

Incident Management Information and Communication System (IMICS)



Bart-Jan Vink

1047906



Delft University of Technology
Media and Knowledge Engineering
Faculty of EEMCS
Man Machine Interaction Group
Delft, August 2009



TNO Bouw en Ondergrond
Mobiliteit en Logistiek

Abstract

Copyright © 2009 by Bart-Jan Vink, BSc. (1047906)
Man-Machine Interaction Group
Faculty of Electrical Engineering,
Mathematics and Computer Science,
Delft University of Technology,

Members of the graduation committee

Prof. Dr. Drs. L.J.M. Rothkrantz
Prof. Ir. L.H. Immers
Ir. H.J.A.M. Geers
Dr. Ir. P. Wiggers

In the past few years, traffic incident management has been a subject of interest in both politics and research. After an incident has occurred, incident management measures are intended to clear the motorway for traffic, inform other road users and treat victims as soon as possible. These measures largely constitute formal agreements, procedures and coordination efforts between all parties involved. Although the new incident management measures currently in effect in the Netherlands have improved the time required to clear an incident scene, information technology is expected to decrease the required time even further by improving communication and situational awareness and supporting analysis and evaluation of the incident response.

In this scope, a new IT system is developed, called the Incident Management Information and Communication System (IMICS). Based on an existing framework, the Java Agent DEvelopment framework (JADE), IMICS offers a blackboard like functionality through which the police, fire brigade, ambulance, traffic control centres, the shared control rooms and the Department of Public Works can update and share incident and task information both at the control rooms and at the incident scene.

In order to provide clear and unambiguous information, an ontology and an information management system have been designed that closely follow the incident management procedures.

Finally, a part of the IMICS design has been implemented and tested in a lab setting as a proof of concept. The results show IMICS improves availability and reliability of data in a lab setting.

Contents

1 Introduction.....	1
1.1 Research challenges	4
1.2 Problem definition	4
1.3 Approach.....	6
1.4 Structure of this document	7
2 Incident management background.....	9
2.1 Stakeholders	9
2.2 Incident situation.....	10
2.3 Current procedures.....	11
2.4 Current problems and project scope	13
3 Theory	15
4 Analysis.....	19
4.1 People.....	19
4.1.1 Stakeholders	20
4.1.2 Roles and responsibilities	20
4.1.3 Personal requirements.....	21
4.2 Activities	22
4.2.1 Priorities	22
4.2.2 Communication.....	23
4.2.3 Procedures.....	23
4.2.4 Roles and responsibilities	24
4.2.5 Evaluation	24
4.3 Context.....	24
4.3.1 Limitations	24
4.3.2 Non-functional requirements	25
4.4 Technologies	28
4.4.1 System decomposition	29

4.4.2 Presentation layer	29
4.4.3 Task layer	31
4.4.3.1 System architecture	31
4.4.3.2 Services and functions	32
4.4.4 Supporting infrastructure layer	32
4.4.4.1 Access points	32
4.4.4.2 Network communication.....	33
4.4.4.3 Adaptive network techniques.....	34
4.4.4.4 Data storage	34
4.5 Summary	35
5 Global design	37
5.1 Supporting infrastructure layer	38
5.1.1 Hardware.....	38
5.1.2 Software architecture	39
5.1.3 Open source middleware.....	40
5.1.4 JADE.....	40
5.2 Task layer.....	43
5.2.1 Server system	44
5.2.2 Client system:.....	45
5.3 Presentation layer.....	46
5.4 User roles	60
5.5 Summary	64
6 Detailed design	67
6.1 Server system	67
6.2 Client system.....	71
6.3 Agent communication	74
6.3.1 Agent messages.....	74
6.3.2 The IncidentMessage object	77
6.3.3 Ontology	78
6.3.4 The IMICS log.....	88
6.4 Features to be implemented	89
7 Evaluation and testing	93
7.1 Comparison with theory.....	93
7.2 System test	97
7.2.1 Test setup	97
7.2.2 Scenarios	98
7.3 Test results	109
7.3.1 Scenario 1.....	109
7.3.2 Scenario 2.....	110
7.3.3 Scenario 3.....	110

7.3.4 System performance.....	112
7.3.5 General remarks	114
7.4 Summary	114
8 Conclusions and recommendations	117
8.1 Conclusions.....	118
8.2 Recommendations	120
8.2.1 Further development	120
8.2.2 IMICS deployment.....	121
8.3 Summary	122
9 Future work.....	123
9.1 Technical improvements.....	123
9.2 Extended functionality	124
9.3 Connection to other systems	125
Bibliography	127
Appendix A	131
Appendix B	141

Chapter 1

Introduction

Increased traffic has led to increased congestion and more incidents on the Dutch road network and motorways in particular. An incident on the already clogged motorway system can result in tremendous delays in transportation, increased air pollution and even more (secondary) incidents [1]. The resulting costs to society can be significant. To minimise this negative impact, it is of great importance to clear an incident scene as quickly as possible, reducing delays and pollution and potentially saving lives. The faster response could also reduce the impact of injuries to the victims' lives. Altogether, improved incident response can save society a lot of money, making research and investments in this area worth considering.



Figure 1 - Congestion, a familiar picture for many commuters.

For this reason, the Dutch government has implemented incident management regulations in the last few years. Although the words 'incident management' can refer to many situations, ranging from dealing with a minor setback in a business deal to overcoming

large scale disasters such as hurricane Katrina, this study focuses on traffic incident management on the Dutch motorway network. In this scope, incident management officially is “the whole packet of measures intended to clear the motorway for traffic as soon as possible after an incident has taken place” [2]. In practice this means cooperation between police, fire brigade, ambulance, the Department of Public Works (Rijkswaterstaat in Dutch), recovery workers and the ANWB for safe and efficient handling of an incident.

In general when an incident occurs on the Dutch motorways, one or more observers notify the emergency response centre. The emergency response centre forwards these calls to the local shared control room, where the police, fire brigade and ambulance service share their coordination effort. Here, the information provided by the observers is processed. In this phase, the control room tries to get a complete image of the incident situation on which they base their initial response. When the initial response has been decided, the required resources are dispatched to the incident scene and other stakeholders such as salvage companies and the Department of Public Works are informed should they be required.

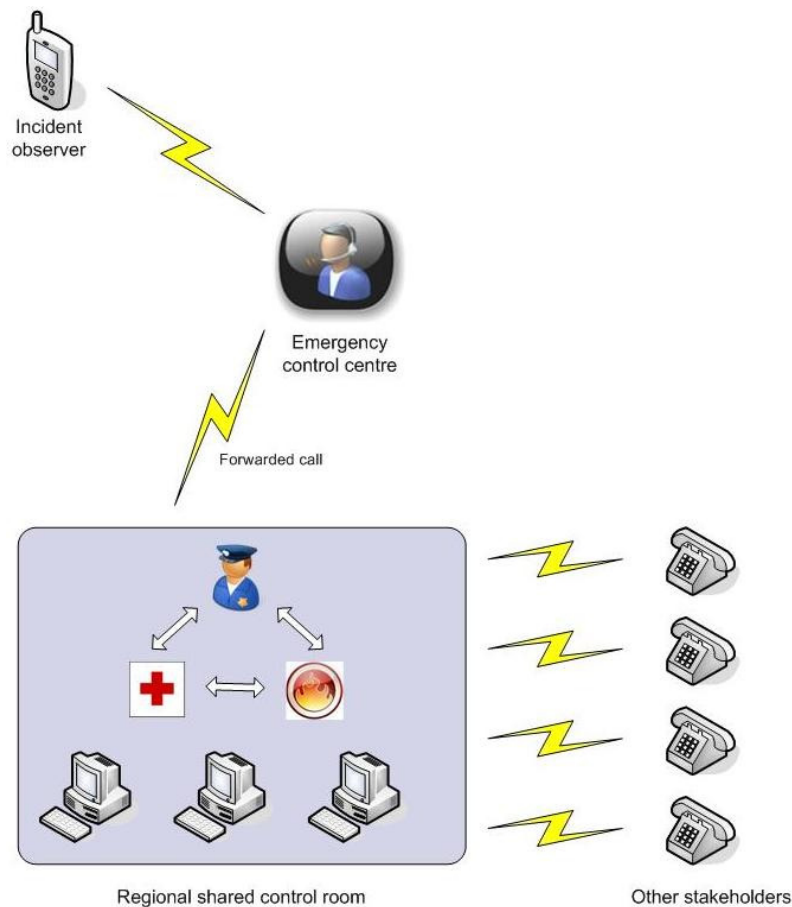


Figure 2 - Communication lines during the first phase of the incident response.

As more information becomes available at the control room or at the incident scene, new facts may emerge and the response may have to be adapted. In this way, incident management follows the OODA loop (Observe, Orient, Decide and Act) [3].

