International Congress on Environmental Modelling and Software

2nd International Congress on Environmental Modelling and Software - Osnabrück, Germany - June 2004

Jul 1st, 12:00 AM

Applying agent technology in Environmental Management Systems under real-time constraints

Ioannis N. Athanasiadis

P. A. Mitkas

Applying agent technology in Environmental Management Systems under real-time constraints

I. N. Athanasiadis and P. A. Mitkas [a,b]

[a]Informatics and Telematics Institute, Centre for Research and Technology Hellas, GR-57001 Thermi, Greece
[b]Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, GR-54124 Thessaloniki, Greece
ionathan@iti.gr

Abstract: Changes in the natural environment affect our quality of life. Thus, government, industry, and the public call for integrated environmental management systems capable of supplying all parties with validated, accurate and timely information. The 'near real-time' constraint reveals two critical problems in delivering such tasks: the low quality or absence of data, and the changing conditions over a long period. These problems are common in environmental monitoring networks and although harmless for off-line studies, they may be serious for near real-time systems.

In this work, we discuss the problem space of near real-time reporting Environmental Management Systems and present a methodology for applying agent technology this area. The proposed methodology applies powerful tools from the IT sector, such as software agents and machine learning, and identifies the potential use for solving real-world problems. An experimental agent-based prototype developed for monitoring and assessing air-quality in near real time is presented. A community of software agents is assigned to monitor and validate measurements coming from several sensors, to assess air-quality, and, finally, to deliver air quality indicators and alarms to appropriate recipients, when needed, over the web. The architecture of the developed system is presented and the deployment of a real-world test case is demonstrated.

Keywords: Agent-based systems; Environmental monitoring systems; Decision support systems

1 INTRODUCTION

Environmental monitoring networks have been established worldwide, primarily in areas with potential pollution problems, in order to observe and record the conditions of the natural environment. Through these networks, vast volumes of raw data are captured, while information systems, called Environmental Management Systems (EMS), are in charge of integrating all recorded data-streams. A typical EMS installation involves the fusion into a central database of all data sensed at distributed locations. Until lately, all recorded data were meant for environmental scientists occupied with off-line studies and post-processing activities in their effort to understand the natural phenomena involved.

However, during the last few years there has been a transition in environmental monitoring systems. The aftermath of the growing societal interest for the environment and sustainable development was the emerging need for providing environmental information to the public. The challenge for EMS is to embrace the new users in the administration, industry, and the society. Unfortunately, stakeholders still hold varying interpretations of the environmental values, thus different types of information are requested by each one. In spite of their diverse needs, all users agree on the necessity to access trustworthy information *on time*. Near real-time identification of environmental incidents affects the response of all stakeholders and the effectiveness of prevention measures.

In this paper near real-time reporting Environmental Management Systems are considered, focusing on recent developments that used software agents. The "near real time" term emphasizes that such systems are capable to deliver timely information, with respect to user- or application- imposed deadlines. In the following sections, a short review of various agent-based EMS is presented and a generic methodology for applying software agent technology to this kind of applications is detailed. Finally, an experimental agent-based prototype developed for monitoring and assessing air-quality in near real time is presented.

2 BACKGROUND

2.1 Environmental Management Systems

Environmental Management Systems (EMS) is a generic term used for describing structures that allow an organization to assess and control the environmental impact of its activities, products or services[1]. EMS vary from systems that provide a framework for monitoring and reporting on an organization's environmental performance (as the auditing schemes of ISO 14001 and EMAS), to systematic processes for assessing, managing and reducing environmental risk. Considering the quest for environmental information involving public, industry and administration, the challenge for EMS is to provide advanced, citizen-centered information services. To address such a challenge, Environmental Informatics[2], the research initiative examining the application of Information Technology in environmental research, monitoring, assessment, management and policy has emerged. Advances in the IT sector provide capable infrastructure for fusing knowledge into the every-day life of citizens, which is expected to lead to a new paradigm for the quality of life within the urban web, with citizen centered, environmental information services that will support societal sustainability while promoting personal well being [Karatzas et al., 2003].

