A Novel WebGIS-based Situational Awareness Platform for Trustworthy Big Data Integration and Analytics in Mobility Context

Susanna Bonura, Giuseppe Cammarata, Rosolino Finazzo
Giuseppe Francaviglia, and Vito Morreale

R&D Laboratory - ENGINEERING Ingegneria Informatica S.p.A.,
Viale Regione Siciliana, 7275, Palermo, Italy
{susanna.bonura,giuseppe.cammarata,giuseppe.francaviglia,
rino.finazzo,vito.morreale}@eng.it
http://www.eng.it

Abstract. The availability of big amounts of dynamic data from several sources in mobility context and their real time integration can deliver a picture for emergency management in urban and extra-urban areas. A WebGIS portal is able to support the perception of all elements in current situation. However, in general, during the observation decision maker's attention capacity is not sufficient to address concerns due to information overload. A situational picture is necessary to go beyond the simple perception of the elements in the environment, supporting the overall comprehension of the current situation and providing predictions and decision support. In this paper we present *MAGNIFIER*, a WebGIS-based intelligent system for emergency management, to entirely support the real-time situational awareness. Starting from the current situation and by using the practical reasoning model by Bratman, *MAGNIFIER* is able to suggest the appropriate course of actions to be executed to meet decision maker's goals.

Keywords: Big Data Real-time Analytics, Big Data Integration, Decision Support System, Emergency Management, Intelligent System, Practical Reasoning, Situational Awareness, WebGIS

1 Introduction

The huge amount of dynamic data from several sources as well as geospatial and temporal information and their integration in real time can serve to deliver a picture of main events to be monitored for emergency management in urban and extra-urban areas. A crucial concept in the field of emergency management is that of *Situational Awareness* (SA) which, in general, with regards to being aware of what is happening around one in terms of where one is, where one is supposed to be, and whether anyone or anything around is a threat to one's health and safety.

Several models exist in literature on SA. Dominguez defines individual SA as the continuous extraction of environmental information to directing and anticipating future events [1].

Bedny and Meister consider a continuous loop on which SA directs the interaction with the world and such an interaction modifies SA. This interaction is motivated by the disparity between the decision maker's goals and the current perceived situation [2].

Smith and Hancock proposed a model stating that SA is neither resident on individuals nor in the world but rather on the interactions that are motivated by decision maker's schemata; the outcome of that interaction will modify existing schemata, which in turn directs further exploration [3].

But the model that has received most attention is the Ensdley's three-levels model: in the first level, training and experience directs attention to critical elements in the environment; the second level integrates elements that aid understanding the meaning of critical elements; and the last level considers understanding the possible future scenarios [4].

Today, the most advanced WebGIS tools (Geospatial Information Systems that use web technologies to communicate between a server and a client), are able to support the situational awareness in what regards the perception of all elements in current situation (according to the first level of the Ensdley's model). Thus, WebGIS can help decision makers in data integration and visualization, its localization to real-time analysis, and mapping of potential disasters to show vulnerable areas, critical situations and potential harm.

However, in general, in dynamic environments, decision maker's attention capacity is not sufficient to address attention demands resulting from information overload. Therefore, a situational picture is necessary which is able to go beyond the simple perception of the elements in the environment, supporting the overall comprehension of the current situation and the user's decision making process (by providing alerts, predictions and recommendations).

Currently, the existing open source WebGIS platforms just refer to real-time geospatial analysis and visualization about a certain phenomena (e.g Ushahidi [1] and Sahana Eden [2] for disaster events). Other advanced commercial solutions (e.g. archGIS Platform by Ensri [3]) are able to integrate information from different sources and to manage real-time data analysis, but they do not offer decision support with regards to the delivered situational picture. Furthermore, some GIS-based decision support systems exploit multicriteria decision analysis (MCDA) techniques to evaluate possible alternatives, but they are lacking a proper scientific foundation and some methods involve stringent assumptions which are difficult to substantiate in real-world situations [5][6].

In this paper we present *MAGNIFIER*, an open source WebGIS-based intelligent system for emergency management, designed and implemented to entirely support the real-time situational awareness, which starts from an appropriate

[1] `https://www.ushahidi.com`
[2] `http://eden.sahanafoundation.org/`
[3] `http://www.esri.com/software/arcgis`

perception of all elements in current situation, to come to an understanding of the meaning of those elements in an integrated form, and to the ability to project future states of the environment that are valuable for decision making.