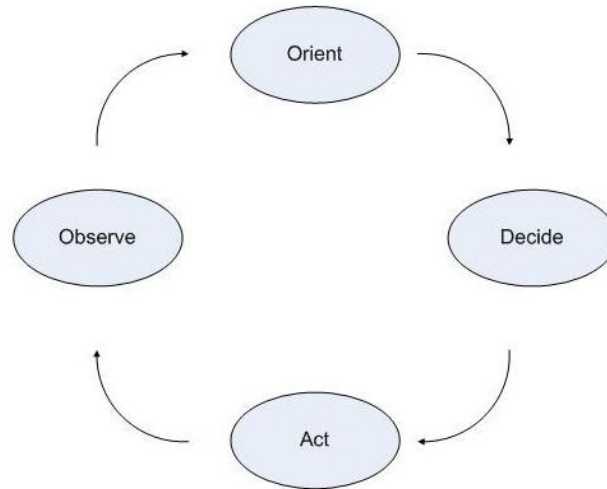


Figure 3 - The OODA loop.

The procedures of most emergency services have been adapted to each other to cooperate in an efficient way. Combined training of personnel across organisational boundaries has improved cooperation even further. Studies show that the implementation of incident management has a positive effect on the time required to reach and clean an incident site [4].



Figure 4 - Emergency services at work.

While the procedures have been designed to match seamlessly in theory, in practice miscommunications and delays occur. In stressful situations emergency respondents might not follow standard procedures or forget to provide crucial information. Conflicts might occur on the incident scene when people from different agencies disagree on authority, priorities, tasks or roles, or base their decisions on different or incomplete information. During complex incidents there is a large risk of information overflow, meaning that individuals are overwhelmed by the large amount of information, often unable to discern between relevant and irrelevant information. When communication equipment fails (e.g. when the GSM network is overloaded) it could hamper communications and the adequate handling of the incident. Since in the current situation most information is shared between agencies using conventional communication lines such as phone lines, information has to be passed on to every agency separately, allowing for communication errors along each step in the information chain. Altogether, there are many facets of incident management that can still be improved in the Netherlands.

1.1 Research challenges

To solve the problems mentioned above, coordinating the information flow and improving situational awareness, a number of challenges must be overcome. The complex situation of a traffic incident, involving many stakeholders with different tasks and responsibilities has to be modeled so the information flow can be determined.

Situational awareness is critical for the decision making phase. In order to ensure correct decision making during incident management, stakeholders must be made aware of the people and objects involved, other stakeholders' tasks and their progress. Data must be shared in a smart way to provide relevant information without overwhelming the stakeholders.

This thesis will contribute to the solution of this problem and will focus on sharing knowledge of the incident situation in an unambiguous way. To this means, an information system is designed. To test our ideas we will develop a running demonstrator. We will define this in more detail in the problem definition.

1.2 Problem definition

Recently incident management has been a hot topic in research. The last years, scientific literature has shown many proposals to improve incident response. These proposals range from strategic to operational (e.g. [5]) and from practical to strictly theoretical (e.g. [6]). What most of these studies have in common is that communication is generally seen as pivotal to adequate incident response as noted in [7]. Timely and reliable communication is important to get a complete and correct image of the situation. Both intra and interagency communication are often considered to be bottlenecks in the incident management process. In order to improve incident management even further, evaluation of the current procedures is essential [1]. Current systems do not support this adequately and improvements could be made.

Considering the complexity of incident management, the scope of this research has to be limited. Although improvements can be made to all phases of the OODA loop, this thesis focuses on the orientation phase, since the information available to the stakeholders is the basis for the following decisions and actions. Once a clear image of the incident situation has been established, decisions and the resulting actions are defined by the incident management procedures, supported by practical experience. Improvements in these phases of the OODA loop are left to experienced policy makers from the field. Although some interesting projects are under development for the observation phase, this is also beyond the scope of this thesis. More specific, this project is aimed at sharing relevant information and not at the procedures or decision making.

This study aims to improve situational awareness of the emergency services during the handling of traffic incidents on the Dutch motorways. This is achieved by improving the availability and reliability of information. By storing this information in a log, improved evaluation can be supported.

The highly standardized communication procedures between police, fire brigades, emergency medical services, the Department of Public Works, traffic control centres and salvage companies (and their respective control rooms) allow for relatively easy integration of these communication lines. By developing a new communication system, the Incident Management Information and Communication System (IMICS), which fuses all communication lines and has all relevant information readily available for those who need it, redundant communication is reduced. The near real time availability of new information about an accident enables its users to get a more complete overview of the situation and allows for a quicker and more adequate response by the emergency services. These two facts combined reduce the time required to clear an accident scene and also prevent miscommunications. The combination and logging of all data in one system supports the good evaluation of emergency services and allows the identification of key areas where improvement can be made.

Summarizing the above, this study pursuits the goals presented in Table 1.

Table 1 - Project goals.

Improve communication and situational awareness by:
1 Improved availability of data
2 Improved reliability of data
3 All information and communication joined in one device
Improve analysis, training of personnel and evaluation by:
4 Logging of data and events
5 Availability of context of information
Design, implement and test a working prototype
6 Create system design
7 Implement prototype
8 Test prototype

1.3 Approach

The first step of this project was a literature survey. By reading directives and reports about incident management from the Netherlands and the United States and many papers from the fields of incident and disaster management, insight was gained in the current incident management practice. A number of experts from the field were interviewed to clarify the procedures even further and to gain insight in the current desires of the incident management stakeholders and the projects currently being developed.

The goal of this survey was twofold. In the first place, it was meant to gain insight in the incident management procedures and background. Secondly, it was about the identification and comparison of applicable information technologies to improve communication and situational awareness between emergency services during handling of incidents. The requirements for such a system and the most promising solutions were identified and studied in more detail.

After the literature survey, a world model was derived from incident scenarios and the incident management procedures. The next step was to create a basic system design that could provide its users with the functionality required to improve their situational awareness. During this phase, experts were interviewed about more specific subjects and asked to comment on some design choices.

After discussing the general design a few existing solutions were compared that could form a basis for the system design. The most suitable solution was selected and using it as a basis, a proof of concept was implemented to show such a system is in fact feasible. During the implementation phase, completed components were tested to ensure correct functionality of the prototype.

The proof of concept was tested by running a number of scenarios with the prototype. Based on its results, some conclusions and recommendations could be drawn. Finally, during the last phases, this report was created.

These steps are summarized in Table 2.

Table 2 - Project phases.

Phase	Goals/results
Literature survey	System requirements and state of the art techniques
Studying agent technologies	Get familiar with agent technologies
Analysis of procedures and scenarios	World model (ontology)
Designing a system	Basic system design
Comparison of existing systems	Choice of best basis for system
Implementation	Running prototype
Test phase (test implementation, functionality and usability of system)	Test results, conclusions and recommendations
Writing phase	Report

1.4 Structure of this document

This Chapter provides an introduction to the subject of this thesis and explains its goals. Chapter 2 provides a more detailed background of incident management and its stakeholders. It gives a short discussion of the current incident management procedures and the practical issues that arise. In Chapter 3 important research in the field of incident management is discussed, while Chapter 4 gives an analysis of practical aspects of the designed system.

The design of the system is discussed in Chapter 5 and details of the implementation and world model are presented in Chapter 6. The implemented system was then analysed in a test setup discussed in Chapter 7. The conclusions of this study and test are described in Chapter 8, followed by some recommendations. Because of the limited time available the scope of this project was limited. Only part of the system envisioned was actually implemented and many ideas were put aside for future studies. In Chapter 9 some of these concepts and how they could complement the designed system are described.

Appendix A gives an overview of some of the techniques reviewed during the literature survey and in Appendix B the results of the held interviews are presented.

Chapter 2

Incident management background

Incident management is the whole range of measures to clear the Dutch motorways for traffic as soon as possible after an incident has taken place. In practice, this means cooperation between police, fire brigade, ambulance, the Department of Public Works (Rijkswaterstaat), recovery workers and the ANWB for safe and efficient handling of an incident [2].

This chapter gives a short overview of the current state of affairs of incident management (IM) on the Dutch motorway network and describes the scope and background of this thesis.

2.1 Stakeholders

Depending on the type and complexity of the incident, different services and organizations are involved. Some stakeholders are not involved in the first line of emergency response and not all stakeholders are involved in every type of incident. Table 3 below gives an overview of the stakeholders involved in incident management in the Netherlands, divided into primary stakeholders and secondary stakeholders. The Dutch names are shown in brackets.

Secondary stakeholders are not directly involved in the clearing of an incident scene, however they can get involved after an incident has been cleared. Since the project focus is on improving the incident response, secondary stakeholders are not included in the system design for now. In fact, the system designed is primarily meant to be used by the police, fire brigade, ambulance, traffic control centres, the shared control rooms and the Department of Public Works. This choice was made because these emergency services form the basis of traffic incident management. This does not mean the other stakeholders are less important to good incident management (in fact the other primary stakeholders are vital to the handling of complex incidents) but it was a necessity to limit the scope of the project considering the limited time available.

