Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 36 (2009) 2603–2614
Expert Systems with Applications
www.elsevier.com/locate/eswa

Modeling and implementing an agent-based environmental health impact decision support system

Marina V. Sokolova [a,b], Antonio Fernández-Caballero [a,c,*]

[a] Instituto de Investigación en Informática de Albacete (I3A), Universidad de Castilla-La Mancha, 02071 Albacete, Spain
[b] Kursk State Technical University, Kursk, ul.50 Let Oktyabrya 305040, Russia
[c] Departamento de Sistemas Informáticos, Escuela Politécnica Superior de Albacete, Universidad de Castilla-La Mancha, 02071 Albacete, Spain

Abstract

This paper presents an approach to the creation of an agent-based system for the assessment of environmental impact upon human health. As indicators of the environmental impact water pollution, indexes of traffic and industrial activity, wastes and solar radiation are assumed. And as human health indicator morbidity is taken. All the data comprise multiple heterogeneous data repositories. The system is logically and functionally divided into three layers, solving the tasks of information fusion, pattern discovery and decision support making, respectively. The outcomes of the system design phase under Prometheus methodology and the complete characteristics of the agents forming the proposal are discussed. The discovered patterns are used as a foundation for real-time decision making, which is of great importance for adequate and effective management by responsible governmental authorities.

Keywords: Environmental impact; Decision support system; Data mining; Intelligent agents

1. Introduction

The necessity of the research described in this paper is stated by the fact that environmental pollution, as a result of energy production, transportation, industry, or lifestyle choices, adversely affects health. The term "environmental pollution" includes factors such as ambient and indoor air pollution, water pollution, inadequate waste management, pesticides, noise and radiation. In addition, people, due to their life styles and the aggressive ecological impact, usually face a deteriorated environment that affects their health and provokes degradation within a population. This is demonstrated by an increasing number of endogenous diseases (such as birth defects, chromosome diseases, etc.) and some classes of exogenous diseases (diseases of the skin and subcutaneous tissue, endocrine and metabolic diseases, neoplasm, and some others). This paper considers the necessity to assist responsible environmental health authorities in decision making. Because there are many approaches to decision making and because of the wide range of domains in which decisions are made, the concept of decision support system (DSS) is very broad. A DSS can take many different forms. In general, it is accepted that a DSS is a computerized system for helping make decisions. A decision is a choice between alternatives based on estimates of the values of those alternatives.

Unfortunately, decision making in environmental health is not a trivial task. Indeed, retrospective environmental data suffer from noise, gaps and outliers, as well as from measurement errors. Working with public health information puts on restrictions caused by the methodologies of data measurement, the standards currently in use, data availability, and so on. For example, it is known that the International Statistical Classification of Diseases and Related Health Problems (ICD) was reviewed 10 times,

* Corresponding author. Address: Departamento de Sistemas Informáticos, Escuela Politécnica Superior de Albacete, Universidad de Castilla-La Mancha, 02071 Albacete, Spain. Tel.: +34 967 599200; fax: +34 967 599224.

E-mail addresses: marina.v.sokolova@gmail.com (M.V. Sokolova), caballer@dsi.uclm.es (A. Fernández-Caballero).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.01.041

the International Classification of Functioning and Disability (ICIDH) another two times, and any local standards are also frequently reviewed (International Classification of Diseases, 2007). In recent years, the tendency to use products and energy lifecycle indicators in order to assess the ecological impact has appeared, as some difficulties in obtaining this information have been noticed. The idea was developed and fixed in ISO 14031 "Environmental Performance Evaluation" (ISO, 1999), which certifies the usage of indirect indicators.

In the last years, some proposals for intelligent and agent-based decision support systems (e.g. Kebair & Serin, 2006; Liu, Qian, & Song, 2006; Ossowski et al., 2004; Petrov & Stoyen, 2000; Urbani & Delhom, 2005) have been described. New approaches of researching intelligent decision support system (IDSS) appear following the rapid progress of agent systems and network technology. Thus, a large range of works dedicated to environment and human health implemented as multi-agent systems (MAS), which are in the center of active research for more than ten years and resulted in many successful applications, have emerged. The application of data mining (DM) techniques for environmental monitoring, medicine, social issues is also a rather common hot topic. After studying some related previous works (Section 2), the proposal for an environmental impact decision support system (Section 3) are provided. Then an improved agent-based system through modeling in Prometheus (Padgham & Winikoff, 2004) methodology is introduced in Section 4. Section 5 shows a significant example of the implementation of the obtained model using JACK Intelligent Agents (Winikoff, 2005).

2. Related works

Pollution and health data altogether form a complex system. Therefore they inherit all the necessary characteristics to be modeled with the MAS approach. These characteristics are modularity, decentralization, changeability, ill structure and weak predictability. Multi-agent approach seems to be the only technique which can help to reduce the complexity of the system by creating modular components, which solve private subtasks that constitute together the whole goal. Every agent utilizes the most effective technique for solving the subtask and does not apply the general approach, acceptable for the system in the whole, but not optimal for a concrete subtask. In case the subtasks are interconnected, agents have to communicate to each other and the system can be represented as a community of semi-autonomous and autonomous distributed intelligent entities, solving the private tasks and constituting a whole complex system.

In Athanasiadis, Mentes, Mitkas, and Mylopoulos (2005) it is reported about applying the software agents' paradigm to environmental monitoring informational system created by a three-layered MAS, which is aimed to provide measurements of meteorological information and air pollution data, to analyze them and to generate alarm signals. The situation assessment was realized by an agent-based system (Gorodetsky, Karsaeyv, & Samoilov, 2005) dedicated to the creation of a general approach to situation assessment, which can be applied for different problem areas. The authors accept the JDL model (Ly, Greenhill, Venkatesh, & Pearce, 2003) as a basis for situation awareness; the new approach to situation assessment learning is described and the structure of the MAS is presented as well. In another paper (Urbani & Delhom, 2005), the authors present the framework of a decision support system for water management in the Mediterranean islands coupling a multi-agents system with a geographic information system. The platform developed makes it possible for users to better understand the current operation of the system, to apprehend the evolution of the situation, and to simulate different scenarios according to the selected water policies and the climatic changes hypothesis. Recently, the development and experimental evaluation of an Internet-enabled multi-agent prototype called AgentStra (Li, 2007) for developing marketing strategies, competitive strategies and associated e-commerce strategies has been introduced.

On the other hand, the specialists working with environmental sciences and public health store huge volumes of relevant monitoring factual information about pollutants and human health that are thought to be analyzed. Continuous processing and maintenance of the information requires essential efforts from the practitioners and professionals not only while handling and storing data, but also when interpreting them. Actually, it seems very hard to handle all the data without using DM methods, which can autonomously dig out all the valuable knowledge that is embedded in a database without human supervision, providing a full life-cycle support of data analysis, using such techniques as clustering, classification, logical and association rule-based reasoning, and other methods, which are highly demanded for a comprehensive environmental data analysis. For instance, DM techniques for knowledge discovering and early diagnostics were utilized for early intervention in developmentally-delayed children (Chang, 2006). In Chen and Bell (2002) it is reported about the MAS named "Instrumented City Data Base Analyst", which is aimed to reveal correlations between human health and environmental stress factors (traffic activity, meteorological data and noise monitoring information) by using wide range of DM methods, including regression analysis, neural networks, ANOVA and others. In Foster, McGregor, and El-Masri (2006) the authors suggest using intelligent agents within an agent-based intelligent decision support system in the area of clinical management and research. The IDSS is aimed to improve the decision quality and be used in urgent cases. The system uses Web services to display and supervise the decision making process, which requires the other layers: solution manager service, which analyze information from Information Integrator layer and builds solutions.