In addition, *MAGNIFIER* provides decision maker with a decision support: it is able to suggest the appropriate course of actions to be executed to meet decision maker's goals, by reasoning on the current situation according to the practical reasoning model, namely the reasoning process directed towards actions [7]. Through such a model, *MAGNIFIER* is endowed with the ability to reason on the states of the environment where it is situated, its goals and its plans, so to meet its objectives while reasoning about failures (e.g. when some unwanted states occur), in order to learn new desired behaviors.

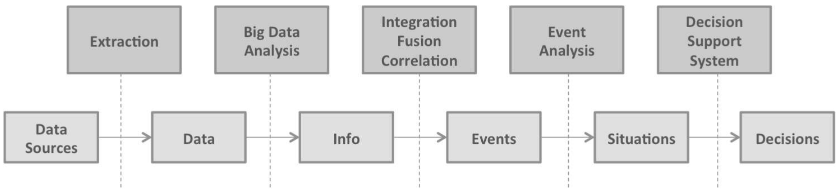

Fig. 1. Intelligent Information Processing Chain

MAGNIFIER is compliant with our data processing model we named *Intelligent Information Processing Chain* (Fig. 1). According to such a model, data collected from different sources (sensor networks, news, rss feeds, comments, tweets etc.) are processed to extract useful information (by means of advanced analytics). Some pieces of such information, properly integrated, fused and correlated, will result in events. Events are evaluated in order to create the situational picture that is at the basis of decision maker activities (possibly supported by automatic reasoning services).

In order to better explain how *MAGNIFIER* works, the following scenario is provided (Fig. 2): in Palermo city two urban areas are highly dangerous due to flooding. Such areas include critical infrastructures (schools, roads, etc.) with different vulnerability levels which need to be continuously monitored in case of adverse weather conditions. Usually, WebGIS platforms allow decision makers to choose a certain number of layers which, overlapped, form a situational picture. This picture helps decision maker to *perceive* the current situation, in terms of dangerousness, infrastructure vulnerabilities and potential risk.

MAGNIFIER proactively builds and provides the layer to help decision maker to better *understand* the situation picture by showing information regarding the danger and the vulnerability of infrastructures and suggesting the actions to be performed in order to restore situations in their normality.

Such a picture is called *MAGNIFIER Situational Picture* and shows a recommendation, characterized by a level of danger (high, medium, low), vulnerable infrastructures, type of risk, the action to be taken to prevent the risk and finally by the risk status (warning, alert, emergency).

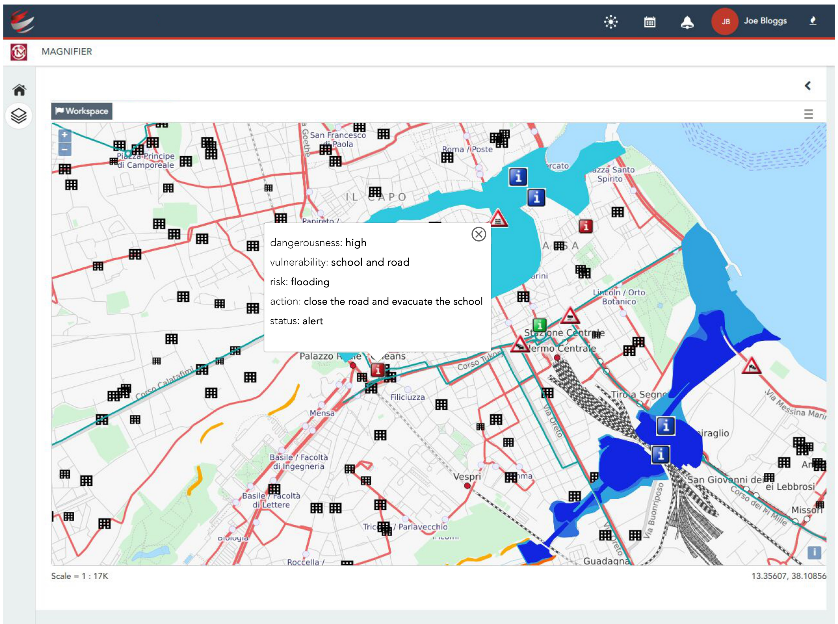

Fig. 2. An example of *MAGNIFIER* Situational Picture

This paper is organized as follows: in section II we give a brief overview of the The MAGNIFIER WebGIS-based Situational Awareness Platform and its functional components; then in sections III and IV we provide a description of the main technological choices focusing on the PRACTIONIST DSS component; finally in section V we describe the DSS execution logic through the scenario above mentioned.