Future projects could build on this project to include the other primary stakeholders and in the future it might be desirable to add functionality which allows secondary

stakeholders to access some of the information in the system. The latter might raise security and privacy issues though. Chapter 5 discusses this matter in more detail.

Table 3 - Primary and secondary stakeholders.

Primary stakeholders	Contractors (for road repairs, pollution, viewer screens) Department of Public Works (Rijkswaterstaat) Emergency medical service (ambulance, hospital) Fire brigade IM recovery dispatch centre (CMI ¹) IM lorry recovery dispatch centre (CMV ²) Local authorities Lorry Incident Management Foundation (STIMVA ³) Lorry Salvage Consultant (STI ⁴) Netherlands Incident Management Foundation (SIMN ⁵) Police Traffic control centres (regional and national) Towing companies (and ANWB) The Dutch emergency response centre
Secondary stakeholders	Environmental protection groups Insurance companies Other traffic on the road Transporters Auditors Department of Public Works

¹ Centraal Meldpunt Incidenten

² Centraal Meldpunt Vrachtwagens

³ Stichting Incident Management Vrachtauto's

⁴ Salvage Transport Incident

⁵ Stichting Incident Management Nederland

2.2 Incident situation

Clearing an incident involves work on different locations. The first obvious location is the site of the incident, which in this study is somewhere on the Dutch road network. Although incidents can occur at any type of road, the Dutch incident management procedures are currently only implemented on the motorways. Therefore the main focus of this thesis is on the Dutch motorway network. There is still a large difference between individual incidents on the motorway, since an incident can be at the motorway shoulder, but can also involve several lanes up to a completely blocked motorway or intersection in a more severe case.

Another part of the incident management process takes place in the shared control rooms. Usually the first notification of an incident is received here and the first information about the scene is gathered here as well. The control room is currently responsible for notifying the emergency services and distributing information to the proper authorities. Processing and filing of information usually takes place at the shared control rooms. Especially during the first phase of incident response, the emergency services can also be on the road on their way to the incident location. While driving to the incident location, emergency services can communicate with the control rooms to gain extra information on the incident.

To facilitate communications between emergency services, their personnel have some tools at their disposal. In the first place, they are supported by the operators in the control rooms. Secondly, most people have access to a GSM allowing them to quickly call any known number, like for instance a colleague. Finally, in the last years, the C2000 communication system has been deployed. This is a secure communications network through which the police, fire brigade and ambulance service can communicate with each other. In special situations, a dedicated antenna can be placed at the incident scene to guarantee a reliable communication network.

One more thing should be kept in mind when dealing with incident management, incidents are inherently chaotic; this means that the situation during the emergency response is also chaotic. People involved with incident management often work with incomplete information that could be ambiguous or wrong, acquiring new information during the process. The work often involves a stressful situation with injured or even dead persons and a time pressure to rescue victims. Clogged roads often add to the pressure of the situation. This is especially important since most incidents take place during times of high traffic. This chaotic nature requires a high flexibility and good coordination. Adequate training of personnel is also very important for a quick and safe emergency response.

2.3 Current procedures

To enable a well coordinated response to crises, the Netherlands have been divided into 25 safety regions, in which different organizations cooperate. These regions are shown in Figure 5. Each safety region has a shared control centre in which police, fire brigade and ambulance service cooperate. Although the centre is shared, operations are strictly separated. In practice, this means that the organization that receives the first notification of an incident will have to inform other organizations if their assistance is required.

The required response in case of a traffic incident is clearly structured through interagency agreements, which are stated in the new IM directive [9]. Although no two accidents are the same, most required procedures are highly standardized. When an incident occurs on the Dutch motorways, usually a call is made to the emergency number 112 (equivalent to the American emergency number, 911) by one of the persons involved in the incident or by someone close to the incident. This call is received by the emergency response centre which is located in the Dutch town Driebergen. Depending on

the location of the incident, the call is forwarded to the shared control room of one of the 25 safety regions. It is also possible that one of the emergency services finds an incident when patrolling the motorway. In that case, they are responsible for reporting the incident to the shared control room.



Figure 5 - The 25 Dutch safety regions [8].

At the shared control room, following a standard procedure, as much information as possible is gained. Depending in the kind and severity of the incident, the necessary emergency services are notified. These emergency services then dispatch the required people and equipment. The shared control room is responsible for maintaining communications with all involved stakeholders and keeping them informed of any developments during the handling of an incident.

At the incident scene, the incident response is coordinated in an informal cooperation in small scale incidents, and by the formal COPI team in complex incidents (Coördinatieteam Plaats Incident in Dutch). COPI is formed by people from the police, fire brigade, ambulance service and the Department of Public Works (Rijkswaterstaat).

Depending on the type of incident and the emergency services available at the moment, one member of the COPI team is in charge and responsible for the correct handling of the incident. Critical decisions however, are usually taken as a team effort.

One of the major improvements of the new incident management directive deserves a bit more attention. About 80% of all incidents on the Dutch road network involve material damage only, without a need for medical assistance or police investigation. In the new directive, vehicles involved in these incidents are towed away immediately in order to return traffic to normal as quickly as possible. For this reason, the police control room contacts the IM recovery dispatch centre (CMI, Centraal Meldpunt Incidenten, in Dutch) immediately after an incident has been reported, without verifying the incident first. The CMI, in turn, notifies a salvage company as soon as possible. Although there is a risk that a salvage company is called to action for nothing, this measure saves about 15 minutes per incident on average [4]. The costs of unnecessary dispatches are covered by the Dutch Department of Public Works.

If the incident involves trucks or other heavy vehicles, a police patrol vehicle is dispatched first to assess the situation. If required, the IM lorry recovery dispatch centre (CMV, Centraal Meldpunt Vrachtautoberging in Dutch) is contacted and the CMV in turn notifies a special salvage company with heavy equipment. This procedure saves about 60 to 90 minutes per incident on average [4].

In case of incidents involving heavy injuries or serious traffic violations the police has to gather evidence before the scene can be cleared. In case of injuries or trapped passengers the ambulance service or fire brigade may be required as well. To streamline the relief operations, a clear set of agreements has been laid down in the incident management measures. A complete overview of these procedures can be found in [9].

2.4 Current problems and project scope

Despite the well-regulated procedures, miscommunications and delays do occur in practice. In stressful situations emergency respondents might not follow standard procedures or may forget to provide crucial information. Conflicts might occur on the incident scene when people from different agencies disagree on authority, priorities, tasks or roles, or base their decisions on different or incomplete information. When communication equipment fails (e.g. when the GSM network is overloaded) it could hamper communications and the adequate handling of the incident. Since in the current situation most information is shared between agencies using conventional communication lines such as phone lines, information has to be passed on to every agency separately, allowing for communication errors along each step in the information chain and large communication overhead.

To cope with these problems, a long term policy has been laid out that describes in which areas improvements should be made [1]. One of the recommendations made in this report is the development of an incident management information and communication system (IMICS). In this scope, TNO Bouw en Ondergrond allowed me to do my research

assignment and my final graduation project. A design for IMICS must be created and a prototype must be developed as a proof of concept. As the following chapters discuss, the designed system focuses on improving the information chain. By combining all relevant information in one integral system, communications and coordination should be simplified, leading to an improved incident response.

Chapter 3

Theory

In current research, there are some general principles that are seen as the basis for successful emergency response information systems. As traffic incident management can be seen as a special case of emergency response, to a certain extent these principles also apply to the design of a traffic incident management system. In their very important survey paper [10], Turoff et al. lay out these general principles in great detail. It would take too far to discuss these principles in detail here, but the main points will be highlighted here. In Chapter 7 the system is reviewed in the light of these principles as well as tried in an actual test with a realistic scenario.

Although traffic incident management is a much more limited domain than general emergency management, the following design premises still apply to this domain. First of all, in order to prevent information overload, relief workers should only receive relevant information. It is also important to understand what actually happened during the incident and to be able to review this information to improve the incident response. Because of the dynamic situation of incidents, it is important that the system can be reconfigured, for example by changing priorities and filtering options. It should also be possible to transfer roles or tasks to other persons, it should therefore be possible to check who is available. Since critical decisions require the best possible up-to-date information, providing this information should be facilitated by the system as much as possible. Table 4 summarizes these design premises.

Table 4 - Design premises.

- | |
|--|
| <ol style="list-style-type: none">1. Prevent information overload (show relevant information only)2. Improve situational awareness (provide and review up-to-date information and context)3. Support configuration (adaptable priorities and filtering)4. Support role and task transfer (monitor and manage tasks and availability of personnel) |
|--|

In their paper, Turoff et al. describe 12 fundamental roles that should be supported by an emergency management system as shown in Table 5. Since some of the tasks will be handled by the control centres and their existing IT systems, not all roles will be directly supported by IMICS.

Table 5 - Fundamental roles in emergency management.

