Self-Organizing Resource Network for Traffic Accident Response

Alexander Smirnov
smir@iias.spb.su

Tatiana Levashova
oleg@iias.spb.su

Andrew Krizhanovsky
aka@iias.spb.su

Nikolay Shilov
nick@iias.spb.su

Alexey Kashevnik
alexey@iias.spb.su

St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, St.Petersburg, Russia

ABSTRACT

Traffic accidents are a common feature of the modern life. The paper proposes an approach addressing response to traffic accidents happened in a smart environment. The idea behind the approach is to self-organize resources of the environment according to the state of the situation caused by the accident. The resources self-organize a collaborative network that comprises physical devices, software services, organizations, and persons. The purpose of the resources is to undertake joint actions for accident response. The disaster response system intended for operating in smart environments has a service-oriented architecture. Some of Web-services making up the architecture are intended to model the accident situations; others model resource functionalities or bear supporting functions. Web-services that model resource functionalities are aligned against the disaster management ontology. This alignment ensures semantic interoperability of the heterogeneous resources. The alignment operation is supported by a tool that identifies similar concepts in the ontology and Web-service descriptions using a machine-readable dictionary. Response to the traffic accident illustrates main ideas described in the paper.

Keywords

Context, self-organization, semantic similarity, service-based architecture, situation modeling, traffic accident, Web-services.

INTRODUCTION

The problem of road traffic accidents is now becoming a cause for concern. The number of cars being in use is increasing dramatically. A lot of factors may lead to traffic accidents. They may be, for instance, effects of weather, seasonal variation, road and lighting conditions, the common human errors, etc. Independently on conditions under which an accident occurs an operative response to the accident situation is needed.

Research presented in this paper comes to the problem of traffic accident response proposing an approach to self-organization of resources of a smart environment according to the current state of the situation caused by an accident. A smart environment is considered to be made up of a lot of heterogeneous resources as electronic devices, computational modules, computer-aided information sources, and humans. The final purpose of the resource self-organization is their joint actions to response to the accident.

In the approach the resource functionalities are modeled by Web-services. This makes it possible to replace the self-organization of the resources with that between the appropriate Web-services. For Web-service interactions formal interface agreement defined by the technology of Web-services is used. To provide the Web-services with semantics the Web-service descriptions are aligned against an ontology.

Originally the approach presented here was developed to support decisions on disaster response planning. The ideas behind that approach were presented at the 5th International Conference on Information Systems for Crisis Response and Management (Smirnov *et al*., 2008). Research presented in this paper goes on developing the earlier presented ideas toward fully self-organized resources avoiding centralized decision making. This paper extends the previous one with regard to the issues of harmonization of Web-service descriptions and the ontology, and service-oriented architecture of the disaster response system. The applicability of the approach is demonstrated by response to a traffic accident as a kind of disaster events.

The remainder of the paper is structured as follows. The next section gives a brief overview of related research. Then the disaster response system indented to operate in smart environments and its service-based architecture are presented. Next, the main ideas described in the paper are demonstrated by a pre-starting procedure of the disaster response system and its functioning to response to a traffic accident. Some concluding remarks are summarized in Conclusion.

RELATED RESEARCH

Research related to the approach presented in this paper fall into two main scopes. The first one is research devoted to finding semantically similar concepts in structured (ontologies, Web-pages) or unstructured (texts) sources. A number of algorithms have been developed for this purpose: Hypertext Induced Topic Selection (HITS) (Kleinberg, 1999), PageRank (Brin and Page, 1998), ArcRank (Berry, 2003), ESA (Gabrilovich and Markovitch, 2007), algorithms for extraction of context related words (Karypis *et al*., 1999; Pantel and Lin, 2000), *etc*. The most close to the presented approach works on finding semantically similar concepts focus on finding such concepts taking into account lexical relations defined for these concepts in a thesaurus (e.g., Bergamaschi *et al*., 1999; Scharffe *et al*., 2007; Lambrix *et al*., 2008). Specially built domain-specific thesauri or WordNet (WordNet, 2006) serve as thesauri in these works. The approach proposed in this paper uses as a thesaurus a machine readable dictionary extracted from Wiktionary (Wiktionary, 2008). Usage of this dictionary instead of a thesaurus enables to obtain for a given concept, besides typical lexical relations defined in the thesauri, concepts associated to it in a context.

