Towards a Decision Support System for the Allocation of Traumatized Patients

Tim A. Majchrzak, Oliver Noack, Herbert Kuchen
Department of Information Systems
University of Muenster
tima@wi.uni-muenster.de

Philipp Neuhaus, Frank Ückert
Department of Medical Informatics
University Hospital Muenster

ABSTRACT

We present a decision support system for the allocation of traumatized patients. The assignment of patients to vehicles and hospitals is a task that requires detailed up-to-date information but has to be carried out quickly. We propose to support medical staff with an IT system. We especially encourage such a system to be used in cases of mass incidents as it is very problematic – yet essential – to provide all injured with adequate healthcare as fast as possible. Our proposal is a system based on business rules. In this paper we describe the development project's background as well as the system's requirements and some details of its implementation. Moreover, we explain an exemplary scenario to show strengths of our approach. Besides discussing related work, we draw an overview of future tasks.

Keywords

Decision support system, crisis management, traumatized patients, patient dispatching, business rules

INTRODUCTION

The assignment of patients to hospitals is an everyday job for emergency call centers. In general, several transportation possibilities are available. Vehicles are positioned at various places and have no uniform equipment. Not all hospitals are ready for all forms of injuries and most of them specialize on some forms of treatment. Therefore, the allocation of patients requires medical personnel to work quickly and precisely on complex transport decisions.

The allocation of patients to vehicles has to be decided continuously over and over again while adapting to varying circumstances. A special form of allocation is required in cases of mass incidents in which a high number of people is wounded and almost immediately require medical care. Examples are the Ramstein airshow disaster (Martin, 1990), which due to inadequate handling led to 70 dead and over 1000 wounded, and the more recent Enschede fireworks disaster (van Kamp, van der Velden, Stellato, Roorda, van Loon, Kleber, Gersons and Lebret, 2006).

We suggest supporting the allocation of traumatized patients using a geographically enabled decision support system. In collaboration with a regional network for the care of traumatized patients we developed a system that incorporates business rule management. This allows it to suggest patient allocation while being versatile and expandable.

This paper is structured as follows. In the next Section we give an overview of the project and related work. Then the decision support system is introduced. The fourth Section illustrates exemplary usage and early evaluation results. Eventually, we draw a conclusion and discuss future work.

PROJECT BACKGROUND AND RELATED WORK

Emergency call centers answer incoming calls locally in Germany. They assign vehicles and sent them to patients' locations; also, the hospital is informed. IT support is limited to information about available vehicles and status data. Decisions are made as arrangements. Occasionally, an ambulance takes an emergency physician with it or joins him

Reviewing Statement: This paper represents work in progress, an issue for discussion, a case study, best practice or other matters of interest and has been reviewed for clarity, relevance and significance.

at the destination. In this case, further decisions are usually delayed until the patient's condition has been checked.

In cases of mass incidents patient allocation is decided on the disaster site. Injured are brought to a place where triage is done (Kennedy, Aghababian, Gans, and Lewis, 1996). This means that basic medical conditions of patients are checked to determine most viable needs and prioritize transportation. Usually, a larger number of vehicles are sent to the incident's location. The order in which patients are transported is given by the triage groups they were assigned to. As the number of patients provided with medical care increases, organization is successively adapted.

A project for the implementation of a system for both routine and mass incidents patient allocation was set up as a joint interdisciplinary project between the University and University Hospital of Muenster. The main course of work can be sketched as follows. Based on the idea to support decision making in emergency call centers, existing systems were identified and the idea was refined. The next step was to specify the system. Then the system was implemented. It was first evaluated by the researchers and then initially shown to medical emergency personnel. The next step will be a close examination by healthcare professionals in general. Ideally, feedback would lead to the development of a system that will be used in emergency call centers as well as in new research projects.

An overview of decision support for the medical sector is given by Berner (2006). Techniques for building decision support systems and Web 2.0 information systems are well-understood. Our work combines various techniques in the context of crisis management and healthcare. Andrienko and Andrienko (2005) describe ongoing research on a system for "knowledge-based decision support" and "advanced methods for handling decision complexity". It is not especially focused on patient dispatching. Newer work on solving locating problems discusses patient transportation strategies (Kiechle, Doerner, Gendreau and Hartl, 2009). Liu and Palen (2009) present a survey of 13 crisis related mashups using Web 2.0 techniques and geographical support. Even though there are a lot other approaches to using geo-information support, there are none that can be directly related to our system.