As all these works have demonstrated novel and promising practical and theoretical outcomes, it seems to be important to create an agent-based decision support system (ADSS) for knowledge discovery and assessment environmental tension upon the population by detail analysis of endogenous and exogenous diseases cases. Thus, the main practical aim of the paper is to create a situation assessment agent-based system, monitoring the environment pollution and following the correspondent changes in human health, generating a set of alternatives for successful and sustainable situation management.

3. Environmental impact decision support system

The main point of using decision support systems (DSS) is to provide a user with the possibility to consult with an automated system while making decisions. The DSS, as a rule, includes a set of procedures, starting from data determination and processing, and finishing by generation and evaluation of alternatives. Thus, a typical DSS can be logically divided and represented by three main calculation modules or levels: the first one, responsible for data fusion and pre-processing, the second, dedicated to necessary calculations (modeling, data mining, etc.) and the third, which executes simulation and manages human–computer interaction.

Fig. 1 shows the proposed agent-based decision support system (ADSS) architecture, which was developed in accordance with the typical structure of a DSS. The system consists of three levels, the first is aimed for meta-data creation, the second is responsible for hidden knowledge discovering, and the third level provides real-time decision support making, data distribution and visualization. The main tasks are completely described at each level.

3.1. The information fusion level

The first level, provided at Fig. 2, is named "Information Fusion" and it acquires data from diverse sources and in different format types. After stating the main aim of the work, an information search has shown to be mandatory, gathering the relevant background information and expert knowledge. This information has been introduced in a concept hierarchy, defining a sequence of mappings from the set of elementary concepts to the more general ones. The concept hierarchy has been created using ontology editor Protégé 3.2 (Gennari et al., 2003), and it includes the information about the regions of interest (e.g. country: Spain, region: Castilla-La Mancha, city: Albacete, etc.) and the examined indicators. So, it has been possible to obtain the domain of interest, making accent in regions, which are characterized with some environmental pollution and human health level (see Fig. 3). As an example, the part of the concept hierarchy for dimension "pollution and health population indicators" is represented in Fig. 4.

The information of interest includes data about morbidity, detailed by classes (ICD codification), age and gender; and data about regional pollution, including data on soil, air, water contamination, traffic and industrial activity, measured directly and indirectly (in accordance with ISO 14031). ICD contains codes of diseases which are used for mortality class determination and as standards in use for statistical retrospective information presentation by responsible committees. ISO 14031 points that there are

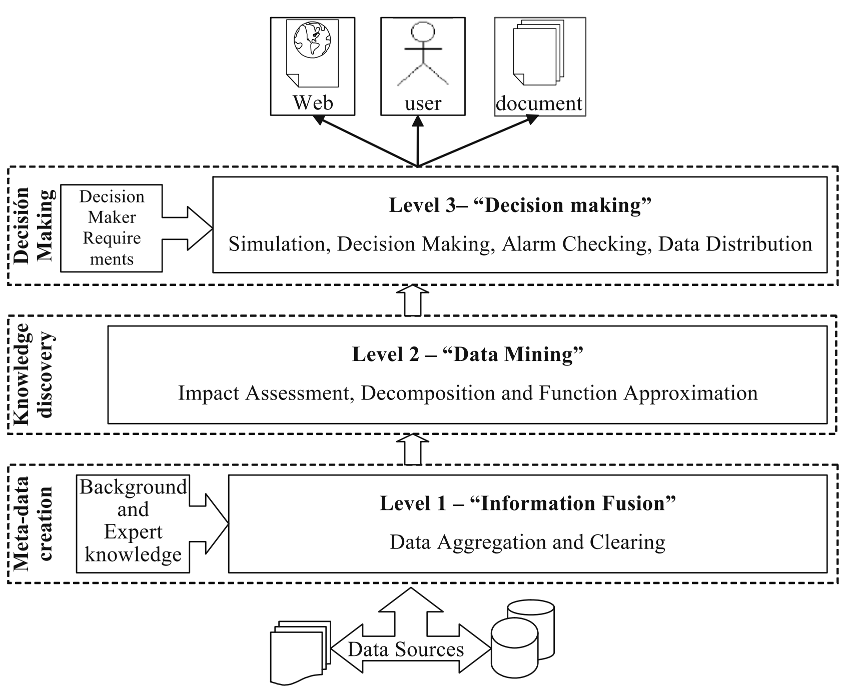

Fig. 1. The general system schema with main tasks.

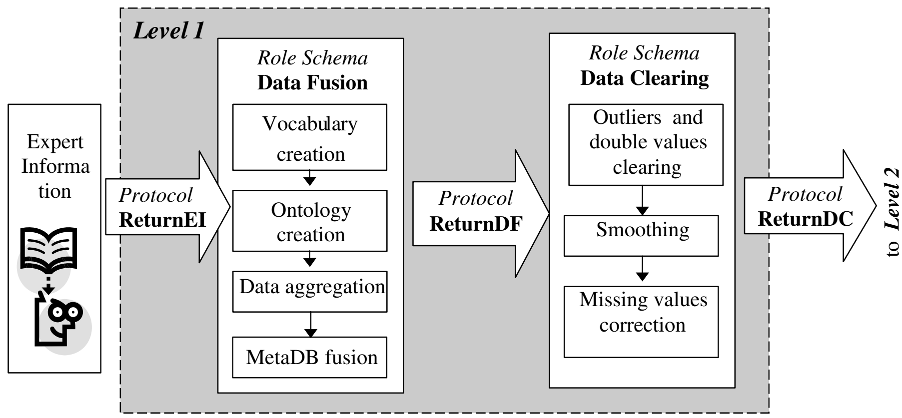

Fig. 2. The structure of the first level.

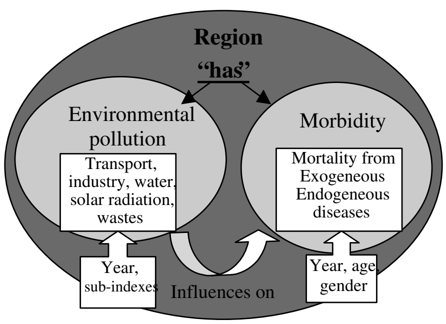

Fig. 3. The domain of interest.

environment performance indicators, such as "Number vehicles in use", "Electro energy usage" and other similar ones can be used as indirect indicators of environmental pollution, as they reflect the life cycle characteristics of respective direct indicators.