2 The MAGNIFIER WebGIS-based Situational Awareness Platform

MAGNIFIER has been designed and implemented with the objective to store and manage data from several sources, relating to urban and extra-urban areas of interest and, therefore, provide support in the real-time investigation of situations of potential danger and in the decision making process.

For this aim, the TOREADOR environment[4] was adopted to support a trustworthy big data integration and analytics.

[4] The TOREADOR environment is part of a bigger platform under development through the TOREADOR initiative (`http://www.toreador-project.eu/`).

More in detail, such environment (Fig. 3) comprehends an integration layer, where some components from the *FIWARE* Catalogue were exploited [5]. The *FIWARE* Catalogue contains a rich library of public, royalty-free and open source components (Generic Enablers) with reference implementations that allow developers to put into effect functionalities such as the connection to the Internet of Things or Big Data analysis, making programming much easier.

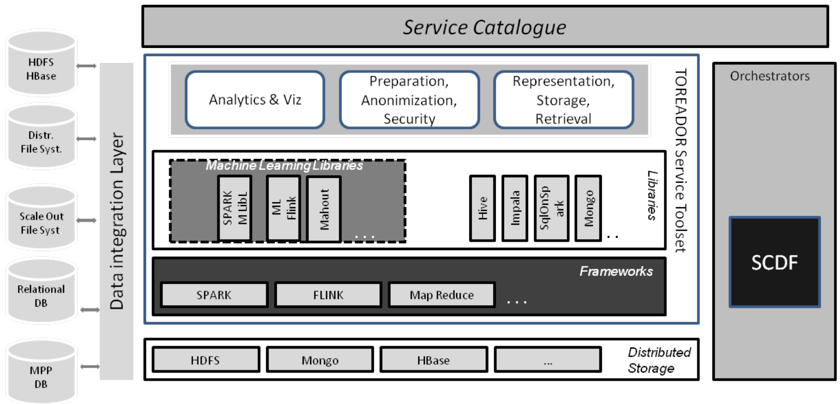

Fig. 3. The TOREADOR environment

In particular, in order to support the integration of data coming from several sources, we have properly installed in the integration layer, the *ORION* Context Broker, which is an implementation of the Context Broker GE, providing the NGSI9 and NGSI10 interfaces. Using these interfaces, clients can do several operations: to register context producer applications, to update context information, being notified when changes on context information take place or with a given frequency, to query context information.

Together with *ORION*, we used *Cygnus*, which implements a connector for context data coming from the *ORION* Context Broker and aimed to be stored in a specific persistent storage (in our case it is Mongo DB). Furthermore, we integrated *PROTON*, the CEP GE which analyzes event data in real-time, generates immediate insight and enables instant response to changing conditions. It provides means to expressively and flexibly define and maintain the event processing logic of the application, and at runtime it is designed to meet all the functional and non-functional requirements without taking a toll on the ap-

[5] *FIWARE* is a PPP European initiative (supported by 25 ICT players; Engineering Ingegneria Informatica S.p.A. is one of them) aiming to build a software platform dedicated to the creation of Future Internet applications, by leveraging on advanced technologies such as Cloud Computing, Internet of Things, Engineering Services, Data & Content Management, Advanced User Interfaces and Future Networks security (`https://www.fiware.org/`).

plication performance, so reducing application developers and system managers concerns.

Besides, the TOREADOR environment is able to make available a huge set of services for batch and stream data analytics, as well as for anonymization, preparation and security and for representation, storage and retrieval. Such applications, using the most cutting-edge Big Data libraries and frameworks (e.g. Spark MLlib, ML Flink, Mahout as libraries, and Spark, Flink, MapReduce as frameworks), are integrated into the environment for the deployment of applications in several domains and scenarios.

The Service Catalogue presents the services available in the TOREADOR environment with REST API.