The second area closely related to the research presented here is context-aware resource management. Research in this area aims at management of sensor networks, industry of context-aware services including self-adaptable, self-configurable, self-optimized services, development of resource-aware services, building smart spaces, etc. Issues close to the subjects, the research proposed in this paper aims to, are considered in the framework of PLASTIC – Providing Lightweight & Adaptable Service Technology for Pervasive Information & Communication (PLASTIC, 2008) project sponsored by the EC FP6. Among other issues in this project a service oriented approach allowing for ad hoc context awareness in pervasive environments has been developed. In the approach context-related entities – all taking the form of contextual services – are purposed to dynamically self-organize and self-adapt to optimally exploit available, possibly heterogeneous, contextual resources at the specific time and place. Comparatively to the PLASTIC framework, distinguishing features of the approach being proposed are using Web-services to ensure interoperability between heterogeneous resources of a smart environment, involving problem-solving resources in customization of the environment functionality, and organization of a collaborative environment comprising not only physical devices and software services but organizations and persons.

DISASTER RESPONSE SYSTEM

The approach focuses on three types of resources to be organized: *information*, *problem-solving*, and *acting*. *Information* resources are resources providing information from information sources as sensors, Web-sites, databases, radio-frequency identification tags, etc. *Problem-solving* resources are computational modules, applications, services, etc. that can be used to solve problems requiring solutions in the current situation caused by a disaster event. *Acting* resources are organizations and persons taking joint actions in the current situation according to their roles.

To model the functionalities of the resources the technology of Web-services is applied. The functions (services) provided by the resources are modeled by a set of Web-services so that each service provided by a resource is modeled by one Web-service.

To the resources would be able to exchange information about their needs and capabilities and to share their knowledge, they are supported by a central application ontology (AO). The AO represents knowledge of the disaster management domain. It formalizes conceptual knowledge of the domain and specifies problems that may require solutions in situations caused by different types of disasters. In the AO problems are represented as classes, problem arguments are represented as class properties. The AO does not hold instances. The ontology classes are instantiated in a particular situation (context) by the resources of the smart environment.

Agreement between the resources and the AO is expressed through alignment of the descriptions of the Web-services modeling the resource functionalities and the AO. As a result of the alignment operation the Web-services get provided with semantics. The operation of the alignment is supported by a tool that identifies semantically similar words in the Web-service descriptions and the AO.

A situation is modeled at two levels: abstract and operational. The levels are represented by abstract context and operational context, respectively. The abstract context is an ontology-based situation model embedding the specification of problems to be solved in this situation. It is created by core Web-services incorporated in the environment. These Web-services are considered as environment components.

The operational context is an instantiated abstract context and the real-time picture of the disaster situation. Producing the operational context is one of the purposes of resource self-organization. On the grounds of the resources are represented by sets of Web-services, the self-organization of the resources is replaced with that between the appropriate Web-services. Besides the operational context producing, the Web-services are purposed to solve problems specified in the abstract context and to get acting resources to take part in activities on disaster response.

SERVICE-BASED ARCHITECTURE

In the architecture (Figure 1) of the disaster response system intended for functioning in the smart environment two types of Web-services are distinguished: *core Web-services* and *operational Web-services*.

The core Web-services are intended to creation of the abstract context and to monitoring the environment. The core Web-services comprise:

- **MonitoringService** monitors the smart environment, identifies the type of the disaster, and produces an alarm message.
- **AOAccessService** provides access to the AO;
- **AbstractContextService** creates, stores, and reuses abstract contexts;
- **ManagementService** manages Web-services to create the abstract context. It operates with the service registry where the core services are registered.