The data the ADSS has to deal with is presented in the forms of distributed data storages, which differ in: (1) data representation formats, (2) data syntaxes and (3) semantics (Cardoso, 2006, 2007). The first problem deals with the format types, as information may be stored in flat files as well as in form of XML, RDF, HTML files, relational data bases, etc. The second problem is caused by the fact that a concept may be measured in different scales and be registered by a different number of attributes. Thus, the aim of this level is to solve these problems and provide the system with homogeneous fused meta-data. The tasks solved at level 1 are grouped into two subtasks: "Data Fusion" and "Data Clearing", fully controlling data retrieval and maintenance and executing all the necessary pre-processing functions at the every step. Firstly, incoming raw data are fused to form a meta-data base, consisting in time series, which then pass throw the sequential steps of data processing, that is to say, noise reduction, outlier elimination and doubling, and inconsistent and missing values checking. The subtask "Data Fusion" supposes the following logical steps:

- create a vocabulary for the domain of interest;
- create an ontology of the domain of interest;
- select data from different sources with respect to the ontology;
- aggregate data in one meta-data base.

Then the protocol ReturnDF delivers meta-data base to subtask "Data Clearing", where data for outliers, double records, dealing with missing values and smoothing time series is checked. The final meta-data base consists of

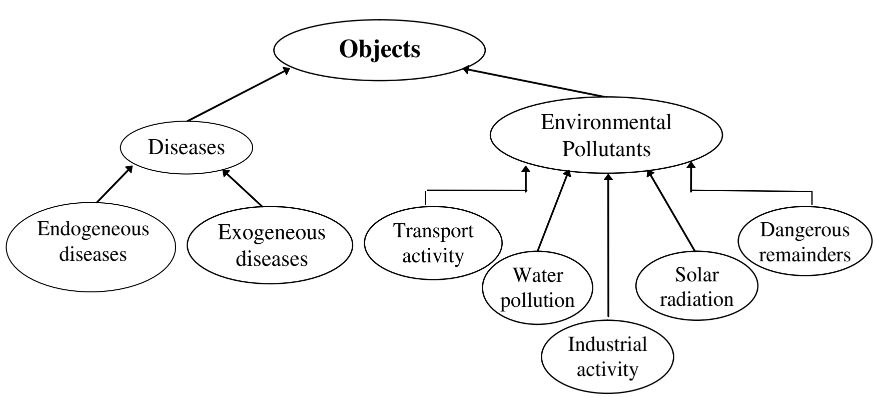

Fig. 4. The part of the concept hierarchy for dimension "pollution and population health indicators".

sequences of ordered indicator values, measured at equal time intervals (time-series). It is delivered to the next layer for knowledge discovering by the ReturnDC protocol.

3.2. The data mining level

The second level is responsible for knowledge mining from meta-data base (see Fig. 5.). There will be a number of different techniques used for data analysis. Notice that there are three subtasks at this level: "Decomposition", "Impact Assessment" and "Function Approximation". Taking into account the nature of meta-data base, the following requirements to DM algorithms are established: robustness, ability to work with missing data values, and ability to deal both effectively with short and large data sets. Only the DM techniques which satisfy these requirements will be utilized.

In general, the aim of this proposal is to reveal interconnections between the health and pollution indicators and qualitatively evaluate the influence of the latter. In order to select and evaluate qualitatively the most significant pollutants for every class, the procedure of principal component analysis (PCA) is sequentially delivered for every class of diseases and the totality of environmental pollution indicators (traffic activity, water, air pollution, etc.). We utilize agents executing the artificial neural networks (ANN) approach and statistical procedure of principal component analysis (Sokolova, Rasras, & Skopin, 2006).

The next subtasks solved at level 2 are "Decomposition" and "Function approximation". Models for computer simulation, forecasting and decision making are required. In order to decrease the number of simultaneously processed indicators and be saved from inter-correlation and multi-co-linearity between them, a procedure of factor space decomposition by calculating the correlation matrix and its further decomposing is initiated (Artemenko, Hudec, Lapina, & Sokolova, 2004). We will receive a set of independent variables, which do not correlate significantly between them and with a certain class of diseases and can be used as factors for modeling this class of diseases. The protocol ReturnD transfers information about "Decomposition" to "Function Approximation".

Then, for every indicator those models revealing their tendencies are extracted. The procedures of function approximation are based on different methods, which are stored in the library of DM methods. For example, statistical methods utilizing regression modeling (linear and non-linear) for activities DataMiningMethod1 and DataMiningMethod2, decision trees techniques for activity DataMiningMethod3, and other techniques for activity DataMiningMethodN, may be called. The methods execute in interleaved mode. As a result, there are several different models for every indicator and the best one will be chosen. Then we check if the models are adequate to data sets: we approximate initial data for them, compare the real and the approximated data sets and evaluate how well these models satisfy the statistics. The models which best fulfill the requirements are selected. All the results of data transformations are distributed to the next level for decision making.

3.3. The decision making level

The third level carries out a set of procedures including models evaluation, computer simulation, decision making and forecasting based on the models created on the previous level and the results of the PCA.

The structural schema of this level is shown in Fig. 6. There are three subtasks: "Computer Simulation and Forecasting", "Decision Making" and "Data Distribution" and protocols ReturnFA (which delivers outcomes of function approximation) and ReturnIA (delivering results of impact

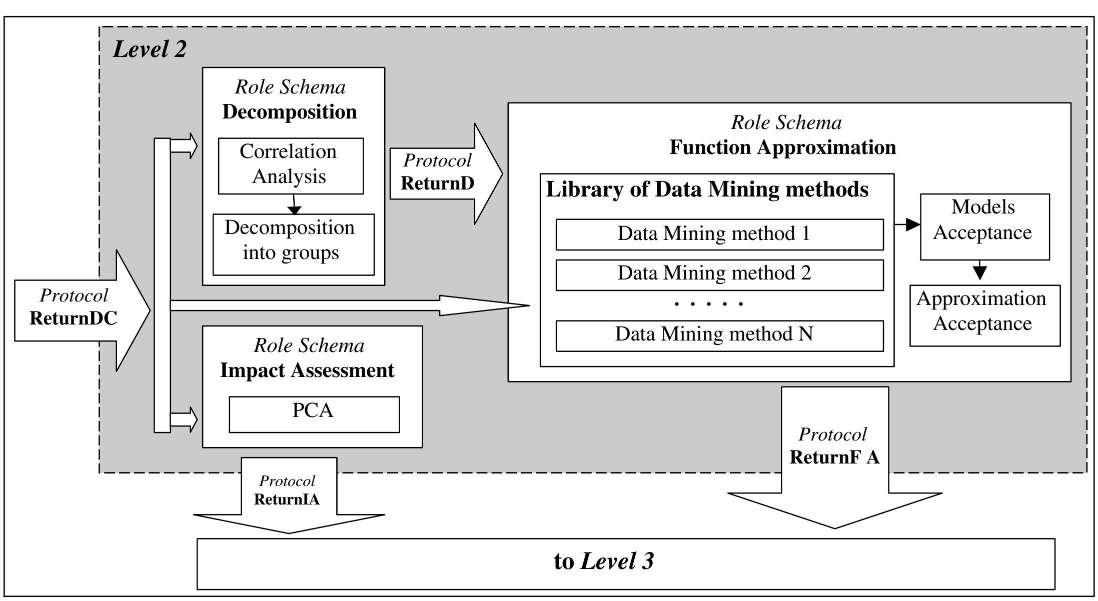

Fig. 5. The structure of the second level.