<ol style="list-style-type: none"> 1. Request resources (people and equipment) 2. Allocate, delay or deny resources 3. Report and update situation 4. Analyze situation 5. Edit, organize, and summarize information 6. Maintain resources (logistics) 7. Acquire more or new resources 8. Oversight review, consult, advise 9. Alert all with a need to know 10. Assign roles and responsibilities when needed 11. Coordinate among different resource areas 12. Priority and strategy setting (e.g., command and control)

Following the premises and roles, Turoff et al. state a number of general design principles as shown below.

Design principle 1 - System Directory: The system directory should provide a hierarchical structure for all the data and information currently in the system and provide a complete text search to all or selected subsets of the material.

Design principle 2 – Information Source and Timeliness: In an emergency it is critical that every bit of quantitative or qualitative data brought into the system dealing with the ongoing emergency be identified by its human or database source, its time of occurrence and its status. Also, where appropriate, by its location and by links to whatever it is referring to that already exists within the system.

Design principle 3 – Open Multi-Directional Communication: A system such as this must be viewed as an open and flat communication process among all those involved in reacting to the disaster. There is no way to predict what information is going to be needed and by whom. People often change roles and carry out processes they were not originally scheduled for. In crises the hierarchical organization reduces to a flat one and an increased number of decisions are made at lower levels [11].

Design Principle 4 - Content as Address: the content of a piece of information is what determines the address. If certain pieces of content are found to be of interest to a common group of users, this information should be linked and be accessible by their users, creating a subgroup of common concern.

Design principle 5 – Up-to-Date Information and Data: Data that reaches a user and/or his/her interface device must be updated whenever it is viewed on the screen or presented verbally to the user. If data that is currently being viewed is updated, a

notification should be issued to the viewer. When a user marks data it might be a good idea to add certain levels of interest, so a user can choose whether to always be alerted in case of an update, or not when extremely busy.

Design Principle 6 - Link Relevant Information and Data: An item of data and its semantic links to other data are treated as one unit of information that is simultaneously created or updated.

Design Principle 7 – Authority, Responsibility and Accountability: Authority in an emergency flows down to where the actions are taking place. This reinforces the need for everything to be available on the scene. Those in a remote “command centre” ensure that individual decisions being made in the front lines do not result in negative cumulative result. The upper chain of command is an oversight and exception operation. They need to be aware when an action is inconsistent or in conflict with other actions elsewhere. Authority for action has to exist with the front line roles and higher levels should deal with monitoring, oversight, resource availability and threat assessment. A wrong action may take place unless someone with oversight authority halts it. Clear accountability is needed. Who is taking what actions? It should be clear to all when a conflict occurs and how it is handled.

Design principle 8 – Psychological and sociological factors: Encourage and support the psychological and social needs of the crisis response team. Social relationships must be allowed and the system should support tier maintenance and development (For example by “coffee break” conferences and chat rooms to develop quick trust). This reduces stress levels and improves handover of roles and dealing with oversight. The system must support a “team spirit”. People must develop trust in each other. The most challenging part of the interface is the reduction of information overload. The user should be able to adapt the system to his method of cognitive problem solving and not impose one rigid approach for all users. Rigidity in the interface will inhibit creativity or improvisation.

Turoff identifies another important issue; communications between different agencies by people who have not worked together probably contains a lot of ambiguity. As a result people often think they communicated and agreed when in fact they did not. This is the common “ambiguity of consensus” that occurs in group meetings. As will become clear later in this report (Section 6.3.3.), the design of a consistent ontology which describes the objects and relationships in a traffic incident setting minimises this ambiguity.

Chapter 4

Analysis

After careful consideration of the current incident management situation, a list of different sides of the communication in incident response has been constructed. Using an analysis technique from the principals of Human Computer Interaction called PACT (People, Activities, Context, Technologies) as described in [12], an overview of aspects taken into account is created. Following the PACT method, this chapter describes these aspects, leading to the basic questions and choices underlying the design of IMICS.

Section 4.1 describes the human side of incident management and the designed system; for whom is the system designed, what are their roles and responsibilities and what are their requirement regarding the system.

Next, Section 4.2 discusses the priorities of the activities incident management comprises, and the way different tasks and responsibilities are communicated between organisations. Finally evaluation of the activities is shortly discussed.

In Section 4.3 the incident management context is discussed. The situation at the incident site and the way the incident response is structured impose some requirements on the designed system.

Section 4.4 elaborates on the different parts of the system and the technologies that support the system. A brief discussion of all technologies considered for the system during the first phase of the project is given in Appendix A.

4.1 People

Since the design of a system is largely dependent on the intended users, it is important to know their tasks and their needs. This section describes the intended users in more detail.

4.1.1 Stakeholders

As stated above there are many stakeholders involved in incident management. Table 3 in Section 2.1 gives an overview of the organizations involved. Not everybody involved in incident management needs direct access to the system. It is important to thoroughly analyse for which stakeholders the system should be designed, who has access to the information in the system and who can adjust or add information. Is the system designed for all organizations involved in incident management or does it focus on primary stakeholders?

Since the majority of the incident management process takes place at the incident scene and involves the police, fire brigade, ambulance service and the road inspector from the Department of Public Works, the system design focuses on these stakeholders primarily. The shared control centres of the emergency services and the traffic control centres are also crucial to the adequate handling of an incident and therefore also have access to the system. Other stakeholders directly involved in the process can be called to aid by phone, or are contacted by the control centres. Stakeholders not directly involved in the recovery process will also receive the required information through the old means for now, although the system could provide limited access to them in the future. Table 6 gives a summary of the organizations the system is designed for. The system is designed in such a way that providing access to other stakeholders in future versions is relatively easy. To keep the information in the system consistent, there should be a clear responsibility for maintaining an overview and keeping information up to date. This will be discussed in more detail in the next section and in Chapter 5.

Table 6 - Stakeholders included in the current IMICS design.

Police
Fire brigade
Ambulance service
Road inspector
Shared control centres
Traffic control centres

4.1.2 Roles and responsibilities

Since all organizations involved in incident management have different tasks, their roles and rights in the system could be different too. Should the system provide different roles and rights or should every involved person have full access rights? If not, should it be possible to reassign roles or tasks to others? Can one person fulfil multiple roles? Since the involved organizations have their own respective system in place, should the system facilitate access to these systems and should the system be accessible from other systems?

The roles and responsibilities in incident management on the Dutch motorways are defined in agreements between emergency organizations [9]. It is possible however to deviate from the rules if the situation requires it. During the incident response, information is exchanged between people and organizations to ensure proper actions are

taken. Different roles may require and provide different information. One of the key issues in incident management is providing personnel with the right, up-to-date information. To ensure correct and consistent information, one person should be responsible for the management of the data acquired on scene. This task is assigned to a director role. The director would be responsible for appropriate incident response and would therefore be the most suitable person to fulfil this task. Because of the importance of this role, it must always be fulfilled. Section 5.4 will discuss the direction role in more detail.

Although the tasks of the involved organisations are quite different, their need for information is similar in the sense that they all normally gather information through the same procedures [9]. In the envisioned system, they can all access the system in the same manner. It is possible to create different roles for the different organizations, however for now the design simply allows the stakeholders mentioned above to access all information on the incident in the same manner. As Section 5.4 discusses, a distinction is made between three roles with different responsibilities, but these can be fulfilled by all stakeholders. In future versions, it should be easy to diversify the different roles for different stakeholders if desired. For example, more sensitive information, like personal information of the people involved in resolving an incident could be accessed by people of their own organisation that have a need for it, but be inaccessible to the rest of the users. Moreover, although the system is designed as a stand alone application, the design will allow it to be coupled to other systems with as little effort as possible. Once more, in Chapter 5 the system design will be fully discussed.

4.1.3 Personal requirements

Now that the intended users and their roles have been determined, it must be determined if they impose any further restrictions on the system design.

Personal needs or limitations

What kind of people accesses the system has a large impact on its design. Are there any specific needs or limitations to the intended user group? For example, should the system have special features for the physically challenged? Are specific means of communication desired and should alternatives be available? All persons involved in incident management are healthy individuals since this is required for the job. Therefore the users themselves have no special needs regarding the system.

Ease of use and required training

As with any IT system, a system supporting incident management should be easy to use and easy to learn. Especially in critical situations such as incidents, it is important that the system provides clear and unambiguous information. The same terminology should also be used among all users, preventing miscommunication. The system is designed with the intended users in mind and should be tested in a field practice. Training of personnel should ensure good knowledge of the system and its features and prevent incorrect use. Since regular training is mandatory for the emergency services this should not affect normal work too much.

Evaluation

To allow further improvements to the incident response, it is important to evaluate previous incidents. People should learn from previous mistakes and recurring issues should be resolved. In order to support evaluation, the system will record all entries and changes made in the system. It will also provide access logs, tracking system use (e.g. tracking access times and the persons that have accessed the information).

4.2 Activities

The purpose of the designed system is to support the incident management activities. It is important that use of the system does not interfere with the primary tasks of the involved personnel. It is not hard to imagine that ambulance personnel trying to save a life should not be distracted by trivial activities required by the system. This section describes the activities the system should support, while Chapter 5 discusses how this is achieved.