The operational Web-services self-organize a network. To make the Web-services active components capable to self-organise an agent-based service model is used. Agents are intended to negotiate services' needs and possibilities in terms of the AO and "activate" Web-services when required. The services' needs and possibilities are respectively input and output arguments of the functions that the Web-services implement. The set of operational Web-services comprises:

- **InformationSourceService** – a set of Web-services responsible for interactions with information sources of different types and for processing information provided by these sources. The following main types of information sources are distinguished: *sensors*, *databases*, *Web-sites*, and *humans*;
- **ProblemSolvingService** – a set of Web-services responsible for problem solving.
- **ParticipantProfileService** creates, modifies, and updates profiles of the acting resources or, in other words, the participants of the disaster response operation; provides access to these profiles; collects information about the participants; in a context-based way accumulates information about the participant activities on disaster response; reveals preferences of the participants;
- **ParticipantInteractionService** – a set of Web-services responsible for support of and interactions with the participants. They communicate between the disaster response system and the participants 1) providing system messages, context-sensitive help, pictures of situations, results of problem solving to the participants, and 2) delivering information from the participants to the disaster response system.

TRAFFIC ACCIDENT RESPONCE

To illustrate the approach to disaster response in smart environments a traffic accident as a particular case of

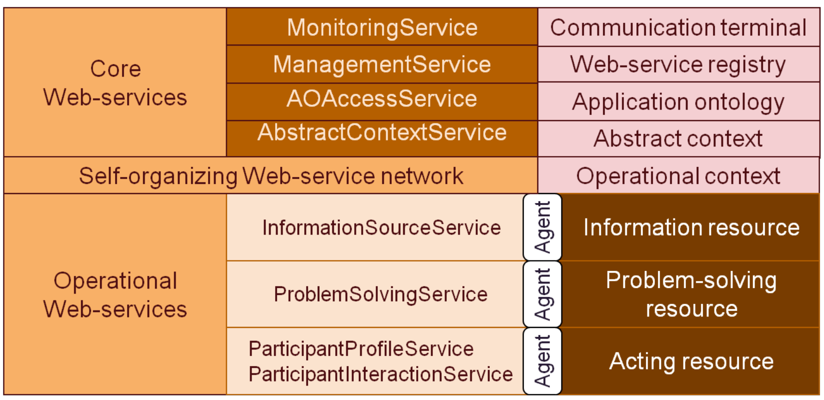

Figure 1. Service-oriented architecture

disasters is considered. The accident is caused by the ignition in the car petrol tank.

Alignment of Web-Service Descriptions and the Application Ontology

Illustration of the approach starts with a preliminary phase at that the Web-service descriptions and the AO are aligned. The alignment operation is based on discovering attributes occurred in Web-service descriptions, values of which can assign values to properties of the AO classes. It is supposed that an AO-property can take on a value provided by an attribute the name of which is semantically close to the name of this property. For discovering semantically close names a measure of semantic distance is used.

For the purpose of measuring the semantic distances between concepts containing in the Web-service descriptions and in the AO a machine readable dictionary extracted from Wiktionary (Wiktionary, 2008) is used. Wiktionary was chosen by reasons of its free use, its multilingual support, and keeping, besides lexical relations, definitions of words. The extracted machine-readable dictionary includes 1) a set of words defined in Wiktionary along with for each word 2) definitions given for this word, 3) a set of synonyms, if any, and 4) a set of associated words. Words associated to a word are considered the hyperlinked words occurring in the Wiktionary definition given for this word.

The AO is represented as a semantic network where names of classes and properties specified in the AO constitute nodes of the network. The nodes corresponding to the AO concepts are linked to nodes representing their synonyms and associated words as this is given in the machine-readable dictionary. The links between the nodes are labeled by the weights of relations specified between the concepts represented by these nodes in the machine-readable dictionary. Weight w of a relation specified between two concepts t_i and t_j is assigned as:

$$w = \begin{cases} 0,5 & - t_i, t_j \text{ are synonyms} \\ 0,3 & - t_i, t_j \text{ are associated words} \\ \infty & - t_i, t_j \text{ are the same word} \end{cases} \cdot \quad (1)$$

The values for the weights were evaluated based on the following principles:

1. Weights for the synonyms are assumed to be greater than weights for the associated words;
2. Semantic distance is proposed to be calculated as inversely proportional to weights raised to a power. The power corresponds to the path between the compared words. The longer the path the greater the semantic distance for the two different words is expected to be. To meet this expectation with reference to the way of the semantic distance calculation, a weight of the relation between two different words should be in the range (0, 1). Taken into account the first principle the weights 0,5 for the relation between the synonyms and 0,3 for the relation between the associated words are chosen empirically;
3. The semantic distance between the same words is equal to 0. Correspondingly, ∞ is assigned to the weight of the relation between the same words.