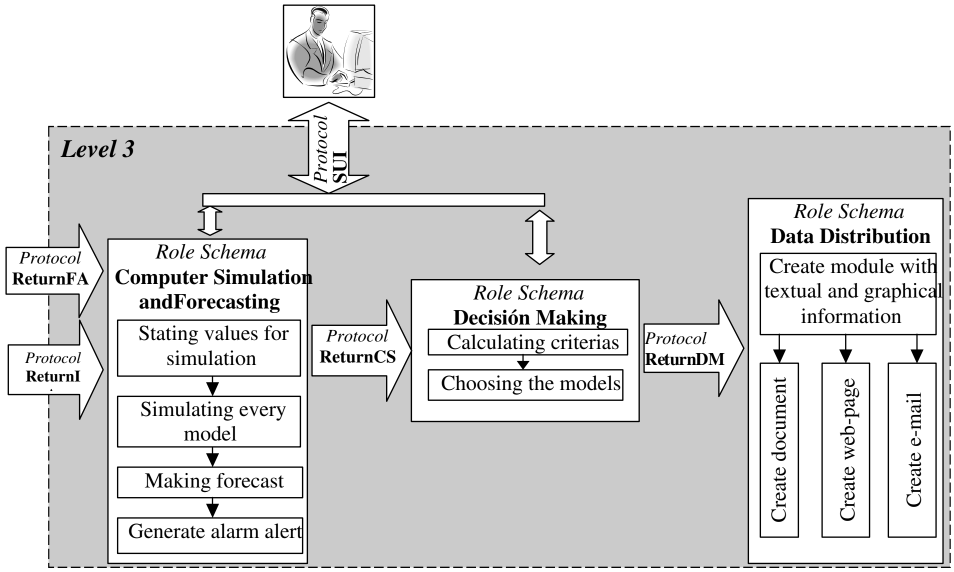

Fig. 6. The structure of the third level.

assessment), protocol SUI (responsible for interaction with the user), protocols ReturnCS and ReturnDM (which deliver simulation and decision making results for further processing).

The end-user (a person who makes decisions) interacts with the system through the protocol SUI (System–User Interaction). The user chooses the indicator he wants to examine and initiates a procedure of computer simulation. He may choose the independent variables, state their values and forecasting period. Then the information is delivered to "Computer Simulation and Forecasting" that performs computations for every model and repeats the operation until outcomes for all the models are received. Then the protocol SUI delivers results to the end-user (through the visual interface) for future decision making.

"Decision Making" recommends the most optimal variants of computer simulation, which are selected by standard decision making criteria (Bayes, minimax, Hurvitz, etc.) in agreement with its knowledge. In case the forecasted values exceed or are likely to exceed the permissible levels, an alarm message will be generated, visualized and sent to the user.

At the last step, the information is delivered to the final destination – end-users and applications in the forms of web-messages, textual files, e-mails and visual presentations. This level transforms the revealed information including results of computer simulation, forecasting and decision making into understandable multiple forms. These tasks are realized in "Data Distribution", which operates on combining textual and graphical descriptions of recommendations.

4. ADSS analysis and design with Prometheus

There are many alternative agent-based System Engineering methodologies, including MaSE (DeLoach, Wood, & Sparkman, 2001), Gaia (Wooldridge, Jennings, & Kinny, 2000), Agent ULM (Bauer, Müller, & Odell, 2001), Prometheus (Padgham & Winikoff, 2004), Tropos (Bresciani, Perini, Giorgini, Giunchiglia, & Mylopoulos, 2004), among others. In this work, the analysis and design of the ADSS was made under Prometheus methodology. The ADSS design has entirely been modeled by means of Prometheus methodology through Prometheus Design Tool (Thangarajah, Padgham, & Winikoff, 2005). The tool provides the possibility of checking the consistency of the created system and of generating a skeleton code for JACK Intelligent Agents (Winikoff, 2005) development tool, as well as design reports in HTML.

The process of creating a multi-system architecture, in accordance with the Prometheus methodology, consists of three phases, which are:

(1) System specification, aimed to the identification of multi-agent system entities, such as actors, system goals, scenarios, actions, percepts and roles.
(2) High-level (architectural) design, which is centered in the description of agent role coupling, general system structure and interaction protocols.
(3) Detailed design, in which each agent is described in detail in terms of capabilities, events, plans and data.

The system is aimed to fulfill the enumeration of general goals, subdivided into plans and actions, in MAS terminology, as shown in Fig. 7. This enumeration includes a set of scenarios starting from meta-ontology creation and finishing with decision making and data distribution.

The analysis has led to the identification of two roles on the first level, three roles on the second level and three roles on the third level of the ADSS. The roles show the detailed functionality of the system. Agents are responsible for execution of the extracted activities.

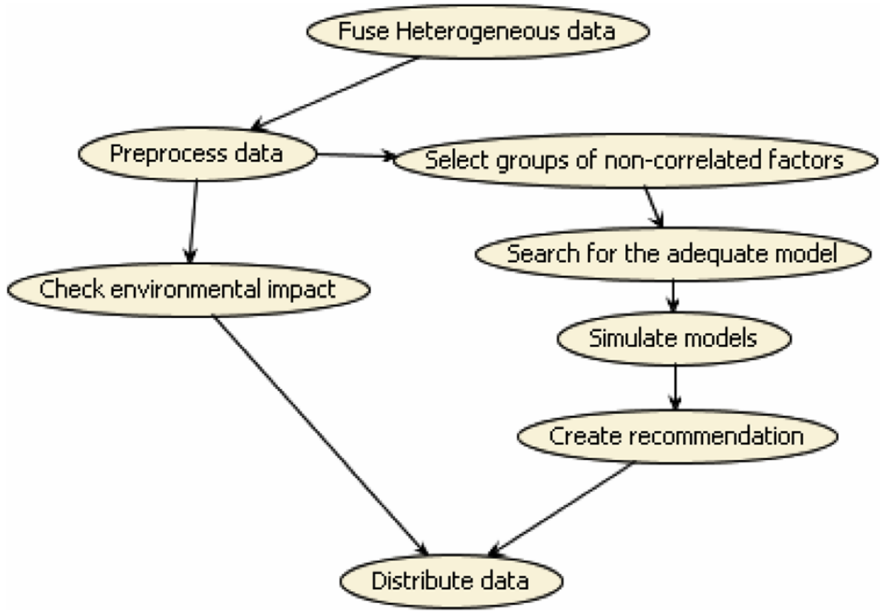

Fig. 7. The tree of aims.

The goals drawn in Fig. 7 repeat the main points of a traditional decision making process, which includes the following phases: (1) problem definition, (2) information gathering, (3) alternative actions identification, (4) alternatives evaluation, (5) best alternative selection, and, (6) alternative implementation. Phases 1 and 2 are performed on the initial step, when the expert information and initial retrospective data is gathered. Phases 3–5 are solved by means of the ADSS, and phase 6 is supposed to be undertaken by the decision maker.

The initial analysis of the system results in obtaining and describing the system roles and protocols (see Fig. 8). Being implemented by means of the Prometheus Design Tool, the Analysis Overview Diagram of the ADSS enables seeing the high-level view composed of external actors, key scenarios and actions (see Fig. 9).