4.2.1 Priorities

Many tasks have to be carried out during incident response, ranging from clearing debris to medical assistance. A clear list of priorities has been set in mutual agreements between emergency services [2]. This is shown in Table 7.

Table 7 - Incident management priorities.

1: Safety of emergency personnel

The main priority in incident management is the safety of emergency personnel. While the lives of victims might be at stake, it is important not to unnecessarily endanger the lives of emergency personnel.

2: Traffic safety

The second priority is traffic safety, it is important that the incident and the incident response do not endanger other traffic.

3: Aiding victims

After the safety of the emergency personnel and other traffic has been ensured, aiding the victims is the next priority.

4: Sustaining traffic flow

When lanes are blocked, it is important to clear them as quickly as possible to prevent congestion.

5: Protecting cargo and vehicles

If possible, cargo and vehicles involved in an incident should be recovered.

Securing the incident location is important since there is a high risk of secondary incidents. Ensuring traffic flow is less important than aiding victims, however salvaging cargo and vehicles on the motorway is less important than traffic flow. Therefore it is

important to clear the road as quickly as possible. In extreme cases (e.g. involving medical equipment transport) the cargo can be too expensive and salvaging it is deemed more important than ensured traffic flow.

These aspects form the crux of incident management, it is about protecting and saving lives and ensuring safe and reliable traffic flow. Keeping these priorities in mind, the procedures discussed in Section 2.3 have been devised to secure the incident scene. To ensure clarity on the procedures, the proposed system will include a prioritized checklist of the required tasks. When certain tasks have been completed, or the situation requires a change in priorities, the responsible person should enter this information into the system and all incident respondents will receive a notification that an updated task list is available. Once more, it is important to keep in mind that sharing information, whether by phone or by making use of the proposed system, is usually not a first priority. Although unambiguous communication is essential for the incident response, the system may never distract from, or interfere with, the tasks at hand.

4.2.2 Communication

Communication is an essential part of incident management in distributing information among emergency services and decision making. During incident response, control rooms call and brief respondents to be dispatched to the incident location. On scene, respondents provide new information or request backup or special equipment from the control rooms. At the site, personnel often discuss the best approach. In short, communication is one of the most important parts of incident management.

The main focus of the proposed system is improving the communication between emergency services and their situational awareness. This comprehends the availability of information to the relevant stakeholders. The designed system provides information from its suppliers to the rest of the respondents. When important updates are made, a warning will be issued, informing other respondents. The improvement in comparison with the existing situation lies in the fact that new information is immediately made available to all directly involved stakeholders. Redundant communication is reduced and all information is easily accessible through one system.

4.2.3 Procedures

Although no two incidents are the same, work at incident sites follows strict procedures to enhance safety as mentioned in Section 2.3. The well structured tasks also prevent mistakes. It is important that the system follows and supports these procedures, yet it should also support a certain level of flexibility. In no way may the system hamper the incident response. So the system is designed to follow the standard procedures, but some flexibility is allowed. For example, it should be possible to reassign tasks and change priorities if the situation requires it. The availability of information to all stakeholders directly involved and the fact that tasks can be reassigned, facilitate a degree of flexibility that allows the respondents to cope with changing situations and priorities.

4.2.4 Roles and responsibilities

As stated before, each involved organisation has its own tasks and responsibilities, which are planned according to the abovementioned priorities. These activities are, however, not independent. It is therefore important that everyone at the incident scene is aware of the other organisations' tasks. As will be discussed in Chapter 5, the system provides information on the required tasks and their dependencies and provide an indication of the current progress.

To ensure the correct actions are taken during incident response, it is important that all information in the system is up to date. Every involved person is responsible for keeping track of his own tasks. In the end however, the director is responsible for updating the information in the system and ensuring the provided information is correct. It may be the case that an incident respondent needs additional or more specific information. It may therefore be necessary to send a request for additional information to the director. This feature should be supported by the system. If required, the system could also ask for additional or more specific information itself.

4.2.5 Evaluation

Just as the individual persons involved should evaluate their work, procedures should also be evaluated and adapted if necessary [1]. For example, should specific roles request a certain specification of information on regular basis, the procedure might have to be adapted to always include that information. Just as personal evaluation, evaluating procedures requires information. Once more, changes to the information in the system will be stored for later evaluation, including time and context information.

4.3 Context

Activities always happen in a context, the context and activities exert influence on each other. To prevent finding limitations to the system in practical use, it is important to take working conditions into consideration in the design phase [12]. This section describes the influence of the context on the system design.

4.3.1 Limitations

Due to the conditions at the incident scene there are some limitations to the system. Since generally there are no power outlets and fixed network connections available along the motorway, system mobility is important. Besides incident respondents will have to move around at the incident scene. To remain highly mobile, system size will have to be limited. As a result, battery life and computing power are limited. It is possible that at a remote incident scene there is no or limited network coverage for GSM. Although network coverage has become quite reliable in normal situations, it is still possible that the network fails due to special conditions. The working conditions also pose some limitations to the system. For example, in winter it might be hard to use a small keyboard because of cold fingers and during heavy showers it should still be possible to use the

system without risking water damage. Visibility might also be an issue when working at night or in very bright sunlight. The noisy environment of the motorway also poses limitations to the system.

The situation in the control rooms, although perhaps more chaotic at times, is comparable to a standard office environment. Control room personnel can work at designated workstations using standard desktop computers and fast, fixed line, network connections. If required they can be equipped with very large screens that can be read by many users at a time during crisis situations. Since people at the control rooms can be working on multiple incidents at a time, they must be able to access all of these incidents simultaneously.

As became clear during an interview at the traffic control centre in Rhoon, The Netherlands, the personnel at the control rooms can be very busy (see Appendix B). At the incident scene, entering data may not distract from the tasks at hand as discussed in Section 4.2.1. It is important to keep these facts in mind during system design, time pressure requires entering data to be straightforward and fast.

4.3.2 Non-functional requirements

The volatile nature of incidents poses strict requirements on any system used in incident management. Since the prototype will not be a fully developed system, some concessions will have to be made. This section describes which non-functional requirements are met by the prototype and why some of them will not be met. The most important requirements are shown in Table 8 below.

Table 8 - Non-functional requirements.

Reliability

The system should never fail and should be usable in any situation which could be expected during incident management. For example during situations of high demand, the system should still be able to function properly and latency times should not exceed acceptable values. Even in unexpected situations the system should still work properly.

Robustness

If a part of the system breaks down, the rest of the system should still be operable. One missing link should not render the system incapacitated. This can for example be facilitated by redundancy and backup systems, as well as adaptive techniques such as ad-hoc networks [14]. Robustness also means resilience, meaning that if incorrect input is accidentally given, the system will still function properly.

Consistency

The system should provide the functionality expected by its users. It should show consistent behaviour, and information in the system should be correct.

Complete reliability is very hard to obtain, in stead systems like the C2000 network usually have a chance of partial or complete failure of a few percent at most. Since the prototype design will rely on the GSM network (as described in Chapter 5), its reliability is dependent on the reliability of this network. It also relies on the supporting storage infrastructure. If for example a file server would crash, it depends on the backup facilities whether the system can still be used. To ensure high reliability, the storage system must be implemented redundantly. The software must be implemented carefully, intercepting errors before they cause the system to fail. At the moment no exact demands are set on reliability, but if the system were to be put to use, the demands would have to be determined and field tests should prove whether the system meets these requirements. Relying on the C2000 system in stead of the GSM network might improve the reliability, however the C2000 system has its own problems [13].

To ensure robustness, the network should be very reliable and so should the PDAs. In case a PDA breaks down, a backup PDA must be available. One PDA breaking down will not influence the rest of the system. Although the prototype design uses the GSM network, the fully implemented system might rely on multiple networks and use advanced techniques such as ad-hoc networking [14] and cognitive radio [15] to ensure a reliable communication network. The storage system must be implemented redundantly to ensure availability. In case of the prototype, the communication systems currently in use are a reliable backup should the system fail. To prevent errors, the system should detect incorrect input and inform the user so mistakes can be corrected.

To ensure correct and consistent information, the director will be responsible for up to date information in the system. If another user detects a mistake, he can correct it himself or notify the director so it can be adjusted quickly. To ensure the system behaves as is expected by the users, the system must be thoroughly tested and users should receive training before they are allowed to use it.

Adaptability

Changing policies and laws and the desire for improvements also require the system to be adaptable. This requires the characteristics shown in Table 9.

Since flexibility is an important issue, reassigning tasks and adjusting priorities must be possible. New information requirements can be added to the system if the data structure supporting it is well designed. For this purpose, an ontology has been designed describing the objects at the incident scene and their relationships. This is discussed in more detail in Section 6.3.3. To support even higher flexibility, the system is designed in a modular fashion. Modules to add new roles or to change access rights to existing ones could be added. Furthermore, the software is built around a basic framework (see Chapter 5) which allows the relatively easy integration of new features. Since the software is developed independent of the supporting hardware it should be possible to migrate to a different infrastructure if required.