In the aforementioned context, EMS goals are no more fettered to integrating raw data-measurements, rather is to fuse information and diffuse knowledge, in a form comprehensible by everyone, not only the environmentalists. One could say that EMS have extended their services from simple *Integration*, to *Assessment* and *Warning Services*, incorporating capabilities for decision support. Following a different pathway, EMS can be categorized based on their overall goals. EMS are called to fulfill dissimilar needs, thus system goals can be classified to the following three categories:

a. *Off-line analysis systems*. Such systems are geared towards gathering historical data in a systematic way and making them available for indepth analysis of the phenomena involved.
b. *Real-time reporting systems*. These are systems responsible for identifying and reporting the current environmental conditions. They satisfy the public need for environmental awareness and the administrative and industrial needs for precaution measures.
c. *Forecasting Systems*. In this case, the goal is to prognosticate the future conditions of the environment, either in the long or in the short term. The need to foresee and forewarn about potential environmental problems is the key for preserving nature and taking preventive actions.

Several EMS systems have been developed worldwide to realize the abovementioned goals. The evolution of EMS systems started with the off-line analysis systems, gathering information used for experimental evaluations of ecological theories. Next came the long-term forecasting systems, starting with the Climate Change Models developed in the 70's, as a response to depreciation of the environmental conditions. The last few years, public growing concern has led governments in Europe and the US to ask for real-time reporting of environmental quality. These actions are on the direction imposed by legislation (i.e. the US Clean Air Act 1990 and the European Directive on Ambient Air Quality, 1996). The European Directive 92/72/EEC aims to provide the public with information when warning and information threshold levels are exceeded. Thus, the "real-time" reporting EMS have emerged.

2.2 Software Agent Technology

This paper focuses on the design and development of EMS systems using software agent technology. Agent-Oriented Software Engineering has emerged as a novel paradigm for building software applications. The key abstraction used is that of an agent, as a software entity characterized by autonomy, reactivity, pro-activity, and social ability [Jennings, 2000]. Certain types of software agents have abilities to infer rationally and support the decision making process [Jennings et al., 1998].

Agent-based systems may rely on a single agent, but the advantages of this initiative are revealed in the case of Multi-Agent Systems, which consist of a community of co-operating agents. Several agents, structured in groups, can share perceptions and operate synergistically to achieve overall goals.

2.3 Related Work

The characteristics of agents and multi-agent systems enable them to process information and solve problems in distributed environments, as those of EMS. Thus, several agent-based EMS have been developed, in an approach to improve the performance of EMS. All these projects used agents or agent-related techniques to achieve EMS goals and supply services, such as Integration, Assessment and Warnings. In a schematic representation, (Figure 1), EMS services and goals are considered as the two axes for a unified classification of agent-based developed systems.

[1] Defi nition given by the Standards Council of Canada.
[2] International Federation for Information Processing, WG 5.11: Computers and the environment, www.environmatics.org

Agent-based EMS development is concentrated in two directions. One is the transparent integration of environmental information. Such systems are InfoSleuth [Pitts and Fowler, 2001], EDEN-IW [Felluga et al., 2003], NZDIS [Purvis et al., 2000] and Kaleidoscope [Micucci, 2002]. A common practice adopted by these projects is to use software agents for distributed information processing and propagation.

The second direction drives towards forecasting systems, which take advantage of the agent abilities for distributed problem solving, in an effort to provide warning services. These systems include FSEP [Dance et al., 2003] and DNEMO [Kalapanidas and Avouris, 2002], which realize intelligent agent features for the identification of environmental incidents in advance. Agent intelligence is implemented using case-based reasoning engines, regression trees, and neural networks.

Agent technology was adopted by all the aforementioned projects successfully, indicating that it is a promising approach to both unify distributed information and implement warning services. Obviously, there is a gap between the two clusters of applications, which comprises the real-time reporting EMS, supporting assessment services. These systems will be discussed in the following section.