The first step in the alignment operation is parsing a Web-service description represented by the Web Service Definition Language (WSDL). The result of the parsing is a set of meaningful words found in the attribute values of WSDL-tags. If this set contains words differing from the nodes of the semantic network built for the AO, the semantic network is extended with the nodes representing the words extracted from the WSDL-file, synonyms for these words from the machine-readable dictionary, words associated in the machine-readable dictionary with the extracted words, and appropriate links. Only those synonyms and associated words for the WSDL-words are added in the semantic network, which differ from the concept the network already represents. If the semantic network built for the AO contains words found in the WSDL-file, the same words in the AO and in the WSDL-file are linked by the relation labeled "∞".

Next, nodes representing WSDL-words are checked for their similarity to nodes representing AO-concepts. As a measure of similarity semantic distance $Dist$ (2) is used.

$$Dist(t_i, t_j) = \frac{1}{\sum_S \prod_{k=s_i}^{s_j} w_k}, \quad (2)$$

where t_i - WSDL-word, t_j – AO-concept; w – weight of lexical relation existing between t_i and t_j; S – a set of paths from t_i to t_j, where a path s is formed by any number of links that connect t_i and t_j passing through any number of nodes.

After the semantic distances between the words from the WSDL-file and the names specified in the AO are calculated, experts are provided with a ranked list of semantically similar words for each word found in the WSDL-file. Based on this list the experts align related, in their judgment, attribute values in the Web-service descriptions and class properties that can take these values.

The method of identification of semantically similar words in a Web-service description and in the AO is illustrated below by an example of comparison of the description of the Web-service that registers the location of the traffic accident and the AO for disaster management.

Figure 2 shows that in the AO *Location* is an attribute (or property, what is the same) of the class *Accident*. The description of the Web-service that registers locations of accidents is given in Figure 3. The purpose is to find in the AO names of classes or attributes which could be considered as similar to the attribute `'Accident point'`. This attribute is the result (the output argument) of the Web-service function returning accident locations.

Figure 2. Specification of accident location in the AO

```
<?xml version ='1.0' encoding ='UTF-8' ?>
<definitions name='StockQuote'
xmlns:soap='http://schemas.xmlsoap.org/wsdl/soap/'
xmlns='http://schemas.xmlsoap.org/wsdl/'>
…
  <message name='getAccidentPlaceResponse'>    'output
    <part name='Accident point' type='xsd:integer'/>
  </message>
…
</definitions>
```

Figure 3. Web-service description

The set of words came of parsing the WSDL-file (Figure 3) comprises two words: *accident* and *point*. Part of the machine-readable dictionary relevant for the example in question is presented in Table 1. An illustrative piece of the semantic network built based on this table and formula (1) is represented in Figure 4. The Figure illustrates three names for classes and attributes in the AO to which from the WSDL-words paths exist: *Accident*, *Location*, and *Accident_type*. The last name is divided into *Accident* and *type*.

Word	Wiktionary definition	Synonyms	Associated words
accident	Literally, a befalling; an event that takes place without one's foresight or expectation; an undesigned, sudden, and unexpected event; often, an undesigned and unforeseen occurrence of an afflictive or unfortunate character	befalling, chance, contingency, casualty, mishap	befall, foresight, expectation, design, unexpected, unforeseen, occurrence, afflictive, unfortunate, character
point	A location or place. A point is defined by its coordinates	location, place, position, spot	location, coordinates
location	A particular place in physical space	–	place, space
type	A tag attached to variables and values used in determining what values may be assigned to what variables	data type	tag

Table 1. Machine-readable dictionary: example

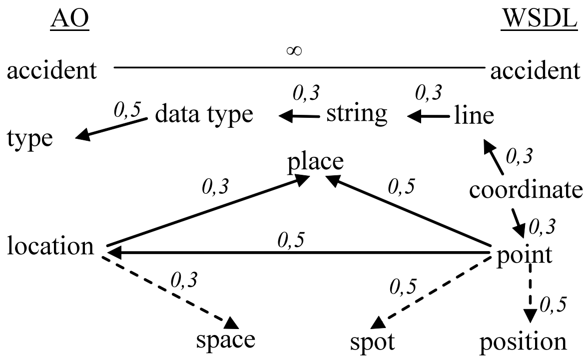

Figure 4. A piece of semantic network relevant to WSDL-attribute "Accident point"