THE DECISION SUPPORT SYSTEM

Requirement Analysis

The basic idea is to have a system that can be used to assign patients to transportation possibilities and hospitals. It has to be possible to manage vehicles such as ambulances or helicopters and hospitals along with their characteristics. Vehicles e.g. have a home position and specific equipment; hospitals are typically characterized by their location, capacities for various forms of treatment and specialization on distinct forms of healthcare. To accommodate the dynamic nature of emergencies, the system should be able to automatically check whether vehicles are available and what their current position is. Unavailable vehicles can be excluded from decisions. Status and position information are very helpful; consider an occupied helicopter that almost reached the destination hospital. Once it finished the current assignment and departs from the hospital, it could immediately be sent to the next patient. If the patient is at a remote place and has a serious medical condition, this could be preferable over ambulance transportation. Interfaces to hospital information systems could ensure that capacities are checked.

The information base of the system should be used to support the decisions of emergency call center staff. If an incident is reported, it should be entered into the system including details such as the patient's location, general information about her such as age or weight and her medical condition. The system should suggest a transportation possibility and also highlight alternatives. Suggestions should be similar to those medical expert would make.

A major requirement is geographical support. Assigning a vehicle depends on the patient's location, the vehicle's location and the location of hospitals suited for the patient. Routing based on the vehicle's abilities is required. Moreover, geographical support should also be available to the dispatching personnel. Showing the locations on a map helps humans to judge about transportation possibilities as well as to keep an overview of a mass incident.

Besides these general requirements a number of further characteristics are demanded. First of all, the system needs an intuitive user interface. It has to be possible to work with it very quickly. The system has to be flexible and adaptable so it can be adjusted to work with a variety of different situations. This includes the possibility to change the reasoning (i.e. how and by which logic decisions are calculated) behind the decision support. Furthermore, the system has to respond timely to make it suitable for decisions in real-time. Allocation suggestions should be calculated in not more than a few seconds since fast transportation of injured increases their survival rate (Biewener, Aschenbrenner, Rammelt, Grass and Zwipp, 2004).

Another important criterion is robustness. If single components fail, the system as a whole must not fail. Adjustable components that can be switched immediately are preferable, for example several route planners with a differing level of quality. The system also has to be able to make suggestions even in problematic cases. If a severely injured person is reported but no hospital that provides the desired level of care can be reached in time, the patient should be assigned to a nearby hospital and staff warned about his condition. Although optimal healthcare cannot be guaranteed in any case, the system should be prepared for limitations that it has to find acceptable temporary solutions for.

The system should be able to both allocate single patients and to support the management of mass incidents. Its user interface has to be suitable for displaying complex allocation suggestions. It has to respond quickly even if calculating plans for a higher two-digit number of patients. Suggestions have to be as precise as they are for the single allocation; at the same time they should reflect reasonable trade-offs. In a case were two injured require fast transportation, the first suggestion could be to assign a helicopter to the patient that is in danger of life. However, if an ambulance can reach him much quicker, he might be transported by it while the helicopter is assigned to the less injured person. Also, terrain accessibility e.g. for non-asphaltic roads has to be taken into account.

A maxim behind our idea of IS for healthcare is to support personnel rather than to relieve them from decision making. Critical decisions that could be harmful to humans should at least be checked by human experts. Therefore, an emergency plan has to be drawn up for situations where the system fails. Even with high robustness, redundancy and self-repairing functionality, dispatching of patients needs to be possible in any case. Failures of system parts must never endanger patients' lives. It also should not be possible to use the system without proper authorization.

Specification and Design Decisions

To enable future extension to mobile devices, we implemented the system web-based. The benefit of enabling physicians to view and change transportation options at the patient's location would justify additional security effort which is needed in order to deploy it outside of closed intranets.

We decided to use a system based on a rule engine. Business rule systems and the underlying *Business Rule Management* are used in corporate contexts for a long time and have proven to be feasible (Ross, 2003). The basic idea behind business rules is to model workflows i.e. sequences of actions using simple rules. Most of these rules obey a simple `if` *`condition`* `then` *`action`* or `if` *`condition`* `then` *`action`* `else` *`alternative action`* scheme. Rules are easy to understand and easy to manage. Systems of rules (*rule sets*) can be used to automate complex decisions. They are executed in a rule engine that automates inference and helps to achieve good performance.

Hard-coding the allocation process would have made a programmer intervention mandatory for even the slightest changes. By using a rule engine instead, any decision processes can be entered as rules. It is even possible to support the creation of rules by e.g. offering drop-down fields of possible actions and conditions. Rules should be maintainable by domain-experts with limited mathematical and rule-writing knowledge (Grzenda and Niemczak, 2004).

A rule system also is flexible enough to handle special needs. To give an example, special care is required for infants (Leslie and Stephenson, 1998). But only a small number of ambulances are equipped for baby rescue. This can be reflected in the rule system. If the patient is less than 2 years old and an ambulance with baby equipment is available, it would be assigned. In cases were other ambulances without baby equipment could reach the location faster, the rule system could select such an alternative while also showing the slower but better equipped ambulance. The rules have to be specified in order to reflect the complex and dynamic nature of healthcare. Rule sets have to enable suggestions that take as much information into account as possible to offer adequate patient care.