3 NEAR REAL-TIME EMS

3.1 Problem formulation

Integrated EMS need to supply administration, industry and the public with validated, accurate information related to the environmental conditions. Human experts and scientists must have adequate data support in their efforts to assess environmental quality and issue alarms on time. The ‘near real-time’ constraint unfolds the most critical problems in delivering such tasks: (a) the low quality or absence of data, and (b) the changing conditions over a long period. These problems are common in environmental monitoring networks and although harmless for off-line studies, they may prove to be critical for near real-time systems.

The main objective of a near real-time EMS is to provide citizen-centred Electronic Information Services, including the following:

a. Acquisition of information from distributed locations,
b. Information fusion and preprocessing,
c. Data storage and organization,
d. Environmental assessment, and
e. Qualitative indicators circulation over the internet

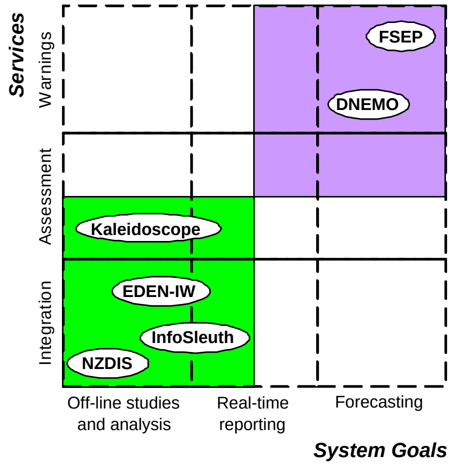

Figure 1: Classification of agent-based EMS

The overall problem that a near real-time EMS is called to solve can be summarized as follows: Let a sensor network monitoring the environmental conditions at distributed locations. An EMS is installed over this network, capturing the sensed measurements, assessing environmental quality and delivering preprocessed information to the final users of the system, over the internet.

These core activities impose the requirements for both distributed information fusion and distributed problem solving abilities. Agent success stories in both information integration and warning services need to be coupled in a common methodology.

3.2 Methodology

Advancing on the way earlier research work has dealt with EMS using Agent Technology, we propose a methodology for the development of a near real-time EMS as a multi-agent system (MAS). Our goal is to assign all tasks involved in the near real-time EMS operation to a software agent society. Through this approach, agents are considered as both **information carriers** and **decision-makers**. Agents as information carriers, act as a distributed community of data processing units, able to capture, manipulate and propagate information efficiently. Agents as decision-makers, behave as a network of problem-solvers that work together to reach solutions. Our integrated methodology provides a pathway, which defines both modes of agent operation in a MAS. An abstract view of our methodology is depicted in Figure 2.

The starting point is to identify all the appropriate resources hidden in the application domain. In-depth understanding of the related domain affects the specifications of the information flows and domain knowledge. Information flows impose

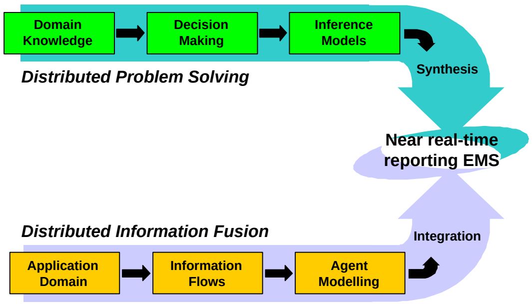

Figure 2: An abstract view of the method

how agents should manipulate data, while domain knowledge implies the decision making process incorporated into the agents. Information flows are specified through agent communication modeling, while the decision-making processes have to be transformed into agent reasoning models.

In the following section, these two procedures are further explained.

3.3 Agents as information carriers

Modeling agents as information carriers involves four steps:

Step 1. *Identify system inputs and outputs.*
Consider the interfaces between the system and both the sensors and the end-user electronic services. Assign agents to realize these interfaces acting either as data fountains or data sinks.
Step 2. *Formulate information channels.*
Detail how information flows through the system. Specify possible data transformations needed. Assign those tasks to data management agents.
Step 3. *Conceptualize agent messaging.*
Based on the two previous steps, realize inter-agent communications for smooth information propagation. Specify the semantics of the communications using ontologies.
Step 4. *Specify delivery deadlines.*
Concrete deadlines are enjoined to agent communication, in order to ensure 'on-time' delivery of information. Exit on failure strategies need to be considered too.

