) [14-16]. Both ontologies are characterized by their own cardinality (n_{IO}=internal cardinality, n_{EO}=external cardinality) and their own interpretation of the adopted terms. So, during a communication, the following cases can occur:

– $n_{IO} < n_{EO}$, not all required services can be satisfied by the MAS, because the message is not completely understood;
– $n_{IO} > n_{EO}$, not all services of MAS can be used by the requestor;
– $n_{IO} = n_{EO}$, there is a perfect correspondence for all the services offered by the MAS.

In *Brokerage Area* there are two different agent types: *Broker* and *Coach*. Other agents do not communicate directly among them, but they can do it thanks to an intermediary, called *Broker* [17-18].

The *Broker* pays attention to messages sent by the *Interface agent*. Then, it must decide which agent to send the message to, analyzing a *local database* where services offered by MAS are stored. If more agents can satisfy this goal, it sends the message to all agents able to satisfy it and informs the *Coach* about sent messages through a Bus.

The *Coach Agent* [19] is necessary when competitive agents are in the environment. It chooses among different agent answers to give the best result.

Analysis Area is the application core system. The agents of this area allow for satisfying the MAS computational goals. This is possible through the use of techniques of cooperation and/or competition among the agents. Agents manipulate and/or collect information from many sources. The information agents would have to be endowed with the capabilities of knowing where to look/try, how to find the information and how to collect it. In the application to the Environmental monitoring, the Analysis Area contains two agent sub-classes: the forecast and the validation agents.

Forecast agents have their own knowledge, based on a forecasting model, known in literature, and characterizing them. Each *Forecast agent* applies its analysis method, starting from the raw data previously acquired by the sensors. These agents extract their trend curve. The best response is selected by the *Coach* in the Brokerage area. The *Coach* records the differences among the measured value and the trend of every *Forecasting* agent, updating a table of reliability. This kind of communication is an exchange of information, and the knowledge of the environment increases. In this case, reactive decisions become cooperative [18, 20]. The goal of *Validation agents* is to certify the data. The *Forecast agents* might be "*myopic*" in this approach since they see only a working dataset. For this reason a temporal validation, followed by a spatial one, is performed. This justifies the presence of two agents, *Temporal-Valid* and *Spatial-Valid* agents, in the system. The last two agents are collaborative, because the unique goal is the data validation that is divided in two tasks. *Temporal-Valid* agent compares raw data with estimated data (that have been extracted using one of the previously explained *forecasting agents*). If the difference is contained within a tolerance range, data are defined as valid. An expert (e.g., a chemist) defines the tolerance range. Another agent checks non-valid data: the *Spatial-Valid* tries to validate data, using the knowledge of the geographic position in which sensors are placed.

In Knowledge Base System Area there are all the system databases that agents use to achieve their goals, or that they write to increase their knowledge.

IV. AGENT STRUCTURE AND COMMUNICATION

Each agent of the MAS for environmental applications will be structured (fig. 2) to include: local data storage system (agent knowledge), processing unit (accomplishment of agent's goal) and control unit (interact by making reciprocal commitments and carry out task data), buffer and interface (management of communication protocol).

The communication architecture among agents in the MAS must be structured, open, and scalable. To achieve this goal we adopt the Bus structure presented in Fig. 2. This implementation allows us to add and/or carry off agents without modifying the overall structure. A similar approach was used in an environmental application shown in [21-22].

Communication and interaction are performed according to the scheme presented in Fig 3. The UML diagram describes the sequence of activations and message exchanges among the various agents. According with the well-known computer bus architecture the first step (easy in term of synchronization) is to negotiate the issues sequentially.

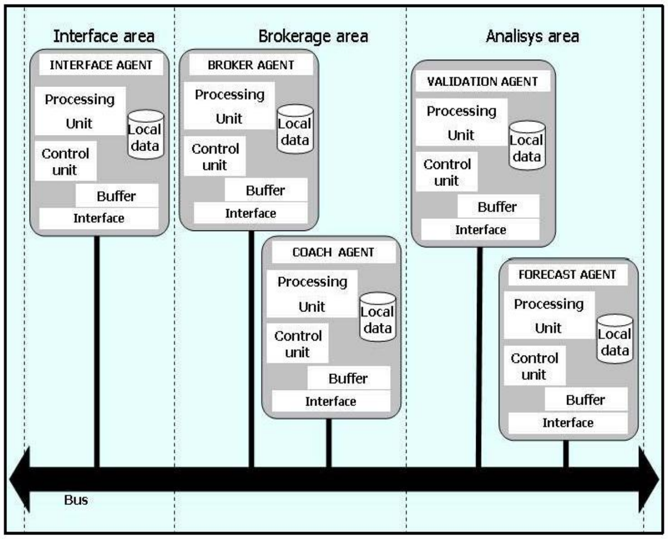

Fig. 2. Detail about BUS model: three **layers** are present. The user area is external to the MAS and the knowledge area is distributed and represented by 'Local Data'.

Although step-by-step negotiation minimizes the complexity of the negotiation procedure, an important question arising from that is in what order the agents must bargain.