Finally, the TOREADOR environment is endowed with Spring Cloud Data Flow (SCDF) [6], a cloud-native programming and operating model for composable data microservices, aiming at creating and orchestrating data pipelines for common use cases such as data ingestion, real-time analytics, and data import/export.

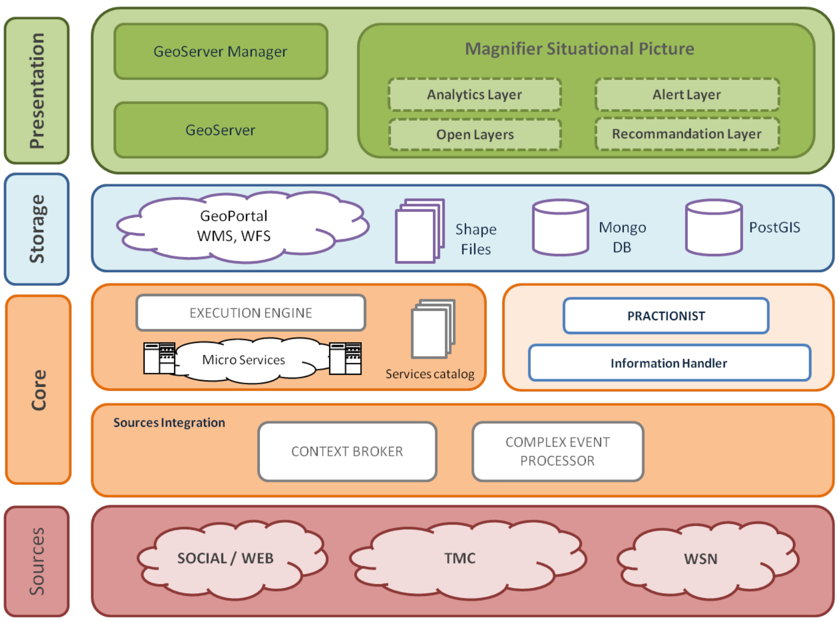

Fig. 4. Conceptual multi-layer architecture of *MAGNIFIER*

Several data sources are managed by the *MAGNIFIER* (Fig. 4). First of all, it is noteworthy that a Wireless Sensor Network prototype was designed and implemented so as to be installed in vehicles. The WSN consists of four sensor

[6] `https://cloud.spring.io/spring-cloud-dataflow/`

nodes, a gateway node and a controller for the network management. The gateway is responsible for forwarding the measures by each individual sensor to the controller. The sensor nodes are respectively composed by the following sensors: (i) noise, light, rain; (ii) speed and vibration; (iii) air quality, pressure, temperature, dust, humidity, carbon monoxide; (iv) infra-red light, soil and air infra-red temperature, ultra sonic distance. The controller is in charge of generating other virtual sensors starting from the measurements of real sensors, through sensor fusion techniques. Furthermore, it is responsible for decoding messages from the gateway and for the WSN management.

Through a Traffic Message Channel (TMC) receiver [7], real-time information which are transmitted on FM frequencies about traffic events and state of the roads/infrastructure, are received, properly decoded (according to the TMC protocol) and standardized (according to NGSI9 and NGSI10 standards).

Other kinds of data managed by *MAGNIFIER* are:

- news, RSS feeds, comments from web sources. This data is extracted from public web sources (i.e. institutions and organizations which are in charge for the control of territory) that provide real-time data on the state of roads and infrastructures, and on traffic events.
- Tweets from the TWITTER source. We access to tweets published by profiles that release information about the status of road infrastructure; the access to tweets is via public APIs provided by the TWITTER platform.
- Traffic open dataset, provided by different public bodies.

The core of *MAGNIFIER* is represented by several technological components which were extended and integrated to meet the system requirements. More in detail, drawing from the data sources, such layer is in charge of making data more and more elaborated until to get to define a situational picture composed by situations, alerts, recommendations (according to the *Intelligent Information Processing Chain* shown in Fig. 1).

Information, generated events and situational awareness acquired by the system are used to populate the various stores: postgis tables and shape files, used by geoserver to publish thematic layers.

PostGIS is the main free relational database with a geographical extension, which implements the support for geographic objects in PostgreSQL. PostGIS follows the directives of the Open Geospatial Consortium Simple Features Specification for SQL; it is developed by Refractions Research , and is an open source project that develops the spatial database technology.