The outcome of the above procedure is materialized to the specifications of a multi-agent system architecture, in the form of:

$$\mathrm{MAS} = \langle A, O, I, D \rangle \quad (1)$$

where:

- $A = \{\mathrm{A}_1, \ldots, \mathrm{A}_n\}$, is a countable set of software agents, each one of which is assigned either to introduce, manipulate or export data.
- O is the domain ontology, which specifies the common vocabulary in order to represent the system environment.
- $I = \{\mathrm{I}_k = (\mathrm{A}_i, \mathrm{A}_j)/\mathrm{A}_i, \mathrm{A}_j \in A\}$, is a set of interactions between agents. These interactions show the relations in the system organization and they allow the definition of a social framework determining the information flows in the system.
- $D = \{\mathrm{D}_k, \forall\, \mathrm{I}_k \in I\}$, is a set of the delivery deadlines assigned to each agent communication.

The modeling procedure and the resulted specifications, formulated in Eq.1, define in detail the architecture and operation of a multi-agent system acting as a near real-time EMS, from an information fusion perspective. State-of-the-art methodologies for software agent modeling, as GAIA [Wooldridge et al., 2000], AUML [Odell et al., 2000], AORML [Wagner, 2003] or iSTAR [Yu, 1997] are used for defining agent roles, types, protocols, and interactions. EMS critical services, such as information acquisition, preprocessing, storage and organization are organized methodically to ensure efficient, on time electronic services to the public.

3.4 Agents as decision-makers

Agents as decision makers are employed to deliver the reasoning abilities of the MAS. Indicatively, decision-making in a real-time EMS involves either assessment services or activities to overcome data uncertainty problems. Based on the domain knowledge, agent decision-making strategies are identified through the following procedure.

Step 1. *Problem formulation and decomposition.*
Consider the overall problem at hand and try to break it down into sub-problems.
Step 2. *Construction of decision points.*
Assign specific agents to solve each sub-problem, taking under account their resources, specified by the system's architecture.
Step 3. *Decision strategy specification.*
For each sub-problem provide a strategy to solve it using the available resources. Consider that in a near real-time system the goal is to find the best solution possible in the available timeframe.
Step 4. *Realization of Inference models.*
Implement the decision strategies designed in the previous step as inference models of the respective agents. Inference agents will be embedded into agents as reasoning engines.

This procedure is highly correlated with the application under consideration. Finding an optimal decision strategy is a rather difficult task, especially when execution time is a parameter of success.

However, three distinct cases of decision-making strategies, suitable for the case discussed, can be identified:

Case 1 *Deterministic Strategies*
These are applied, when domain-specific, certain, explainable rules for decision-making exist. Such rules may encompass natural laws, logical rules or physical constrains. In such cases, these rules are incorporated as a static, confident, explainable expert system into the agents.
Case 2 *Data-driven Strategies*.
When historical datasets are available, the application of machine learning algorithms for knowledge discovery can yield interesting knowledge models. These models can be used for agent reasoning in a dynamic, inductive way. In EMS, there are large volumes of data continually recorded. When natural laws describing the monitored phenomena do not exist, or they are too complex, data-driven models, such as decision trees, case-based reasoning, or neural networks provide an option to the application developer. In this case, the procedure involves the creation of an inference model from historical data. This model is later incorporated into the agents.
Case 3 *Heuristic strategies*.
When neither of the above cases is applicable, heuristic models or 'rules of thumb' may be incorporated into agents.

Following the aforementioned procedure, and having identified the appropriate decision strategies among the three cases, decision-making required by a near-real-time EMS can be realized.

The combined methodology, presented in this section, provides an integrated pathway for developing a near real-time EMS using software agents.