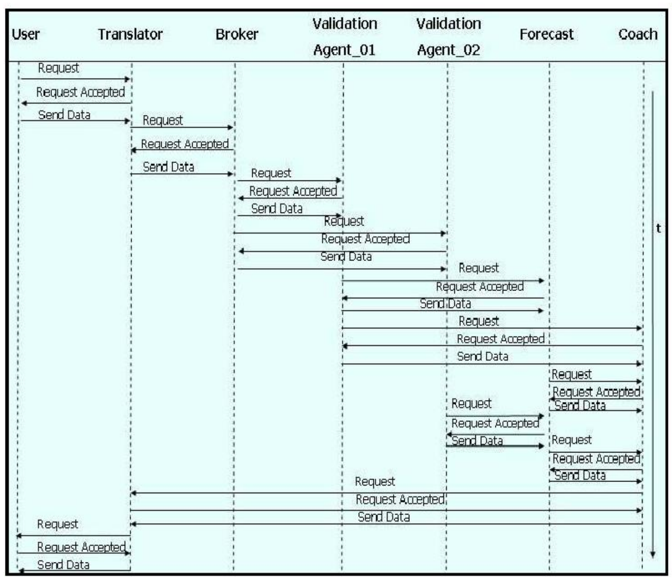

Fig. 3. UML chart of a typical MAS communication

V. DATABASE INTEGRATION

A critical task carried out by proposed system is the "spatial validation" of data acquired by sensors. As previously said, when a sensor acquires an outlier data, it starts a validation process involving all the sensors installed in its neighbor. A key point of the proposed system is the integration of many different systems in each layer of the proposed MAS structure. For example, the knowledge base area is composed of a set of different databases. The analysis area of the proposed system is able to integrate different kinds of sensor. Each sensor can be seen as an independent data source. Independent data sources are often heterogeneous even when they cover the same information domain. For example, for the sake of generality, in the proposed system different kinds of sensor measuring the same environmental parameter were integrated and each kind of sensor uses a different schema to memorize its data. In order to provide the system with a homogenous and coherent database, the authors propose two database integration methods:

1. Schema mapping method. This method is applied to classic monitoring stations. These stations are composed of many sensors interfaced with a processing unit. The latter stores the data acquired**,** according to a schema defined by device constructor, in a local database. The schema mappings are specified using high-level declarative formalisms that describe the correspondences between different schemata at a logical level following the approach proposed in [23].
2. Intelligent agents. This method is applied to the monitoring stations equipped with a web-based interface. These monitoring stations have the same equipments of the classical ones. In this approach, each monitoring station is modeled as an information source (IS). Each IS is handled by an agent called "host agent" (HA). It has the task to discover the information available in its IS and share them with the authorized mobile agents. In this work several IS provided with static and dynamic html pages have been analyzed. The HA produces a parsing of each interface html page . Parsing is divided into lexical analysis and semantic parsing. In this work lexical analysis is focused on dividing strings into components, called tokens, based on html tags. For each analyzed page the most representative words are extracted and used to enrich the knowledge base of the HA. Semantic parsing then attempts to determine the meaning of words. This task aims at measuring conceptual similarity between words. The proposed similarity criterion is based on the concept of hierarchical concept graphs. A graph is built for each word in the knowledge base by using all its synonyms. If there is no point of connection between the graphs of two words they are considered as dissimilar. Else they are similar and their distance is proportional to the number of arcs that divide the two words. A mobile agent (MA) is a lightweight software program, developed, for example, using the Java 2 Enterprise Edition (J2EE) technology, that can be created in a node of a monitoring network and then it is sent to a remote node whence it is necessary to acquire data. The schema mapping is achieved by means of the interaction between host agents and mobile agents. The information discovery process starts up when a monitoring system recognizes an outlier. Broker Agent generates as many MAs as there are the information sources in its neighbour. The MA, once gained access to the hosting IS, sets up a communication with the HA. The latter answers to MA's query retrieving data from its knowledge base according to the semantic distance described above. When MA gets the required information from the HA, it comes back to its brokering node that will analyze the retrieved data to validate the measured outlier.

VI. EXPERIMENTS AND RESULTS

A prototype of the proposed MAS was realized using the Matlab/Simulink environment. At the same time an ontology has been defined in cooperation with chemistry researchers of the *Politecnico di Bari* to describe each measured environmental parameter and its semantics.

The simplified schemes of a user's query, and the related data-flow can be synthesized as follows: User asks for a query to Interface agent that translates the query bridging the external ontology with the internal one and communicates the message to the Broker; this reacts to stimulus, sending a goal to all competitive agents, that can do it; each agent solves the problem and transfers results to Coach; eventually, the Coach evaluates whatever results and informs the User.

At present the proposed spatial validation method is under investigation. Preliminary tests, carried out using different database schemata related to different sensors have been showing good results both in terms of efficiency and effectiveness. The first method of database integration (schema mappings for monitoring node without web based interface) gives good results, but it could be considered as a "classical" solution. While the second method (schema discovering by means of interaction between agents) is a new approach proposed by the authors that have successfully applied it in medical database integration [24] and European Port Authority database integration [25].

VI. CONCLUSION

This work presents a project of database integration partly supported by European Region Development Fund, "New.ton" Project. The case study was the integration of heterogeneous distributed databases for environmental monitoring applications. The authors use a solution based on a multi agent system and proposes a new method for databases integration. The MAS architecture is based on five layers each one characterized by homogenous functions; the implementation is realized on a bus allowing to add/delete agents in the proposed MAS.

The proposed method for databases integration based on interaction among agents is giving encouraging results in the