[7] Traffic Message Channel is a technology for delivering traffic and travel information to motor vehicle drivers. It is digitally coded using the ALERT C protocol into RDS Type 8A groups carried via conventional FM radio broadcasts. It can also be transmitted on Digital Audio Broadcasting or satellite radio. TMC allows silent delivery of dynamic information suitable for reproduction or display in the user's language without interrupting audio broadcast services. Both public and commercial services are operational in many countries. When data is integrated directly into a navigation system, traffic information can be used in the system's route calculation.

The shapefile format is a popular geospatial vector data format for GIS software. It is developed and regulated by Esri as a (mostly) open specification for data interoperability among Esri and other GIS software products. The shapefile format can spatially describe vector features: points, lines, and polygons, representing, for example, water wells, rivers, and lakes. Each item usually has attributes that describe it, such asname or temperature.

The store MongoDB is used primarily to store all data acquired from the sources and any information or derived events, for further processing by the core components to make predictions and learning.

The layer on top of the Fig. 4 includes all the elements which characterize the user interface, that is the situational picture composed by situations, alerts, recommendations as interpreted by *MAGNIFIER*.

In the following section, the components of each layer will be detailed and the relation among them explained.

3 The MAGNIFIER Technological Architecture

MAGNIFIER is the result of the integration (and in certain cases of the extension) of a number of innovative open source technological solutions.

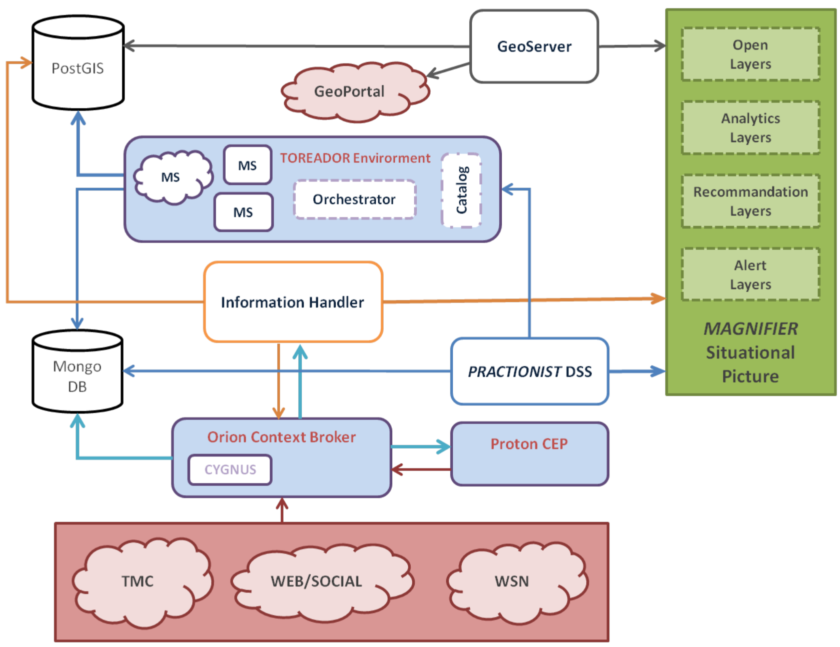

Fig. 5. Technological architecture of *MAGNIFIER*

The data acquired from the sources previously explained, is referenced with respect to time and space, normalized according to the NGSI9 and NGSI10 standards, and sent to *ORION*.

Besides these sources, other producers for *ORION* are:

- *PROTON*, which sends to it the complex events generated;
- the Information Handler, which send to it information properly elaborated.

The components subscribed to *ORION* as consumers, are:

- the *Cygnus* connector, which stores all elements passing through *ORION* (raw data, information, simple events, complex events) on MongoDB;
- *PROTON*, which is subscribed for certain events to generate complex events;
- the Information Handler, which is in charge of the transformation of information coming from sources and *PROTON* in more complex information by invoking the advanced analytics engine (e.g. R engine (`https://https://www.r-project.org/`)).
- *PRACTIONIST*, which manages high level events (coming from *PROTON* and the Information Handler), to generate alerts and recommendations so to recognize and analyze situations needed to be monitored (please refer to the section 4 for a brief understanding of PRACTIONIST).