4 AN EMS FOR AIR QUALITY

The methodology described in the previous section has been evaluated for the development of O_3RTAA, a near real-time reporting EMS. O_3RTAA is a multi-agent system for monitoring and assessing air-quality attributes, which uses data coming from a meteorological station. A community of software agents is assigned to monitor and validate measurements coming from several sensors, to assess air-quality, and, finally, to fire alarms to appropriate recipients, when needed, via the Internet.

Agents as information carriers undertake the following main functions of the system:

A. Data collection from field sensors.
B. Data management, including activities as data preprocessing, normalization and transformation.
C. Information propagation, which involves posting information over the internet.

Thus, system agents are organized into three groups (or layers): Contribution, Management and Distribution. Contribution Agents (CA) act as the data

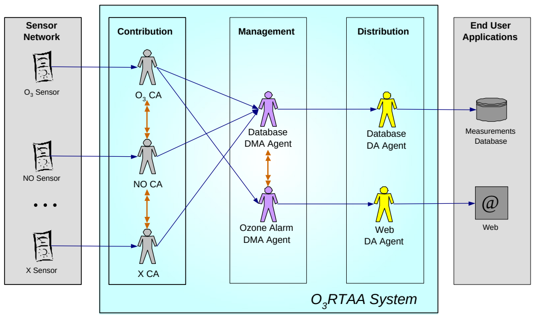

Figure 3: O_3RTAA System Architecture

fountain for the system, realizing the appropriate interfaces with the sensors. Each CA is associated with a single sensor. Data Management Agents (DMA) are responsible to fuse data coming from CAs. Each DMA produces a joint view on the sensed data in the appropriate format required by the end-user applications. In O_3RTAA two DMA agents are employed. The first is responsible for posting sensed data into a Measurements Database, for future use. The second is assigned to calculate Ozone Alarms, to be distributed over the Internet. Finally, two Distribution Agents (DA) are instantiated occupied to realize the interfaces to the end-user applications. One is in charge of the Measurements Database, while the second posts Air Quality Alarms over the Internet.

The overall O_3RTAA system architecture and the agent communication are shown in Figure 3. Intra-layer communication amplifies individual agent ability to make trustworthy decisions, in a distributed AI manner. Inter-layer communication ensures the successful transfer of data to the final destination. O_3RTAA agent messages follow a generic ontology developed using the Protégé-2000, ontology editor[3]. Agent delivery deadlines have been specified to less than a minute, while FIPA[4] compliance in agent communication ensures the proper handling of missing or erroneous message transmission. O_3RTAA agent architecture is described in detail in Athanasiadis and Mitkas [2004].

O_3RTAA agents as decision-makers are responsible for the following activities:

A. Incoming measurement validation, inspecting the quality of the data sensed.
B. Estimation of erroneous measurements, substituting the missing values.
C. Estimation of qualitative indicators, assessing the overall environmental quality.

The first two activities are left to CAs, while Alarm DMA is in charge of the third. Each CA is respon-

[3] http://protege.stanford.edu
[4] Foundation of Physical Intelligent Agents, http://www.fipa.org

sible for suppling the O_3RTAA with data coming from a particular sensor. Thus, they are assigned to validate incoming measurements and estimate the missing ones. These two activities are both realized using data-driven strategies. This comes as a consequence of the nature of these tasks, which are subject to the local conditions. Data validation and missing measurement estimation activities are both related to the changing conditions over a long period. Deterministic strategies are unsuitable, while vast volumes of historical data are available. Thus, two types of data-driven reasoning engines are incorporated in each CA. One is the Validation Reasoning Engine and the second is the Estimation Reasoning Engine. Details on the specification on these Engines are given in Athanasiadis et al. [2003a, b].

Estimating air quality indicators involves the application of specific thresholds, imposed by legislation. These thresholds represent a deterministic decision-making strategy. Thus, the Alarm DMA implemented an ozone alert system, by the use of a *Deterministic Inference* Reasoning Engine.

The O_3RTAA system has been successfully installed as a pilot case at the Mediterranean Centre for Environmental Studies Foundation (CEAM), Valencia, Spain, in collaboration with IDI-EIKON, Valencia, Spain.