The proposed ADSS presupposes communication with two actors. One actor is named as "Expert" and it embodies the external entity which possesses the information about the problem area – in more detail, it includes the knowledge of the domain of interest represented as an ontology – and delivers it through protocol ReturnEI to the ADSS. The second actor, named "Decision Maker", is involved in an interactive process of decision making

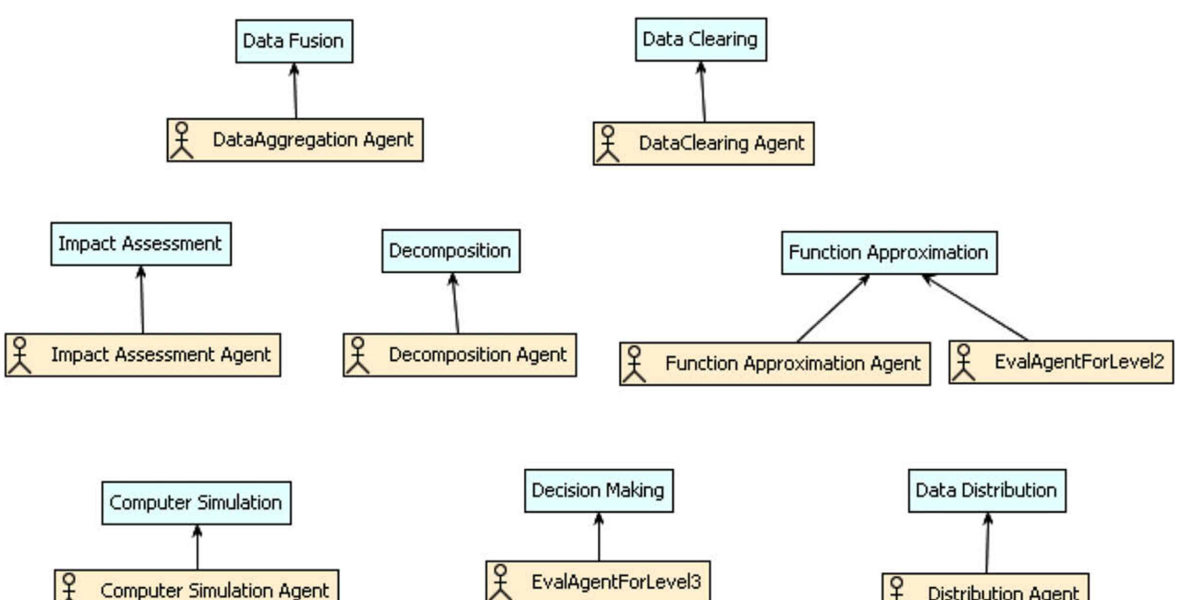

Fig. 8. The agents of the ADSS and the corresponding roles.

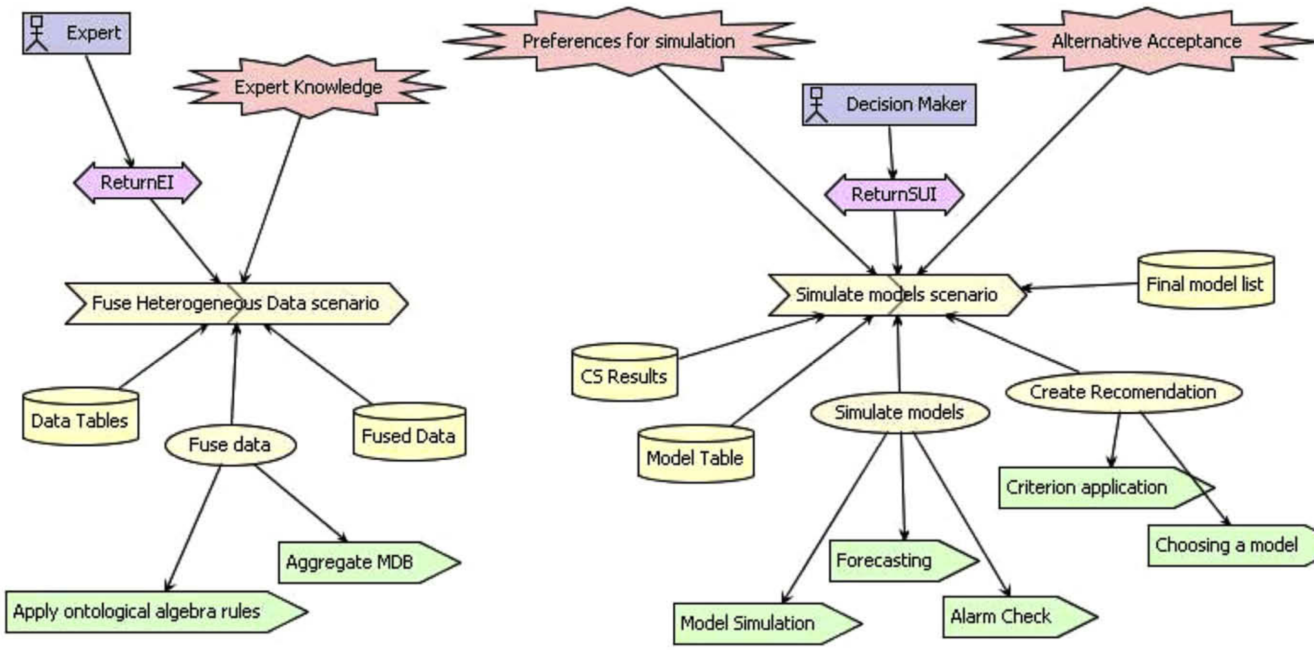

Fig. 9. The ADSS analysis overview diagram.

and choosing the optimal alternative. This actor communicates with agents by message passing through protocol ReturnSUI, stating the model, simulation values, prediction periods, levels of variable change, etc. It accepts the best alternative in accordance with its proper believes and the ADSS recommendation model.

The architecture of the ADSS is offered in Fig. 10. There, the proposed system is logically and functionally divided into three layers; the first is dedicated to meta-data creation (information fusion), the second is aimed to knowledge discovery (data mining), and the third layer provides real-time generation of alternative scenarios for decision making. Fig. 10 also provides a look on connections between agents with correspondent interactions and undertaken sets of actions. This view gives a sufficient understanding of the entire system design.

The system resembles a typical organizational structure, as the agents are strictly dedicated to work with the stated sets of data sources, to solve the particular tasks, and these are triggered when all the necessary conditions are fulfilled and there are positive messages from previously executed agents (Weiss, 2000). The system includes a set of roles, which are correlated with the main system functions and a set of agents related to each role (see Fig. 8). Actually, mostly every agent is associated to one role; only in case of "Function Approximation" role, there are two agents, one for data mining, and the other one for validation.

Each type of agent has its own sphere of competence, belonging to some agent type. Agents possess different characteristics and use various techniques. From an initial analysis the following agent types are obtained: *DB Handling agent*, *Analysis agent*, *Evaluation agent*, *Simulation agent*, and *Distribution agent*. Agents belong to BDI (beliefs-desires-intentions) (Rao & Georgeff, 1991) type, and their "Believes" are represented by information data resources necessary for every correspondent agent, their "Desires" are consist from a set of possible methods, which agents may execute, and the "Intentions" are formed as a result of reasoning, with respect to agent state and changes in its environment. The detailed description of the agents and their BDI characteristics are given in work (Sokolova & Fernández-Caballero, 2007) and the description of agent types is given below.