Since both the WSDL-file and the AO contain the word *accident* semantic distance for this word is equal to 0:

$$Dist(acident, acident) = 0\,.$$

The set of paths from the WSDL-word "`point`" to the AO-word "*Location*" comprises two paths:
(1) *location* → *place* (weight 0.3), *place* → *point* (weight 0,5); and (2) *point* → *location* (weight 0,5). Thus, semantic distance between the two words is calculated as:

$$Dist(point, location) = \frac{1}{0.3 \cdot 0.5 + 0.5} = 1.54\,.$$

The set of paths from the WSDL-word "`point`" to the AO-word "*type*" in Figure 4 comprises one path:
point ← coordinate (weight 0.3), coordinate → line (weight 0.3), line → string (weight 0.3), string → data type (weight 0.3), data type → type (weight 0.5). Semantic distance between the WSDL-word "`point`" and the AO-word "*type*" is

$$Dist(point, type) = \frac{1}{0.3^4 \cdot 0.5} = 246.9\,.$$

It can be seen that for the same word *accident* the distance between the concepts *point* and *location* is much shorter than between the concepts *point* and *type*. So, the experts can align the attribute *Location* of the class *Accident* and the attribute `Accident point` contained in the Web-service description. This alignment means that the Web-service responsible for registration of accident locations instantiates the attribute *Location* of the AO class *Accident*.

Abstract Context

As soon as the **MonitoringService** has registered a disaster it sends the information about the disaster type to the core Web services. For the traffic accident considered in the paper the type of disaster is vehicle accident with fire. Based on this information the core Web services create the abstract context. This context integrates knowledge represented in the AO, which is relevant to the accident type. An exemplified piece of the abstract context (the taxonomy) for this type of accident situation is shown in Figure 5.

Within the abstract context the acting resources fall into classes *Actor* and *Job Role*. The *emergency medical service organisation* is responsible for providing emergency teams (*emergency medical technicians* in Figure 5), *ambulances*, and *rescue helicopters* for emergency medical care of injured people and / or for transportation them to *hospitals*. The *fire department* is responsible for providing firefighter brigades (*firefighters* in Figure 5), *fire trucks*, and *fire helicopters* for fire extinguishing. *Local police organisation* is responsible for providing *police officers* and *police trucks* to investigate the accident and to go through formalities. Two types of transportation are possible: *Air transportation* used by fire and rescue helicopters and *Automobile transportation* used by ambulances, fire trucks, and police trucks. Although in case of resource self-organization *emergency decision makers* are not supposed to be involved in accident response operations, the class representing this role is provided for the cases when by some reasons resource self-organization is failed.

Problem solving knowledge is collapsed in the class *Emergency Response*. This class formalizes tasks to be solved to respond to the traffic accident. They are as follows.