5 CONCLUSIONS

In this paper, we presented a methodology for developing near real-time reporting EMS using software agents, and evaluated it for the development of the O_3RTAA prototype. The benefits of this methodology are twofold: First, it can handle data uncertainty problems through the employment of a distributed problem solving approach, employing agents that collaborate and synergistically make decisions. Secondly, it employs a distributed information processing approach, using software agents for data fusion, in order to provide at near real time, trustworthy information over the web.

REFERENCES

Athanasiadis, I. N. and P. A. Mitkas. An agent-based intelligent environmental monitoring system. *Management of Environmental Quality*, 15(3):238–249, 2004.

Athanasiadis, I. N., V. G. Kaburlasos, P. A. Mitkas, and V. Petridis. Applying machine learning techniques on air quality data for real-time decision support. In *Information Technologies in Environmental Engineering*, Gdansk, Poland, June 2003a. ICSC-NAISO Publishers.

Athanasiadis, I. N., P. A. Mitkas, G. B. Laleci, and Y. Kabak. Embedding data-driven decision strategies on software agents: The case of a multi-agent system for monitoring air-quality indexes. In *Concurrent Engineering: The Vision for the Future Generation in Research and Applications*, Madeira, Portugal, July 2003b. Balkema Publishers.

Dance, S., M. Gorman, L. Padgham, and M. Winikoff. An evolving multi agent system for meteorological alerts. In *Proc. of the second international joint conference on Autonomous agents and multiagent systems*, pages 966–967. ACM Press, 2003.

Felluga, B. et al. Environmental data exchange for inland waters using independed software agents. Report 20549 EN, Institute for Environment and Sustainability, European Joint Research Centre, Ispra, Italy, April 2003.

Jennings, N. R., K. Sycara, and M. J. Wooldridge. A roadmap of agent research and development. In *Autonomous Agents and Multi-Agent Systems 1*, pages 7–38, Boston, 1998. Kluwer Academic Publishers.

Jennings, N. R. On agent-based software engineering. *Artificial Intelligence*, 117:277–296, 2000.

Kalapanidas, E. and N. Avouris. Air quality management using a multi-agent system. *International Journal of Computer Aided Civil and Infrastructure Engineering*, 17:119–130, 2002.

Karatzas, K., A. Bassoukos, A. Kaprara, A. Masouras, T. Slini, J. Papaioannou, and N. Moussiopoulos. Environmental informatics and urban air quality management systems. In *5th International Symposium on Environmental Software Systems (ISESS 2003)*, volume 5 of *Environmental Software Systems*, pages 134–146, Semmering, Austria, 2003. IFIP.

Micucci, D. Exploiting the kaleidoscope architecture in an industrial environmental monitoring system with heterogeneous devices and a knowledge-based supervisor. In *Proc. of the 14th international conference on Software engineering and knowledge engineering*, pages 685–688. ACM Press, 2002.

Odell, J., H. van Dyke Parunak, and B. Bauer. Extending UML for agents. In *Proc. of the 2nd Int. Workshop on Agent-Oriented Information Systems*, Berlin, Germany, 2000. iCue Publishing.

Pitts, G. and J. Fowler. InfoSleuth: An emerging technology for sharing distributed environmental information. In *Information Systems and the Environment*, pages 159–172. National Academy Press, 2001.

Purvis, M., S. Cranefi eld, and M. Nowostawski. A distributed architecture for environmental information systems. In *Environmental Software Systems-Environmental Information and Decision Support*, pages 49–56. Kluwer Academic Publishers, 2000.

Wagner, G. The Agent–Object–Relationship metamodel: Towards a unifi ed conceptual view of state and behavior. *Information Systems*, 28(5):475–504, 2003.

Wooldridge, M., N. R. Jennings, and D. Kinny. The Gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, 3 (3):285–312, 2000.

Yu, E. Towards modelling and reasoning support for early-phase requirements engineering. In *Proc. of the 3rd IEEE Int. Symp. on Requirements Engineering*, Washington, USA, 1997. IEEE.