4.1. DB Handling agent type

The *DB Handling agent* type contains two agents, the *Data Aggregation agent* and the *Data Clearing agent*. The difference between their competence areas has been introduced in their names; the first one is oriented to data fusion, and the second one is focused on data pre-processing.

The *Data Aggregation agent*. The *Data Aggregation agent* (DAA) is responsible for initial information reception and meta-data creation. The agent plays an essential role as it has to realize the plan of important tasks, including: (1) reception of expert information and initial data, (2) scanning of data sources for selecting the values related to identical hierarchical and semantic levels of the meta-ontology, (3) pooling homogeneous data into a meta-data structure. The DAA communicates with the *Data Clearing agent* (DCA) through communication protocol ReturnDF, sending a message that confirms meta-data creation and triggers data pre-processing.

The *Data Clearing agent*. The *Data Clearing agent* performs all data pre-processing procedures, including outliers

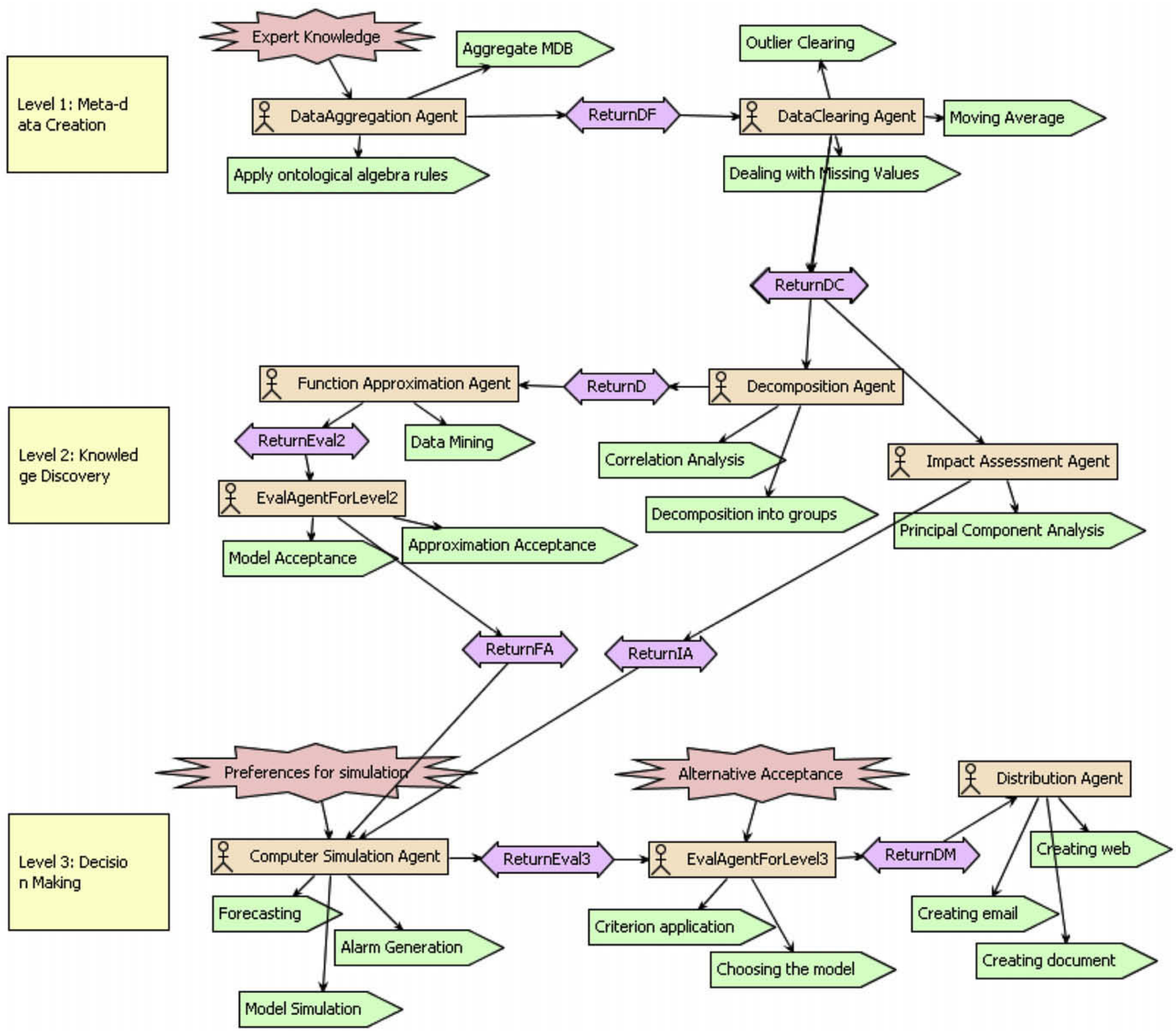

Fig. 10. The ADSS system overview diagram.

and anomalies detection, dealing with missing values, smoothing, normalization, etc. The final meta-data consists of sequences of ordered indicator values measured at equal time intervals (time series).

4.2. Analysis agent type

The *Impact Assessment agent*. The single agent that belongs to this type, namely the *Impact Assessment agent* (IAA), solves the PCA procedure for every type of disease and the totality of pollutant indicators. The outcomes are interpreted as the determination of the most influencing subsets of pollutants for every nosology. It is message-triggered from the DCA and after being executed it creates a file with the results of its work. The IAA delivers messages to the *Simulation agent* through protocol ReturnIA.

The *Decomposition agent*. The main task of the *Decomposition agent* (DA) is to separate the totality of factors (pollution indicators) and output variables (morbidity) into subsets (commonly intersected) with respect to every output variable (nosology). Decomposition is aimed to select for every output variable those factors that (1) do not correlate significantly with the output variable – in order to avoid inter-correlation, or, (2) do not correlate significantly between themselves – in order to avoid multi-co-linearity. The DA creates a correlation matrix and analyzes it. The *Decomposition agent* creates an output source used then by the *Simulation agent* and the *Distribution agent*.

The *Function Approximation agent*. The *Function Approximation agent* (FAA) carries out a core ability: it explores unknown relationships, trends and connections in data, received from the DA agent. The FAA is competent in applying DM techniques, including theory-driven methods (linear, non linear regression, ANOVA, etc.) as well as data-driven methods (neural networks, decision trees, rules, etc.).

4.3. Evaluation agent type

The *Evaluation agent*s are involved in two roles: "Function Approximation" and "Decision Making", which also belong to different levels. In role "Function Approximation" the *Evaluation agent* is thought to calculate statistical performance characteristics for the model, created by the FAA, and the model's approximation abilities. To do it, we approximate initial data by the models, compare the real and the approximated data sets and evaluate the performance statistics. In role "Decision Making" the *Evaluation agent* applies decision making theory criteria.

4.4. Simulation agent type

The Computer *Simulation agent*. Actually, there is one agent that belongs to this type, namely the *Computer Simulation agent* (CSA), which works with models created by the FAA and accepted by an *Evaluation agent*. The simulation process involves the interaction with the user. During this process the CSA (Fig. 11) learns the user's preferences: the model to be simulated, the factors the user is interested in analyzing, the forecast period, etc. The CSA simulates a model within capability "Simulation", compares the outputs with the permitted and hazardous levels within capability "Alarm Check" and shows the results to the user in order to be revised and accepted, or refined and repeated. This creates a decision support system, which helps the user in decision making by providing several alternatives.