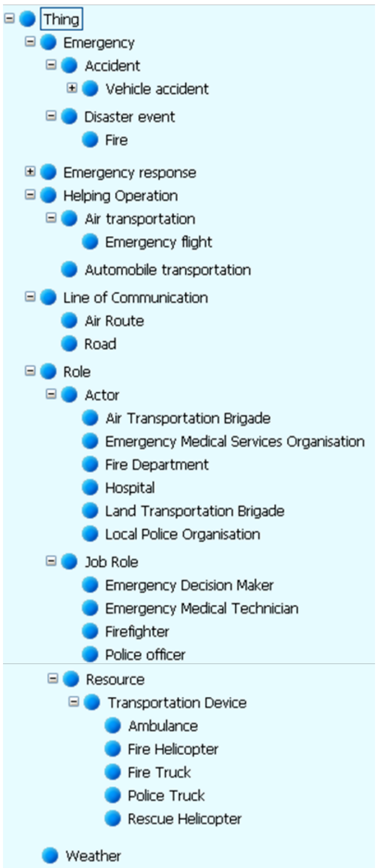

Figure 5. Abstract context: taxonomy

- The task *Quantity of emergency teams and firefighter brigades* calculates the required quantity of these kinds of groups for the response action.
- The task *Brigade availability* determines the availability of emergency and traffic police teams, and firefighter brigades.
- The task *Brigade location* determines the current location of emergency teams, firefighter brigades, and traffic police teams.
- The task *Hospital availability* returns a list of hospitals of the region, hospital addresses, free capacities, and hospital availabilities.
- The *Route availability* task determines availability of a particular route depending on its type (road, air route, etc.) taking into account (i) the types of vehicles used by the emergency teams, firefighter brigades, and police teams; (ii) the closed roads; (iii) the traffic jams; and (iv) the weather conditions.
- The *Shortest routes* task calculates the shortest routes for the appropriate acting resources.
- Joint solution for the tasks *Firefighter brigade selection*, *Emergency team selection*, *Hospital selection, Police team selection*, and *Route selection* produces a set of feasible plans of actions for the acting resources.

Web-Service Network

The network of operational Web-services self-organized to solve the tasks described above is shown in Figure 6. These tasks are solved by Web-services that model functions of information and problem-solving resources. Arrows in the figure depict execution sequences of the Web-services. Web-services that are not linked by arrows are executed simultaneously (e.g., *Firefighter brigade selection*, *Emergency team selection*, and *Police team selection*).

Web-services modeling functions of information sources use the following kinds of information resources. The current weather conditions are taken from the *sensors* and *Web-sites*. Information about the locations of the roads of the region is taken from the *GIS*. Information about emergency teams, firefighter brigades, and police teams available in the region is read from a *database*. Information about the locations of these teams and brigades is provided by the *GPS-based devices* installed on the vehicles of these

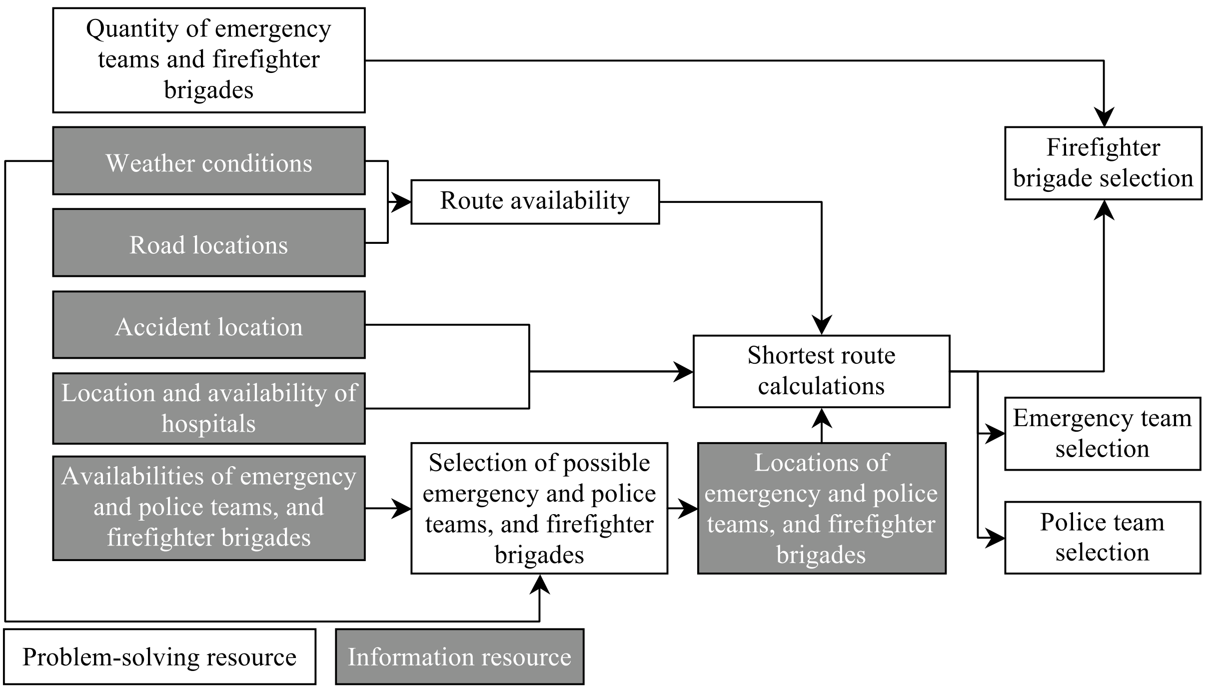

Figure 6. Network of information and problem-solving Web-services

teams and brigades. Information about the accident location, its type (traffic accident with fire), and the approximate number of victims is provided by the car *smart sensor*. Information about hospitals available in the region and their locations is read from the healthcare infrastructure *database*, hospital free capacities are provided by *hospital administration systems*.