The following pseudo-code shows three examples of rules to generate complex events starting from events notified by *ORION* to the *PROTON* CEP:

$$fog \Leftarrow dust\_poll \geq 0.30mg/m^3 \, and \, air\_qual \equiv fresh. \tag{1}$$

$$snow \Leftarrow soil\_temp \leq 4 \, andson\_dist \leq max\_son\_dist. \tag{2}$$

$$flooding \Leftarrow soil\_temp \geq 4 \, and \, son\_dist \leq max\_son\_dist \, and \, son\_dist \leq ir\_dist. \tag{3}$$

In the first case, the *fog* complex event is generated when pollution level is greater than 0.30 mg/m^3 and air is classified as *fresh*. In the second case, if the temperature of soil is less or equal than 4 °C, and the sonar distance from soil is less than maximum pre-configured sonar distance, then *PROTON* generates the *snow* complex event. In the last case, *PROTON* creates the *flooding* complex event if it comes to know the soil temperature is greater than 4 °C, sonar distance from soil is less than maximum pre-configured sonar distance and infrared distance greater than sonar distance.

As an example of Information Handler task, the tilt and the speed measures provided by sensors are used by the Information Handler to generate a *hole* event. The hole event includes information about the status of the hole (Unavoidable, easly and hardly avoidable, potentially missing) and entities (low, medium and hight shock). The *hole* event is stored in a PostGIS table for the visualization by layer, other than stored on MongoDB for the at-rest analysis.

Periodically the huge amount of data and information generated at any level are processed and made available on MongoDB and PostGIS, to prepare data for the visualization and decision maker's query by and *MAGNIFIER* DSS, in order to become more aware of the situation.

The information stored in PostGIS is to be used by GeoServer so to be displayed in the form of thematic layers. The set of layers generated at run time, together with other thematic layers gathered from geo-portals as geo-services, will allow decision maker to have an understanding of the situation. *PRACTIONIST* DSS through the management and aggregation of high-level events from *ORION*, is able to define the danger and vulnerability and possibly identify risk situations for people and/or things present within a certain area. Furthermore, *PRACTIONIST* DSS, through the huge amount of data processed can run simulations to predict the evolution of the situation.

Situational awareness gained by the *MAGNIFIER* system is made available by means of layers, alerts and recommandations to end users, who can benefit from this support to manage risks and emergencies.

4 PRACTIONIST DSS

We exploited the *PRACTIONIST* framework [8][9][10][11][12] to design and implement a goal-oriented decision support system, based on practical reasoning model [13][14][15] and able to reinforce the decision maker's awareness in regards with the current situational picture; the system reasons about the status of the environment, its internal status, and the high-level goals defined by domain experts (the decision maker or everybody who is able, starting from the analysis of the environment and its dynamics, to highlight system needs in terms of desired states and goals) in order to point out desired states and to achieve and maintain them, and to support the decision maker by suggesting him alerts and recommendations.

The *PRACTIONIST* framework aims at supporting the programmer in developing agents endowed with the following elements: *(i)* a set of perceptors able to listen to some relevant perceptions; *(ii)* a set of beliefs, which represents the information the agent has got about both its internal state and the external world; *(iii)* a set of goals, which are some objectives related to some states of affairs to bring about or actions to perform; *(iv)* a set of plans that are the means to achieve its intentions; *(v)* a set of actions the agent can perform to act over its environment; *(vi)* and a set of effectors that actually support the agent in performing its actions.

In short, we modeled situation awareness levels according to the Endsley's model [4] by exploiting some specific components of our framework:

- Perception of the elements in the environment: in *PRACTIONIST*, *perceptors* listen to some relevant external stimuli, while *beliefs* represent the information about these stimuli; we de-fined a customizable perception logic which is able to adapt to the decision maker's needs and priorities, in order to focus the perceptors' attention on specific elements of the environment, which are consequently represented by beliefs and stored in a knowledge base;
- Comprehension of the current situation: in *PRACTIONIST*, the belief logic is built upon a prolog-like language able to infer and deduct new beliefs by

means of *formulas*; that means the system is able to comprehend patterns of elements and to integrate them, creating new information and beliefs;
– Projection of future status: in *PRACTIONIST* we use some *formulas* and *belief revision rules* to identify situations that could become very dangerous in the future, in order to investigate about their projection by means of some prediction algorithms developed in R; moreover, feedbacks generated by these algorithms will be managed by *plans* designed to *(i)* understand the feedbacks, *(ii)* review the perception and attention logic, *(iii)* notify the decision maker regarding dangerous situations, *(iv)* suggest decisions to the decision maker.