Table 9 - Requirements following from adaptability.**Flexibility**

Policies change quite regularly, shifting responsibilities and changing procedures. Adapting the system to changing policies should therefore be relatively easy. Although incident characteristics, and therefore the information in the system, will probably remain the same for the foreseeable future (or be expanded, not changing existing data) it should be relatively easy to change or add tasks and rights, adapting to new tasks and responsibilities.

Maintainability

The system should require little maintenance and if required it should be relatively easy to perform. There should also be a backup system to sustain function while maintenance is performed.

Expandability

When new incident management tools are developed, it should be possible to expand the system with this new functionality.

Maintainability is ensured by the same modular design as mentioned above. Modules can be updated one at a time. Hardware maintenance could be performed by replacing or repairing devices one at a time since each part is implemented redundantly.

The modular approach combined with the basic framework is designed to allow easy expansion of the system. For example, modules could be created to map the ontology to data formats supported by other systems and vice versa.

Other restrictions

Since the system is designed for government organizations a few other restrictions, shown in Table 10 should be kept in mind.

Table 10 - Other restrictions to the system.**Costs**

Since large investments in incident management are in the public domain, they should be justified first. Both investment costs and costs of use, training and maintenance should be balanced by the advantages of the system to society.

Privacy and security

Since the system could involve delicate information, privacy and security are important issues. What are the consequences of the shared access regarding privacy and security and does the law demand certain security measures or prohibit certain information to be shared?

At the moment no estimation of the costs involved has been made. What can be said is that if the system does indeed reduce the time required to clear an accident scene, the

money that can be saved by reducing road congestion and the risk of secondary accidents could well outweigh the costs. Field tests will have to show whether the system does indeed improve incident response times and good estimates of the costs and benefits involved will have to be made.

It should be impossible for unauthorized users to access the system and change data. If an unauthorized person could change critical information it could put human lives at stake or cost society a lot of money. Some of the involved stakeholders are reluctant to share information due to privacy reasons. To make sure no new issues are raised here, the prototype only shares information that is already shared through the existing procedures. Although the design anticipates sharing additional information about responsibilities and authority, this cannot be implemented unless the stakeholders agree this is acceptable. Detailed information about the persons involved will only be accessible by people from their own organisation only as will be discussed in Chapter 5.

Despite its importance to a fully operational system, the prototype features only basic security, since advanced security is beyond the scope of this thesis. Although there are many ways to improve on the security in the completely implemented system, some suggestions are made here. Use of the C2000 network would strongly improve the security since it features data encryption and its communication cannot be overheard by amateur radios. The file servers used should feature good security as well. Many off the shelf solutions exist providing adequate protection. The most important remaining issue is how to protect access to the information in the system. Aside from using user names and passwords, limiting access to known PDAs only (for example by checking SIM card number) might improve security. In that case it would be important to keep track of the PDAs though and in case a PDA would be lost its SIM card should be blocked immediately. Although security is not fully addressed in this project some further suggestions are made in Section 6.4.

4.4 Technologies

The system envisioned is an integral information and communication system. It should facilitate communication and information management at the incident scene. While the software part of the system consists of newly designed modules, the system could also be linked to existing systems such as vehicle registration. Aside from the software solution, the hardware components must be determined. For the supporting data communication infrastructure for example, possible choices are the C2000 system or the GSM network. The physical device on which the system runs could for example be a laptop computer or a personal digital assistant (PDA). It would take too far to describe all techniques that were considered at the first phase of the project in detail here. In stead, this section describes the different parts of the system, their purpose and the aspects underlying their design. In appendix A the possible techniques considered are discussed briefly.

4.4.1 System decomposition

The system can be decomposed into three distinct layers like in [12]. The three parts will be further described below.

Presentation layer

This part constitutes the user interface; its main function is to support the interaction between the users and the system. It should provide the users with clear, unambiguous information and should allow for simple input to the system. Although this part does not facilitate the main goals of the system directly, it is required for the user to interact with the system. A good design will make the difference between a nice gadget which will hardly be used in a true crisis situation and an essential tool in incident management. The ease of use and learnability of the system are largely determined by this layer.

Task layer

The task layer is where the system's actual functionality is provided. It determines what information should be passed on, displayed and stored. It provides (near) real-time and unambiguous communication between emergency services during incident response and supports good evaluation afterward.

Supporting infrastructure layer

The supporting infrastructure layer provides the basic framework on which the system relies. It facilitates data storage and the information flow through the system. It is directly linked to the operating systems and hardware on which the system runs.

Although all layers are required for a well functioning system, this project will focus mainly on the task layer, since here the actual communication takes place. The presentation layer will be designed too, since without it, the system cannot be tested properly in an actual field test. For the supporting infrastructure layer an existing solution is chosen. Although better solutions might be possible for an actual fully implemented system, using existing options, that are easily accessible, allows us to focus on the system design. In the next parts the different components will be examined in more detail.

4.4.2 Presentation layer

The presentation layer is mainly about the interface, what information should be displayed or filled in by the user and in what way should data input or output be facilitated? This can be provided in many different ways (e.g. keyboard and mouse for input and a screen as output). As noted above (Section 4.3.1) there are some limitations to the possibilities, depending on the situation in which the system is used. There are many options using different modalities, each of which has its own advantages and disadvantages.

Auditory

When used for one-to-one communication through cell phones or communication on site speech is the most used form of communication. An advantage is that while communicating, both hands can be free and a user can look around and quickly shift

attention to other, more urgent matters (e.g. secondary accidents) if required. When used for direct data input into the system auditory input becomes impractical because of the often noisy environment on the motorway and limitations to speech recognition software and its required processing power. Auditory output has the same limitations and does not allow for quick focus on a specific part of the system as visual output does.

Visual

Visual input systems like camera's are often bulky and often need human interpretation and processing before practical information is available. Video especially requires high communication and processing speeds. Single pictures might clarify the situation greatly though without demanding too much bandwidth. For output, a simple screen can be used like on a PDA, in this way, information can easily be presented. For the system envisioned, a screen seems to be the most practical (and most familiar) form of output to the user.

Haptic (touch)

Although not a truly haptic means of communication (e.g. like Braille), most used input systems are a keyboard and mouse, or a variety of the mouse, a stylus (in combination with a touch screen). Disadvantages of these systems are that to remain mobile, they are often very small, which might raise issues when used outside in cold weather. On the other hand, these forms of input are highly familiar nowadays with the emerging use of personal computers and personal digital assistants.

Other

Other modalities like taste and smell are not considered to be practical for communication systems.

Considering the points discussed above, the basic modes of interaction for the presentation layer can be distilled. The system should support auditory input and output to allow for direct phone calls to other emergency respondents, but for the rest of the system, auditory input will not be supported for reasons discussed above. The most practical form of output is through a screen, which is a standard feature on PDAs and laptops. For input the visual modality is not very practical. Most PDAs support a stylus and small keyboard, which are also the most practical forms of input. Special attention should be paid to the usability of these options on the incident scene though. Altogether, the hardware interface can be fully facilitated by a laptop, tablet PC or PDA and therefore these devices are all suitable for the system. Considering the fact that the device will probably have to be put away regularly at the incident scene, a PDA is the favoured option there, since it will fit in a pocket.

Aside from the modalities used for user interaction, user interfaces come in many forms and shapes. The interface presents the information requested by the user and objects like buttons or text fields to facilitate this interaction. The interface should be simple, unambiguous and clear and easy to use and learn. A good interface design displays the necessary information without overwhelming the user with irrelevant options. By grouping similar functionality and focusing specific subjects, clarity to the user can be

obtained. It is also essential that before it is put to actual use, the system is thoroughly tested in accurate and realistic test runs involving actual intended users.

Fully discussing the subject of Human Computer Interaction here would simply take too far, it suffices to say that using techniques and conventions from the Human Computer Interaction paradigm a suitable interface has been designed [12]. The system has been tested in a lab setting as discussed in Chapter 7, test results. Section 5.3 discusses the interface in more detail.

4.4.3 Task layer

The task layer provides the actual functionality of the system. There are many possible architectures that could provide the functionality required for IMICS. In addition, incident management can be supported by a variety of services and functions. This section describes the architectures on which the design is based and the functions supported by it.

4.4.3.1 System architecture

For problems such as data storage, sharing and synchronisation many solutions exist. Since every situation favours a different approach, it must be decided which solution is the most suitable to incident response on the Dutch motorways. Since designing a complete system from scratch would go beyond the scope of the project, an existing solution is chosen as a foundation on which the rest of the system is built.

Of course, the current situation already implements the basic needs of the emergency services at the involved control rooms. One of the possibilities for the IMICS system is to simply build on top of this situation by collecting all information in the control room and having one person, an information manager, check and distribute it to the right persons as in [16]. This approach is quite similar to the current situation. In all probability, this functionality will always be supported and can serve as a backup system. The goal of this thesis however, is to design an automated system that could improve on the current situation.