Web-services modeling functions of problem-solving resources generate a set of feasible plans for actions. Web-services modeling functions of acting resources negotiate these plans and agree upon one. At this moment, for the negotiation a constraint-based contract net protocol (Smirnov *et al*., 2003) is used.

The traffic accident scene (operational context) and the agreed plan (Figure 7) are delivered to acting resources in the person of the leaders of the emergency teams, firefighter brigades, police teams, and the hospital administrators. They have access to the operational context through any Internet browsers (a browser supported by a notebook, PDA, mobile phone, *etc*.). The leaders of the acting resources are provided for the possibility of declining this plan. In this case Web-services modeling functions of acting resources reconsider their choice of the plan.

The network of Web-services (Figure 6) was tested in 100 Mbit local area network of 3 computers (Pentium IV

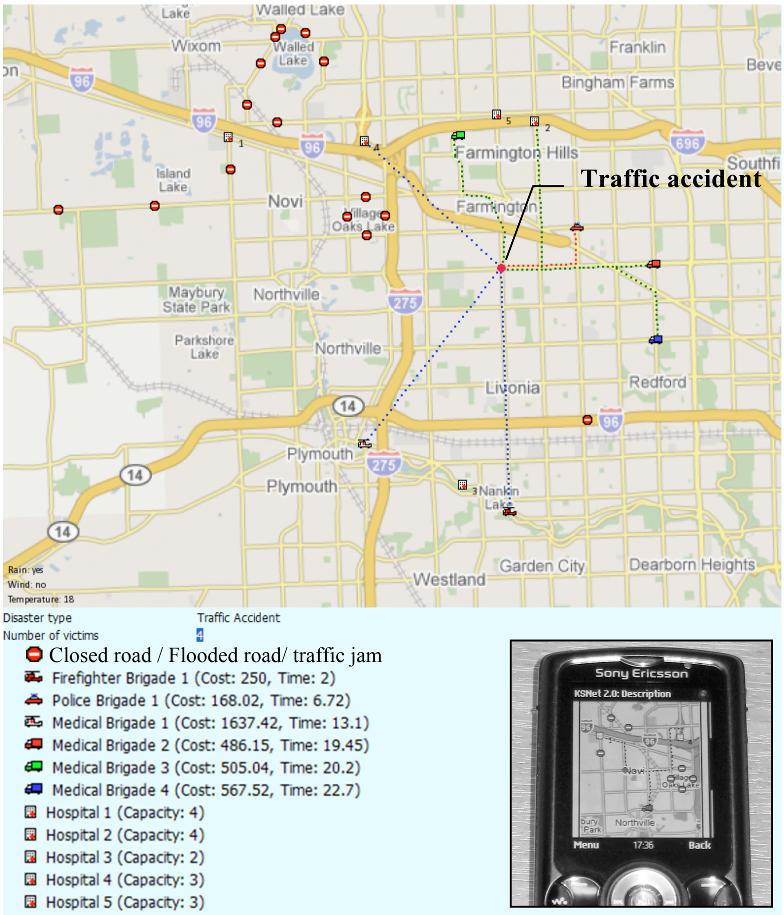

Figure 7. Operational context: a plan for actions

3.00 GHz, 1 GB RAM; Pentium Dual Core 1.8 GHz, 512 GB RAM; and Pentium Dual Core 3.00 GHz, 1GB RAM). Acting resources that were available for the accident response operations comprised 5 hospitals, 7 ambulances and 1 rescue helicopter allotted to 8 emergency teams, 7 fire trucks and 1 helicopter allotted to 8 firefighter brigades, and 2 police trucks being at the disposal of 2 police teams. Execution time from the moment the **MonitoringService** introduced to the core Web-services the type of disaster to the moment of producing the picture of the traffic accident scene took 0.0007 s. Time taken for the generation of the set of plans of action was equal to 7.8328 s.