We chose to represent ideal states by means of policies that the system has to apply and maintain over time; in case a policy is no longer satisfied, probably a dangerous situation is occurring, so the system will suggest decisions to the decision maker, and will require the prediction component to inquire into the projection of the situation.

Furthermore, we took advantage of the goal-orientation programming to specify ideal states of the system by means of state of affairs to be either achieved and maintained, or ceased and avoided. In *PRACTIONIST* we exploit the *Maintain* and the *Avoid* goals to define policies: in details, the first type of goal is used to define a situation to be maintained, while the last is used to model a situation to be avoided. These goals are always active and in execution, so every time they are no longer successful, some plans could be executed to restore their successful conditions. In the following figure, a general view of the elements involved to design the decision support system is shown:

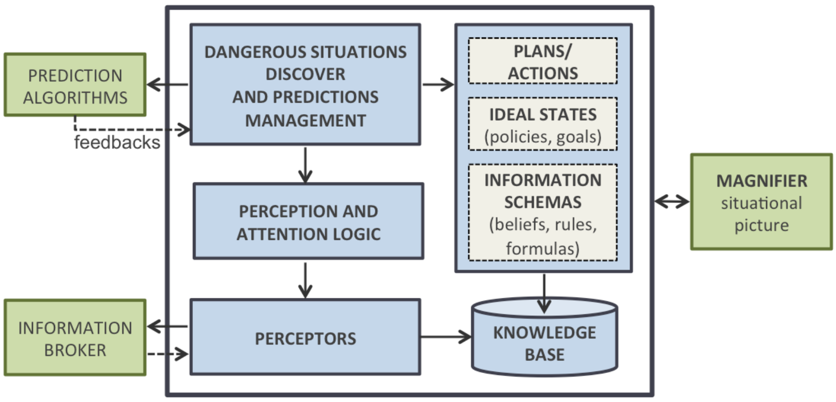

Fig. 6. MAGNIFIER DSS designed by using PRACTIONIST

Blocks with green background represent entities that are external to the decision support system boundaries, but which are included in the MAGNIFIER solution: that are the information broker (it represents the central node in charge of information routing), the prediction algorithms, and the MAGNIFIER situational picture, which includes GUIs used to show both alerts and suggested decisions.

In the case of the scenario described above, the PRACTIONIST DSS associates different vulnerability levels to infrastructures depending on several factors: their proximity to highly dangerous areas, their position, who attends the infrastructure (children, adult etc.). Furthermore, the DSS monitors communication routes, whose vulnerability level due to flooding, also depends on low quality of the itself infrastructure and on the presence of potholes.

Because of the presence of dangerous elements and vulnerable infrastructure, the DSS associates to the area and in particular to infrastructure monitored a level of risk that depends on several factors. Depending on the risk level and the presence of other negative factors which may occur (accidents, narrowing the carriageway and/or work in progress), the DSS begins assessing measures to be taken in case of danger. The monitoring of an area already begins with the presence of negative elements.

In this case the system by exploiting predictive analysis techniques, evaluates the conditions of the area in case of adverse weather conditions with different degrees; according to the prediction results, it starts showing alerts and recommendations to restore the initial conditions of the area. In case of imminent weather alert, the system begins showing an alert.

The result of the DSS reasoning process results in recommendations, which are located near critical infrastructure. As already mentioned, any recommendation is characterized by a level of danger (high, medium, low), vulnerable infrastructures, type of risk, the action to be taken to prevent the risk and finally by the risk status (warning, alert, emergency).

5 Conclusion and Future Works

In this paper we presented *MAGNIFIER*, a novel WebGIS-based platform for emergency management. Through practical reasoning mechanisms, it is able to provide decision maker with a complete situational picture, enriched with predictions, alerts, recommendations. In the mobility context, *MAGNIFIER* aims at being an innovative Decision Support System. It results from the integration of several innovative open source technologies enabling the *Intelligent Information Processing Chain* model which is showed in this paper.