The proposed system is an IT solution that borrows from a few existing software architectures. The design is based on an existing framework called Java Agent Development Environment, shortly JADE [17]. JADE is a hybrid peer-to-peer system that provides an infrastructure that supports the creation of automated agents and a messaging structure through which these agents communicate. Using JADE, a blackboard like system is designed. Chapter 5 discusses JADE and the designed system in detail. In Appendix A an overview and short description of the alternative architectures considered during the first phase of the project is given.

4.4.3.2 Services and functions

As described in Chapter 3, there are some basic requirements to incident management systems. Although these functions should all be supported, to limit the scope of the project, not all of them have been implemented in this proof of concept. Table 11 shows the services and functions that were considered in the first phase of the project (see Appendix A for more details). The table also shows the functions that are supported by the system design, and which are actually implemented. The functions that are included in the system design are described in detail in Chapter 5. An overview of the different services and functions considered during the first phase of the project is given in Appendix A.

Table 11 - The different functions considered at the start of the project.

Function	Included in design	Implemented in prototype
Integrated communication	yes	yes
Phonebook and address book	yes	no
Human operator	yes	available
Decision support system	no	no
Relevance inference	no	no
GIS (Geographic Information Systems)	yes	no
Tracking & tracing	yes	no
Logging	yes	yes
Statistics	no	no
Automatic data gathering	yes	no
Synchronization	yes	only basic
Flexibility	yes	yes

4.4.4 Supporting infrastructure layer

The functions described above require a basic supporting infrastructure. Data storage, communication and display all require specialized systems. This section elaborates on the different components and the proposed solutions.

As noted before, in the current project, a prototype is developed using existing solutions for the underlying infrastructure. The chosen techniques are not necessarily the most suitable ones for a fully developed system, but they allow for a relatively simple implementation and testing of the prototype, while providing the necessary functions. Which techniques are used and which techniques would be best for a system put in actual use is described below.

4.4.4.1 Access points

At the incident scene, emergency respondents should be able to access the system. Depending on the situation and user preferences the system itself could take different forms. Access to the system could be provided by laptops in the emergency vehicle or

emergency respondents could be equipped with PDAs or a tablet PC. Although a laptop has more computing power and storage capacity, the advantage of mobility a PDA offers has its merits. A combination of both is also possible, for example equipping personnel with a PDA and placing a laptop in their vehicles. A tablet PC is somewhere in between, offering a larger screen and more computational power than a PDA, but being less powerful and more portable than a laptop computer. An important consideration is whether all emergency respondents should be equipped with a device or just some, like the director or the person in charge of the COPI team.

Since it would be cumbersome to carry a laptop or tablet pc while working at an incident scene, a PDA appears to be the most logical choice. Most PDAs fit in a pocket while still offering decent performance. To cope with usability issues such as cold fingers making the small keyboard useless, personnel could be equipped with a laptop or tablet pc in their car. Since the information available in the system should be accessible to every incident respondent, every respondent should at least be equipped with a PDA. Although the system design is based on this choice, the proof of concept implemented for this project has been developed and tested on a desktop computer with a fixed line network connection.

4.4.4.2 Network communication

To support communication and data transfer a network connecting all users is required. There are many considerations involved in the choice of a network and the way information is communicated. The most important are listed below.

Communication standard

It is important to consider what information should be communicated and the way information is processed and sent to other users. Should the information follow predefined structures or procedures? Some existing standards are XML [18], the XML based Common Alerting Protocol (CAP) [19], which is especially designed for emergency management, and the ACL message structure defined by FIPA[20]. A nice overview of examples can be found in [21]. The choice of a standard depends on the organizations involved and the communication standards supported by other systems in use as well as legislation.

Using a standard like CAP could prove useful and could improve compatibility with other systems in use or under development. Since the situation is quite specific, a choice has been made for a different approach however. The final system design is based on JADE since it provides much of the required functionality. JADE agents communicate by sending ACL messages as defined by FIPA [20]. Within the structure of an ACL message, user defined information can be sent. To facilitate consistent communication, an ontology has been designed specifically for incident management on the Dutch motorways. The information sent in the ACL messages is structured according to this ontology. The JADE framework and the designed ontology are discussed in more detail in Chapters 5 and 6.

Availability and robustness

The system should preferably be available at all times, which means the network should support complete coverage and no down time.

Wired/Wireless

A network can be connected physically by cables, or it can be connected wirelessly. Since incident locations cannot be predicted in advance, wireless communication is the obvious choice for the IMICS system.

Data transfer rate

To ensure robust performance, the network should support a fast connection so users do not have to wait very long to enter or look up information. When many users are involved (e.g. in case of a large, complex incident) system performance should not degrade.

Security

Since incident management could involve delicate information, all communication should be encrypted. It should also be impossible for unauthorized people to access the system.

Supporting network

There are many types of network available with different characteristics. For IMICS important issues are reliability and security, however, for the proof of concept for this thesis, a network that is widely available and easily accessible is more suitable. For that reason, the design is based on both the GSM network and fixed lines, while the actually implemented system is tested on a fixed line network only. More details on different networks are given in Appendix A.

4.4.4.3 Adaptive network techniques

Since the GSM network the system is based on cannot be expected to be 100% reliable, the system must be designed with interrupted connections in mind. Ad-hoc networking techniques could be used to enhance reliability, or a development called cognitive radio [15] could be used to provide extra bandwidth if required. These techniques are not discussed in detail for this thesis, but Appendix A provides some background.

4.4.4.4 Data storage

To provide access to information, this information must be stored somewhere. Before deciding on storage mechanisms, it should be clear what information has to be stored. Is a simple form containing textual descriptions of certain properties of an incident enough, or is more information needed, such as meta-data and relevance data. It should also be clear what type of information should be stored. Can all data be stored as plain or structured text, or is a more advanced type of data required, such as image or video files? As has been discussed above, an ontology has been created to determine the structure and types of the data stored in the system. Its design is based on the incident situation, the procedures currently in place, and the data required by the involved stakeholders. Chapter 6.3.3 elaborates on the details.

Aside from what data is stored, where the information is stored is also an important issue. Since local storage may require too much processing power for a PDA, this is probably not a practical solution. Distributed storage could have the same problem, and also requires a lot of overhead for synchronisation. The most suitable option seems to be centralized storage, using dedicated servers. This setup requires backup facilities and strict security policies though. The use of backup servers requires synchronization schemes to keep all data consistent. To get the best of both worlds, a hybrid solution is chosen. The design details can be found in Chapters 5 and 6. Once more, a more comprehensive overview of techniques considered for IMICS is given in Appendix A.

4.5 Summary

To recapitulate, our proposal for IMICS constitutes an information system based on desktop computers and PDAs using a stylus and keyboard for interaction. The first prototype is designed for the police, fire brigade, ambulance service and road inspector and offers a supporting role for the emergency and traffic control rooms. For data communication, the system connects to the Internet through a fixed line or the GSM network. If the situation requires it, the PDA can be used to call any user, or stakeholders not using the system, over the same GSM network.

The system is designed to share data between its stakeholders at any location. It is designed as a JADE based system that provides a blackboard like functionality. For storage the prototype relies on a hybrid system with both a centralized server and local storage. Evaluation is supported by logging all entries to the system and storing them with additional information like timestamps and the responsible user. Since many other additional functions may be added later, such as automated incident notification, decision support and GIS, the system is designed in a modular fashion. This also improves maintainability.

To regulate access, the system could support different roles each with its own rights. For the first prototype however, the system provides the same access rights to all involved stakeholders. When important updates are made, all respondents will receive a notification automatically. To improve flexibility, it is possible to reassign roles. The director role must always be fulfilled however. Basic security will be provided to protect the sometimes delicate information in the system. Before accessing the system, a user will have to identify himself using a user name and password.

This chapter has laid out the basic analysis that has served as the basis for IMICS. Now that most aspects of the intended system and the possible solutions have been described, the system design can be explained. The next chapters discuss the design details.

Chapter 5

Global design

This chapter describes the design of the Incident Management Information and Communication System (IMICS) as it currently stands. As stated in Section 1.2, the goal of this project is to improve communication and situational awareness and evaluation of incident management on the Dutch motorway network. It aims to improve the distribution and management of information among participating users that are distributed over different locations. The system must also keep track of all changes in a log to support evaluation of the incident management procedures. All this is achieved by a distributed blackboard system that can automatically distribute information using a multi agent platform.

Although the system will run on both desktop computers with a fixed line network connection and mobile devices, its design is based on a PDA connected to the GSM network. Designed with the limitations imposed by a PDA in mind, the system can be run on a (more powerful) desktop computer without trouble. Using stylus and keyboard based input the system can provide functionality similar to a very basic personal computer. The use of these devices is fairly familiar nowadays thanks to modern cell phones and personal computers. Although the system will be designed to be as flexible as possible, the PDA can be used as a cell phone for direct communication with other stakeholders if required. Considering the computation power of current PDAs, the software part of the system should be designed with limited memory capacity and processor speed in mind. For the underlying communication network the GSM network seems a fair choice at the moment since it is widely available and allows for fairly high data transfer speeds. In a final design the C2000 network might be used because of its secure and reliable nature. Considering the limited availability of information due to its classified status, and the restricted access to the C2000 system however, the GSM network is a better choice for this proof of concept. For data storage the design relies on both a centralized server and local storage.