One notable design decision is the specification of a *role system*. The prototype is used by various people such as call center personnel or emergency physicians. A role system enables setting up distinct roles and aligning them with the operational permissions. Call center personnel could e.g. get the right to make transportation decisions. Entering new hospitals could be limited to a supervisor and altering the rule set could be allowed for experts only.

Implementation

The system is implemented as a 4-tier web application following the *Model View Controller* design pattern. It is backed by a *database system*, has distinct tiers for its *logic* (i.e. the computation) and the *generation of output*, and is generating pages that can be *displayed on web browsers*. As we wanted to keep the system flexible, we put special attention on its modularization. Instead of having hard-coded interfaces to third-party systems, we use wrappers.

This gives a great flexibility with regard to the third-party systems and is e.g. utilized for vehicle routing. Since the rules can be replaced as well, the system is adaptable to any changed circumstances or endeavours.

We describe the rule system by showing its exemplary usage in the next Section. In general, the system is initialized and gets an input condition. It then determines rules meeting the input conditions. Those rules are *fired* i.e. executed. The results are saved and inference is continued until any rules that could fire have been fired. The sequence of simple rules yields complex results. The actual rule execution is done by an algorithms based on the *Rete Algorithm* (Forgy, 1982) which increases inference speed dramatically. *JBoss Drools*, the rule engine employed, uses this algorithm. Additionally, *Drools Guvnor* is used to manage rules and to access them via a web-based console which helps to easily alter them. It not only checks syntactic correctness of rules but can be used to run exemplary scenarios.

With regard to mass incidents the rule system allows setting up special rule sets. In addition to conventional emergency plans special rule set can be defined. They can be tailored to the events (e.g. with regard to geographical particularities) and actively support crisis management in cases of actual incidents.

We did not only implement the rule-based system but combined it with a value benefit analysis. Rules alone cannot lead to fixed suggestions in any case. If several transportation options are available, e.g. one to a better suited hospital and one to a standard one, and transportation times differ, the system will not be indifferent between options. The actual calculation is done by assigning weights (i.e. value benefits) to the components a decision is made of. Initially, we set a high impact on transportation time. Routinely using our system might lead to readjusting weights. Of course, decisions are just suggestions. They can be reviewed – and changed – if required.

The web-based interface allows using so called mashups. Instead of implementing commonly used functionality, it can simply be integrated. We did not implement a geographical system from scratch but included it. The underlying components of the Mashups could both be accessed via the Internet or from a local server.

EXEMPLARY USAGE AND EVALUATION

The standard workflow can be sketched as follows. First, an incident is registered by the executive emergency physician and is recognized by the corresponding emergency call center. After the registration of the necessary data, the predefined rules in the system are evaluated with regard to the current data situation by the integrated inference engine. Every transportation possibility will be selected and a benefit value is assigned to it. The *best* one is allocated to the considered patient. If there is more than one patient at a time, an optimization mechanism improves the result.

Of course, the above description is simplified. Rules are arranged in a so called *ruleflow* which is a similar construct to a workflow. Rules can be part of rule groups and the order in which the different groups are evaluated can be arranged. The evaluation of the different possibilities is also mostly done with rule sets. For example, the benefit system includes that unnecessary equipment strikes negative in the benefit. Consequently, no equipment would be sent out unused and will be available for a future emergency. A very simple rule which describes this situation can be seen in Figure 1. In this screenshot the editor Guvnor is used to edit the rule set and to retrieve the result.