In particular, *PRACTIONIST* was adopted to implement the intelligent goal-oriented decision support module, based on practical reasoning model, in order to reinforce the decision maker's awareness with regard to the current situational picture while suggesting alerts and recommendations to prevent or mitigate risks. TOREADOR environment provides support in data management for both *at-rest* and *in-motion* analysis. Other advanced technologies in Cloud Computing and Data & Content Management contexts which were adopted, are from the *FIWARE* Catalogue. *FIWARE* Catalogue contains a rich library of public, royalty-free and open source components (Generic Enablers) with reference implementations that facilitate developers' service innovation activities.

However, some further work should be done with respect to several *MAGNIFIER* open issues. Among them, our intention is to improve the DSS learning capability: when an unexpected alert arrives, the log analysis and the past situations investigation (by zooming through several time intervals) could help the MAGNIFIER system to identify cause-effect relations, thus improving the DSS performance. Finally, we are working to provide decision maker with a tool set to interact in real-time with the system so to customize the operating logic.

6 Acknowledgments

This work is partially supported by the EU's Horizon 2020 Research and Innovation Programme through the TOREADOR project (GA n. 688797).
The final publication is available at Springer via `http://dx.doi.org/10.1007/978-3-319-69459-7_6`.

References

1. Salmon, P. M., N. A. Staton, et al. (2008). "What really is going on? A review of situation awareness models for individuals and teams." Theoretical Issues in Ergonomics Science 9(4): 297-323
2. Bedny, G. and D. Meister (1999). "Theory of activity and situation awareness." International Journal of Cognitive Ergonomics 3(1): 63-72
3. Smith, K. and P. A. Hancock (1995). "Situation awareness is adaptive, externelly directed consciousness." Human Factors 37: 137-148
4. Endsley, M. R. and M. M. Robertson (2000). "Situation awareness in aircraft teams." International Journal of Industrial Ergonomics 26: 310-325
5. Geneletti, D. A GIS-based decision support system to identify nature conservation priorities in an Alpine valley. In Land Use Policy 21(2): 149-160, April 2004.
6. Jacek Malczewski GIS-based multicriteria decision analysis: a survey of the literature, International Journal of Geographical Information Science, 20:7, 703-726, 2006
7. Wooldridge, M. "Introduction to Multiagent Systems". John Wiley and Sons, 2002
8. V. Morreale, S. Bonura, G. Francaviglia, M. Cossentino, S. Gaglio. *PRACTIONIST*: a New Framework for BDI Agents. In Proceedings of the 3rd European Workshop on Multi-agent Systems (EUMAS'05) , Brussels (Belgium), 2005.
9. V. Morreale, S. Bonura, G. Francaviglia, F. Centineo, M. Cossentino, S. Gaglio. Reasoning about Goals in BDI Agents: the *PRACTIONIST* Framework. In Proceedings of the Workshop on Objects and Agents (WOA'06), Catania (Italy), 2006.
10. V. Morreale, S. Bonura, G. Francaviglia, F. Centineo, M. Cossentino, S. Gaglio. Goal-Oriented Development of BDI Agents: the *PRACTIONIST* Approach. In Proceedings of Intelligent Agent Technology, Hong Kong (China), 2006, IEEE Computer Society Press.
11. V. Morreale, G. Francaviglia, F. Centineo, M. Puccio, M. Cossentino. Goal-Oriented Agent Patterns with the *PRACTIONIST* Framework. In Proceedings of the 4th European Workshop on Multi-agent Systems (EUMAS'06) Lisbon (Portugal), 2006.

12. V. Morreale, S. Bonura, G. Francaviglia, F. Centineo, M. Puccio, M. Cossentino. Developing Intentional Systems with the *PRACTIONIST* Framework. In Proceedings of the 5th IEEE International Conference on Industrial Informatics (INDIN'07) Vienna, 2007.
13. A. S. Rao and M. P. Georgeff, "BDI agents: from theory to practice," in Proceedings of the First International Conference on Multi—Agent Systems. San Francisco, CA: MIT Press, 1995, pp. 312–319. [Online]. Available: http://www.uni-koblenz.de/˜fruit/LITERATURE/rg95.ps.gz
14. M. E. Bratman, Intention, Plans, and Practical Reason. Cambridge, MA: Harvard University Press, 1987.
15. M. Winikoff, L. Padgham, J. Harland, and J. Thangarajah, "Declarative & procedural goals in intelligent agent systems," in KR, 2002, pp. 470–481