4.5. Distribution agent type

The *Distribution agent* (DDA) emphasizes in data visualization and delivering to the user. The revealed information is transformed by the DDA into multiple understandable forms. These tasks are performed by the *Distribution agent*, which operates by combining textual and graphical descriptions of recommendations. This information jointly with computer simulation results for chosen models (including forecast and alarm check) are delivered. This includes impact assessment outcomes (which discover the exposure of urban pollution on the morbidity of people), correlation results (which reveal dependencies between pollutants and different types of diseases), and, optionally, those models recommended by the system to be accepted as optimal for regional development and situation correction.

5. ADSS implementation in JACK design environment

The JACK design environment (JDE) is a software package for agent-based applications development in Java-based environment JACK Intelligent Agents. The JDE has a visual interface, which supports application creation, which can be created directly in JDE, or be imported, for example, from Prometheus Design Tool, the graphical editor which provides agent systems design in accordance with its methodology. The JDE enables building applications by providing a visual representation of the system components, in two modes: agent mode and team mode.

Let us now show the use of JACK Intelligent Agents for one particular agent of the overall system. Fig. 12 gives an illustration of the *Data Aggregation agent* diagram, created in JACK. Remember that this agent is responsible for data fusion. It executes six plans: FuseMorbidityData, FuseWaterData, FuseWaterData2, FuseMinesData, FusePetrolData and FuseWasteData to retrieve and to extract different types of records from flat files, and to integrate them into a single storage. The necessity to create many plans was stated by the fact that data are stored in tables with different data organization that require particular data retrieval strategies for every type of data source.

An agent has private believes, which are formed as a result of data extraction, and are stated in form of two belief sets: retrievedData and retrievedDiseases. Belief sets have different number of fields, as it follows from determination of properties for "Morbidity": (1) Region, (2) Measurement

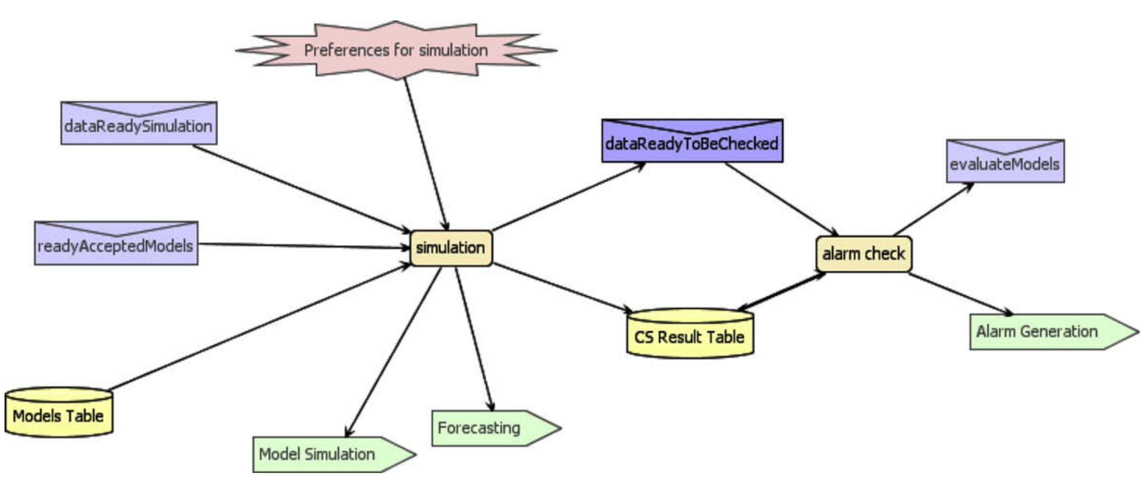

Fig. 11. The *Data Aggregation agent* capabilities diagram in Prometheus.

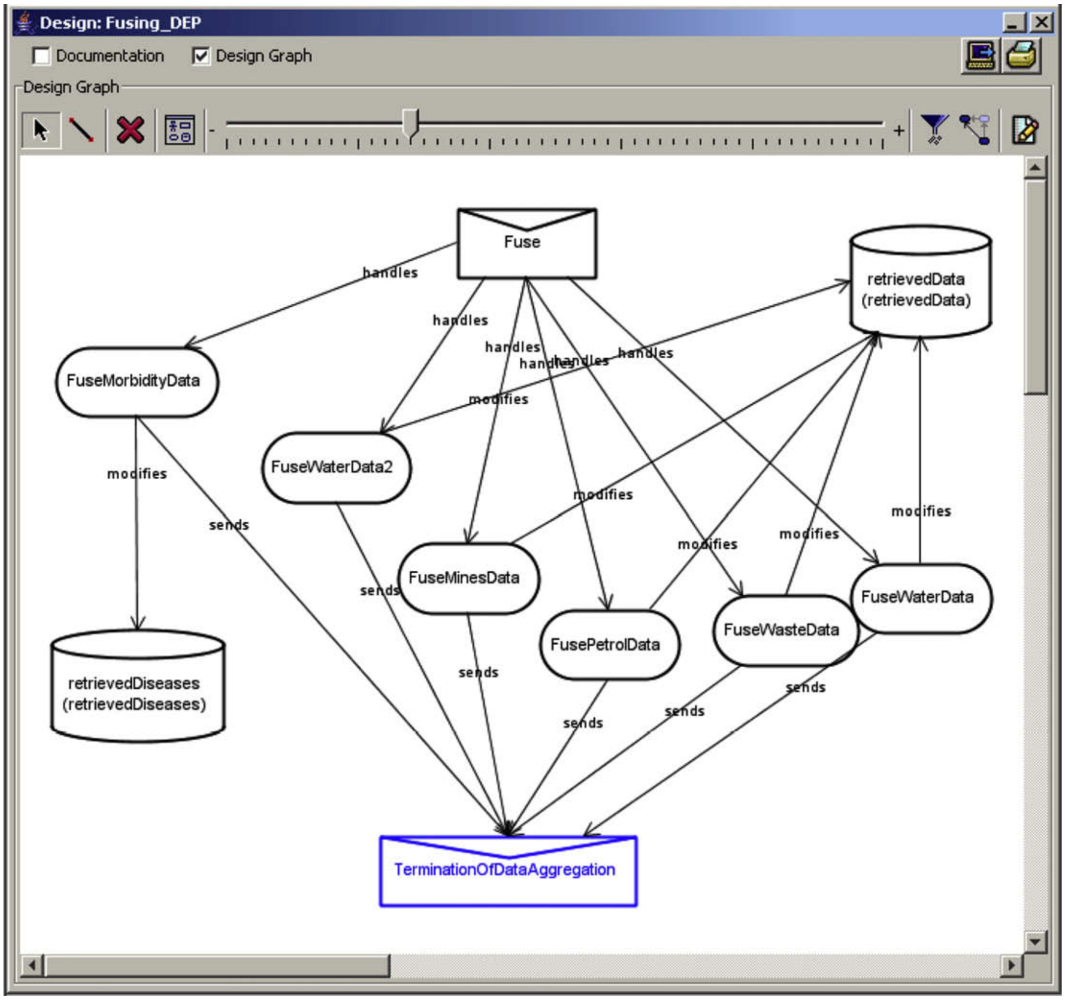

Fig. 12. "Data Fusion" in JACK design environment.

period, (3) Disease class, (4) Gender, (5) Age, (6) Value, and, (7) Scale; and for "Pollution": (1) Region, (2) Measurement period, (3) Pollutant, (4) Value, and, (5) Scale. During the fusion, the agent controls if duplicate records appear, and eliminates them. According to Fig. 12 the *Data Aggregation agent* has a capability *Fusing*, which includes the plans, triggered by the Fuse event. As soon as the agent executes a plan, it sends a message with external event "TerminationOfDataAggregation", triggering the *Data Clearing agent*, which starts to pre-process data.