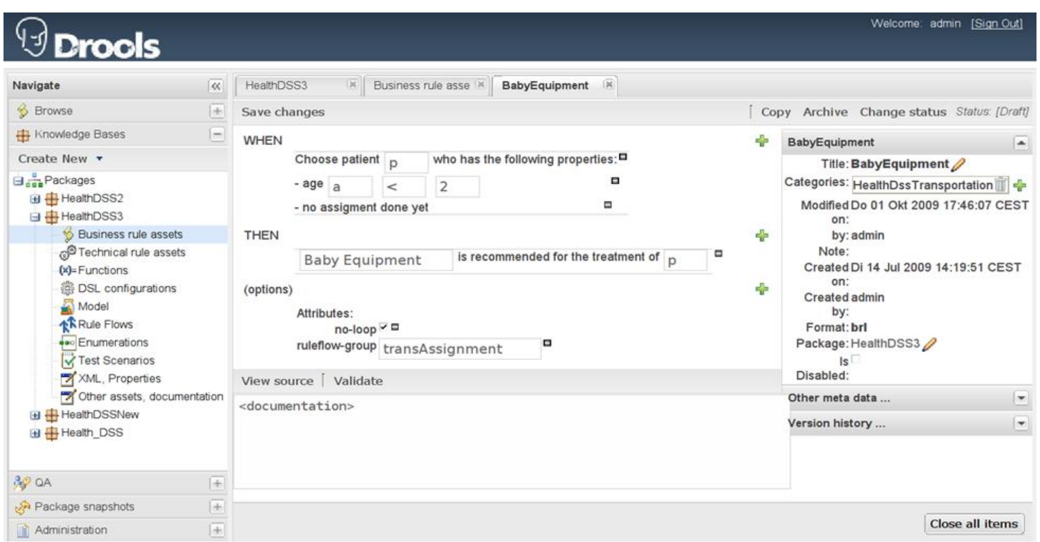

Figure 1. Usage of the rule editor delivered by JBoss to edit the rule sets

Even though our implementation is a proof of concept, early results are promising. We evaluated it in cooperation with medical personnel. While changes to parts of the system are inevitable, the general idea was received positively. For the actual implementation the rule set has to be refined. It has to be ensured that the system will not fail even in conditions where data is only available partly. It is likely to see decision support systems that are capable of handling both routine tasks and particularities such as mass incidents enter the emergency call centers.

After assigning a patient to a destination and a vehicle, results can be analyzed. The rules that were fired can be retraced. Thereby, consecutive examinations are possible and the rule set can be optimized to better suit future incidents. With an elaborated rule set the system can help to use resources more efficiently. It is possible to define different rule sets for different types of emergency incidents. Furthermore, it is possible to create new rule sets to enable the system to react to specific situations within a short period of time without changing the program itself.

We compared the execution time of an implementation using a rule set only and an implementation with some rules replaced by hard-coded calculations. It can be observed that the hard-coded implementation performs better and execution time only increases slightly with the number of transportation possibilities. However, even with the implementation that is based on rules only, execution time just increases linearly with the number of transportation possibilities. Therefore, we prefer the rule-bases approach since it offers much greater flexibility.

CONCLUSION AND FUTURE WORK

We presented a system for the support of the allocation of traumatized patients to vehicles and hospitals. It utilizes a business rule engine which makes it versatile. The system relies on routing and positioning services and provides personnel with cartographic views. It can both be used to support single patient and mass incident dispatching.

Research can be continued by refining the rules and testing more real-world scenarios. A future system could automate *any* medical decisions that are simple enough to relieve humans of them. It should provide medical personnel with information while making decisions transparent so that they can be checked and revised by humans at any time.

REFERENCES

1. Andrienko, N. and Andrienko, G. (2005) A Concept of an Intelligent Decision Support for Crisis Management in the OASIS project. In van Oosterom, P., Zlatanova, S. and Fendel, E. M.: Geo-information for Disaster Management, Springer, Berlin, 669-682.
2. Berner, E. S. (2006) Clinical Decision Support Systems: Theory and Practice, Springer, Berlin.
3. Biewener, A., Aschenbrenner, U., Rammelt, S., Grass R. and Zwipp, H. (2004) Impact of Helicopter Transport and Hospital Level on Mortality of Polytrauma Patients, The Journal of Trauma, 56, 1, 94-98.
4. Forgy, C. (1982) Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem, Artificial Intelligence, 19, 17–37.
5. Grzenda, M. and Niemczak, M. (2004) Requirements and Solutions for Web-Based Expert System, Lecture Notes in Computer Science, 3070, Springer, Berlin, 866-871.
6. van Kamp, I., van der Velden, P. G., Stellato, R. K., Roorda, J., van Loon, J., Kleber, R. J., Gersons, B. B. R. and Lebret, E. (2006) Physical and mental health shortly after a disaster: first results from the Enschede firework disaster study, The European Journal of Public Health, 16, 3, 252-258.
7. Kennedy, K., Aghababian, R., Gans, L. and Lewis, C. (1996) Triage: Techniques and Applications in Decisionmaking, Annals of Emergency Medicine, 28, 2, 136-144.
8. Kiechle, G., Doerner, K. F., Gendreau, M. and Hartl, R. F. (2009) Waiting Strategies for Regular and Emergency Patient Transportation, Operations Research Proceedings 2008, 271-276.
9. Leslie, A. J. and Stephenson, T. J. (1998) Transporting sick newborn babies, Current Paediatrics, 8, 2, 98-102.
10. Liu, S. B. and Palen, L. (2009) Spatiotemporal Mashups: A Survey of Current Tools to Inform Next Generation Crisis Support, Proc. of the 2009 ISCRAM Conference, Gothenburg, Sweden.
11. Martin, T.E. (1990) The Ramstein airshow disaster, Journal of the Royal Army Medical Corps, 136, 1, 19-26.
12. Ross, R. G. (2003) Principles of the Business Rule Approach, Addison-Wesley, Boston.