CONCLUSION

In the paper an approach developed for disaster response in smart environments is described. The main idea behind the approach is to self-organize resources of the smart environment for their joint actions in disaster response operations. The applicability of the approach is tested through organization of resources to response a traffic accident. The prior implementation of the approach was tested by fire response actions. It can be supposed that the disaster response system built in compliance with the approach can be successfully applied to planning response actions in various kinds of disasters.

The disaster management system is built upon service-oriented architecture. Some of Web-services are used to model the resource functionalities. These Web-services self-organize a collaborative network instead of the resources. It is shown that alignment of Web-service descriptions and the application ontology at the pre-starting procedure allows the Web-services to exchange information about their needs and possibilities in terms on the ontology vocabulary. The ontology-based situation model embedding the specification of problems to be solved in this situation provides the Web-services with awareness of the problems to be solved and information needed for this. As a result, the Web-services become capable to self-organize for a common purpose.

The alignment operation is supported by a tool aiming at finding similar concepts in the ontology and Web-service descriptions. For this purpose a measure of semantic distance between two concepts has been introduced. Comparatively to the known measures, the proposed measure takes into account, besides typical lexical relations, associations existing between the compared concepts. This is believed to give more accurate results, since the concepts are compared in their broad semantic environment. The tool supporting the alignment operation can be used independently to finding similar concepts in any structured documents.

ACKNOWLEDGMENTS

The research is supported partly by the project funded by grant 08-07-00264 of the Russian Foundation for Basic Research, and project 213 of the research program "Intelligent information technologies, mathematical modelling, system analysis and automation" of the Russian Academy of Sciences.

REFERENCES

1. Bergamaschi S., Castano S. and Vincini M. (1999) Semantic Integration of Semi structured and Structured Data Sources. *SIGMOD Record*, 28, 1, 54-59.
2. Berry M. (2003) Survey of text mining: clustering, classification, and retrieval, Springer-Verlag, New York.
3. Brin S. and Page L. (1998) The Anatomy of a Large-Scale Hypertextual Web Search Engine, *Computer Networks*, 30, 1-7, 107-117.
4. Gabrilovich E. and Markovitch S. (2007) Computing Semantic Relatedness Using Wikipedia-Based Explicit Semantic Analysis, *Proceedings of The 20th International Joint Conference on Artificial Intelligence (IJCAI)*, Hyderabad, India, 1606-1611.
5. Karypis G., Han E.-H. and Kumar V. (1999) Chameleon: a hierarchical clustering algorithm using dynamic modeling, *IEEE Computer: Special Issue on Data Analysis and Mining*, 32, 8, 68-75.
6. Kleinberg J. (1999) Authoritative Sources in a Hyperlinked Environment, *Journal of the ACM*, 5, 46, 604-632.
7. Lambrix P., Tan H. and Liu Q. (2008) SAMBO and SAMBOdtf results for the Ontology Alignment Evaluation Initiative 2008, *Proceedings of the Third International Workshop on Ontology Matching*, Karlsruhe, Germany, 190-198.
8. Pantel P. and Lin D. (2000) Word-for-Word Glossing with Contextually Similar Words, *Proceedings of ANLP-NAACL 2000*. Seattle, Washington, USA, 75-85.
9. PLASTIC (2008) available at http://www.ist-plastic.org/.
10. Scharffe F., Euzenat J., Duc C.L., Mocan A. and Shvaiko P. (2007) D2.2.7: Analysis of knowledge transformation and merging techniques and implementations, Knowledge Web EU-IST Project deliverable 2.2.7, available at http://www.scharffe.fr/pub/kweb-227.pdf.
11. Smirnov A., Pashkin M., Chilov N. and Levashova T. (2003) Constraint-Based Contract Net Protocol, *Proceedings of the Fifth International Conference on Enterprise Information Systems (ICEIS 2003)*, Angers, France, 2, 549-556.
12. Smirnov A., Shilov N., Levashova T. and Kashevnik A. (2008) Web-Service Network for Disaster Management*, Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM),* Washington, USA, DC, 516-525.
13. Wiktionary, the free dictionary (2008), available at http://en.wiktionary.org/.
14. WordNet (2006) A Lexical Database for the English Language, available at http://wordnet.princeton.edu/.