6. Conclusions

In this paper an approach to developing an agent-based decision support system has been introduced. The software agents use DM methods for knowledge discovery, which will be used as a foundation for decision making and recommendation generation. The nature and peculiarities of experimental data and expert knowledge used in the system proposed have been discussed. An ontology has been described and a general system structure with detailed explanation of the roles and interactions in accordance with Prometheus methodology has been presented.

The ADSS provides all the necessary steps for standard decision making procedure by utilizing intelligent agents. The levels of the system architecture, logically and functionally connected, have been presented. Real-time interaction with the user provides a range of possibilities in choosing one course of action from among several alternatives, which are generated by the system through guided data mining and computer simulation. The system is aimed to regular usage for adequate and effective management by responsible municipal and state government authorities.

The proposed ADSS architecture may be generalized and applied in closely related areas, such as economics, sociology, public health, etc. Some specific DM techniques, for example, econometric system dynamic models, can be used in these kinds of applications. The system developed is being used as a pilot project in University of Castilla-La Mancha, Spain and Kursk State Technical University, Russia.

Acknowledgements

Marina V. Sokolova is the recipient of a Postdoctoral Scholarship (*Becas MAE*) awarded by the *Agencia Españ-*

ola de Cooperación Internacional of the Spanish *Ministerio de Asuntos Exteriores y de Cooperación*.

References

Artemenko, M., Hudec, O., Lapina, T., & Sokolova, M. V. (2004). The information-analytical program tool for regional decision making in social sphere. *Telecommunications, 9*, 42–44.

Athanasiadis, I. N., Mentes, A. K., Mitkas, P. A., & Mylopoulos, Y. A. (2005). A hybrid agent-based model for estimating residential water demand. *Simulation, 81*(3), 175–187.

Bauer, B., Müller, J. P., & Odell, J. (2001). Agent UML: a formalism for specifying multiagent software systems. *International Journal of Software Engineering and Knowledge Engineering, 11*(3), 207–230.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., & Mylopoulos, J. (2004). Tropos: An agent-oriented software development methodology. *Autonomous Agents and Multi-Agent Systems, 8*(3), 203–236.

Cannoy, S. D., & Iyer, L. (2007). Semantic Web standards and ontologies in the medical sciences and healthcare. In *Semantic Web technologies and e-business: Toward the integrated virtual organization and business process automation*. Idea Group Inc..

Cardoso, J. (2006). Developing dynamic packaging applications using Semantic Web-based integration. *Transactions on Information Science and Applications, 3*(4), 729–736.

Chang, C. L. (2006). A study of applying data mining to early intervention for developmentally-delayed children. *Expert Systems with Applications, 33*(2), 407–412.

Chen, H., & Bell, M. (2002). Instrumented city database analysts using multi-agents. *Transportation Research, Part C, 10*, 419–432.

DeLoach, S. A., Wood, M. F., & Sparkman, C. H. (2001). Multiagent systems engineering. *International Journal of Software Engineering and Knowledge Engineering, 11*(3), 231–258.

Foster, D., McGregor, C., & El-Masri, S. (2006). A survey of agent-based intelligent decision support systems to support clinical management and research. In *First international workshop on multi-agent systems for medicine, computational biology, and bioinformatics*, MAS*BIOMED'05.

Gennari, J. H., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubézy, M., Eriksson, H., et al. (2003). The evolution of Protégé: An environment for knowledge-based systems development. *International Journal of Human Computer Studies, 58*(1), 89–123.

Gorodetsky, V., Karsaeyv, O., & Samoilov, V. (2005). Multi-agent and data mining technologies for situation assessment in security-related applications. In *Advances in soft computing* (pp. 411–422). Springer.

International Classification of Diseases (2007). http://www.who.int/classifications/icd/en/.

ISO 14031:1999. Environmental management – Environmental performance – Guidelines. http://www.iso.org/.

Kebair, F., & Serin, F. (2006). Multiagent approach for the representation of information in a decision support system. *Lecture Notes in Computer Science, 4183*, 98–107.

Li, S. (2007). AgentStra: An Internet-based multi-agent intelligent system for strategic decision-making. *Expert Systems with Applications, 33*(3), 565–571.

Liu, L., Qian, L., & Song, H. (2006). Intelligent group decision support system for cooperative works based on multi-agent system. In *Proceedings of the 10th international conference on CSCW in design* (pp. 574–578), CSCWD 2006.

Ly, T. C., Greenhill, S., Venkatesh, S., & Pearce, A. (2003). Multiple hypotheses situation assessment. *Proceedings of the Sixth International Conference of Information Fusion, 2*, 972–978.

Ossowski, S., Fernández, A., Serrano, J. M., Pérez-de-la-Cruz, J. L., Belmonte, M. V., Hernández, J. Z., et al. (2004). Designing multiagent decision support system: The case of transportation management. In *Proceedings of the 3rd international joint conference on autonomous agents and multiagent systems* (pp. 1470–1471), AAMAS 2004.

Padgham, L., & Winikoff, M. (2004). *Developing intelligent agent systems: A practical guide*. John Wiley and Sons.

Petrov, P. V., & Stoyen, A. D. (2000). An intelligent-agent based decision support system for a complex command and control application. In *Proceedings of the sixth IEEE international conference on complex computer systems* (pp. 94–104), ICECCS'00.

Rao, A. S., & Georgeff, M. P. (1991). Modeling rational agents within a BDI-architecture. In *Proceedings of the 2nd international conference on principles of knowledge representation and reasoning* (pp. 473–484), KR'91.

Sokolova, M. V., & Fernández-Caballero, A. (2007). An agent-based decision support system for ecological-medical situation analysis. *Lecture Notes in Computer Science, 4528*, 511–520.

Sokolova, M. V., Rasras, R. J., & Skopin, D. (2006). The artificial neural network based approach for mortality structure analysis. *American Journal of Applied Science, 3*(2), 1698–1702.

Thangarajah, J., Padgham, L., & Winikoff, M. (2005). Prometheus design tool. In *Proceedings of the 4th international joint conference on autonomous agents and multiagent systems* (pp. 127–128), AAMAS 2005.

Urbani, D., & Delhom, M. (2005). Water management policy selection using a decision support system based on a multi-agent system. *Lecture Notes in Computer Science, 3673*, 466–469.

Weiss, G. (2000). *Multiagent systems: A modern approach to distributed artificial intelligence*. The MIT Press.

Winikoff, M. (2005). JACK™ Intelligent Agents: An industrial strength platform. Multi-Agent Programming: Languages, Platforms and Applications. *Multiagent Systems, Artificial Societies, and Simulated Organizations, 15*, 175–193.

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