The remainder of this chapter describes the system architecture and the interaction with the user. Important design considerations are highlighted in this section and the pros and cons of each option are discussed. Section 5.1 describes the supporting infrastructure layer on which the system is based. It describes the system topology and the software

architecture providing the basic infrastructure for connecting to, and communicating with, other peers. In Section 5.2 the task layer and the client and server parts it consists of are discussed. Also an overview of the tasks these parts are responsible for is given. Section 5.3 describes the presentation layer. It presents the user interface of the implemented proof of concept and design sketches of the rest of the system. The intended users and their roles and responsibilities are discussed in Section 5.4.

5.1 Supporting infrastructure layer

As mentioned in Section 4.4.1, the supporting infrastructure layer provides the basic framework on which the rest of the system relies. It provides the link to the hardware on which the system runs and facilitates basic features such as data storage and communication. The functions required in this layer are fairly general and therefore many systems exist that can provide this functionality. To save time, an existing open source system has been chosen as a basis for the system infrastructure.

5.1.1 Hardware

The proposed system consists of a client-server network architecture, in which the client systems can be both desktop computers and PDAs. While the emergency control room has desktop computers with a fixed line network connection available, the officers in the field are able to access the system through the use of PDAs connected by the GSM network. The server facilitates data storage and distribution while the client system facilitates the interaction with the users. For our prototype, data transmission will be facilitated by the JADE framework over fixed lines. In a completely implemented system, the network connection could be based on the C2000 network or may require an ad-hoc infrastructure to be implemented on top of the GSM network to improve reliability.



Figure 6 - A police officer using a PDA.

5.1.2 Software architecture

The system is modelled as a distributed blackboard system. A blackboard system is an artificial intelligence application where a common knowledge base, the ‘blackboard’, is iteratively updated by a diverse group of specialist knowledge sources. Users submit information to the blackboard and the information on the blackboard can be accessed by other users. Automated agents are responsible for distributing and managing the information. In a distributed blackboard, the blackboard is divided in multiple parts, each part containing a part of the knowledge base. In the IMICS system, each client device has its own local blackboard. The server system is responsible for updating the information on each client, taking care that all clients involved in an incident have access to the same information.

Each client node has a local blackboard facilitating storage and functionality. The server has a central blackboard with which all client blackboards exchange data. The central blackboard is responsible for receiving updates and distributing the new data to the client blackboards. The central blackboard is also responsible for data exchange with external systems (e.g. license registration database). This is shown in figure 7. If a client node loses its connection to the system, it cannot receive updates. To keep users as up-to-date as possible, the server will send the missed updates to the client as soon as the connection has been re-established.

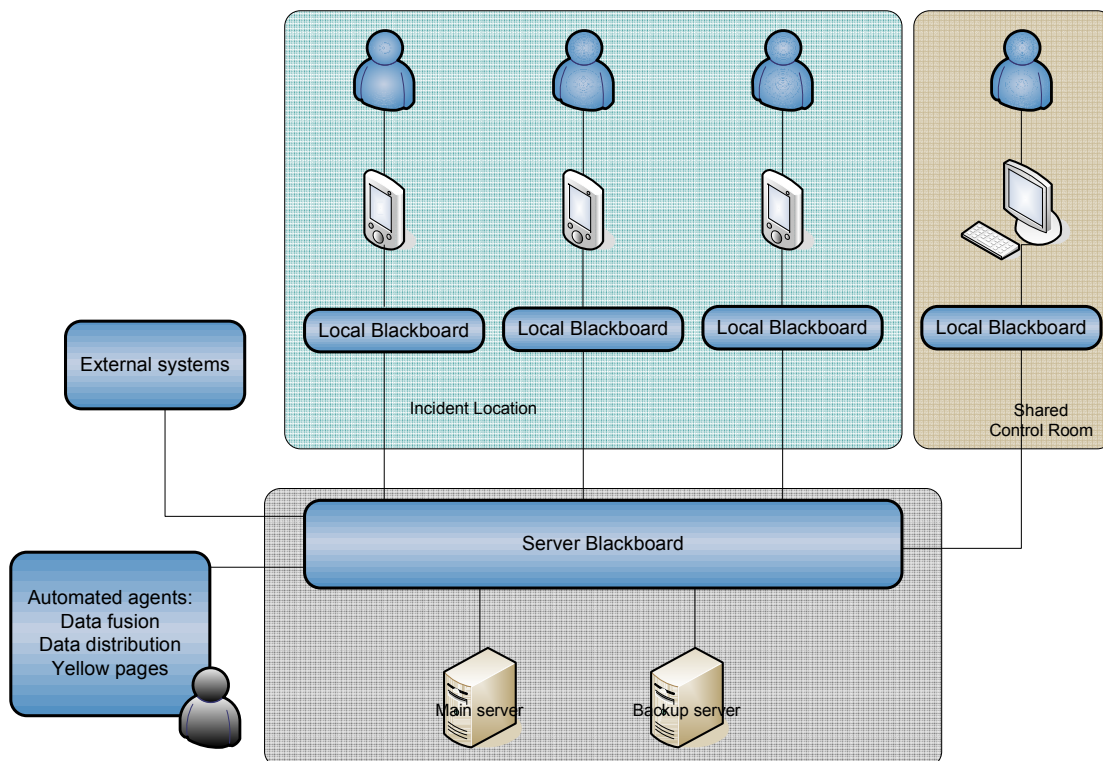


Figure 7 - System overview.

5.1.3 Open source middleware

As mentioned above, an existing open source project was chosen as a basis for the design of IMICS. A few middleware systems with which other people at my faculty have had some experience were considered. These systems provide the basic tools for connecting mobile devices, allowing them to send messages and share data among each other. Since the considered systems are all Java based, they are platform independent, meaning they can run on most operating systems, also those specialised for PDAs.

Cougaar

Cougaar, an acronym for “Cognitive Agent Architecture”, is a Java-based architecture for the construction of large-scale distributed agent-based applications. It offers a blackboard to which agents can publish and subscribe, allowing them to share data. Based on its documentation and the experience of other people at my faculty, it was found that Cougaar was quite complex and it appeared to be not as well documented and supported as JADE. More details on Cougaar can be found at [22].

Lime

Lime is a Java-based middleware application specifically designed to support ad-hoc networking among mobile hosts. It too lacks the large user base and the ample documentation that is available for JADE. At [23] more information on Lime can be found.

JADE

JADE (Java Agent DEvelopment framework) is a software framework fully implemented in the Java language [17]. It simplifies the implementation of multi-agent systems through a middleware that complies with the FIPA specifications [24]. JADE enables connecting mobile devices and sharing data among them. It also provides basic security and synchronization tools, making it well suited for the design of IMICS. It also has the most extensive documentation available on its website and has a very large active user base. For these reasons, it was decided to base the system on the JADE framework. The next section provides more details about JADE.

5.1.4 JADE

JADE provides a platform on which automated agents can run, discover other agents and send messages to each other. Aside from managing agent startup or suspension, agent communication (including basic security) and discovery, the platform itself does not provide actual functionality. The agents run on the platform are responsible for the functionality of the IMICS system.

Agents

JADE conceptualizes an agent as an independent and autonomous process that has an identity, possibly persistent, and that requires communication (e.g. collaboration or competition) with other agents in order to fulfil its tasks. This communication is implemented through asynchronous message passing and by using an Agent Communication Language with well-defined and commonly agreed semantics [25].

When a JADE agent is started by the JADE runtime, it first runs a setup method in which the agent is initialized. After its initialization, an agent usually runs one or more agent behaviours, which are discussed below. Even if no more behaviours are active, an agent does not cease to exist until its *doDelete()* method has been activated. When this method is called, the agent's *takeDown()* method is activated to do any last operations before the agent is finally terminated. (See the JADE tutorial [26])

Agent containers

The agent platform can be distributed across machines as Figure 8 illustrates. On each machine an instance of the JADE runtime is activated. These JADE instances function as agent containers. Agent containers can connect to other containers, allowing the agents active on these devices to communicate and share data. However, JADE is a hybrid peer to peer system, meaning it has peer to peer functionality, but it also has a centralized index managing access to the network. The central index is a special node that provides a service that simplifies the look-up and discovery of the active peers, their list of capabilities, and their list of provided services. A single special main container must always be active to provide the central index. All other containers register with the main container when they start in order to form a distributed JADE platform. The other containers must know the host name and port of the main container to be able to connect to it. To reduce the risk of a single point of failure caused by the centralized index, the main container can be replicated over multiple nodes to provide a backup in case of a system crash.

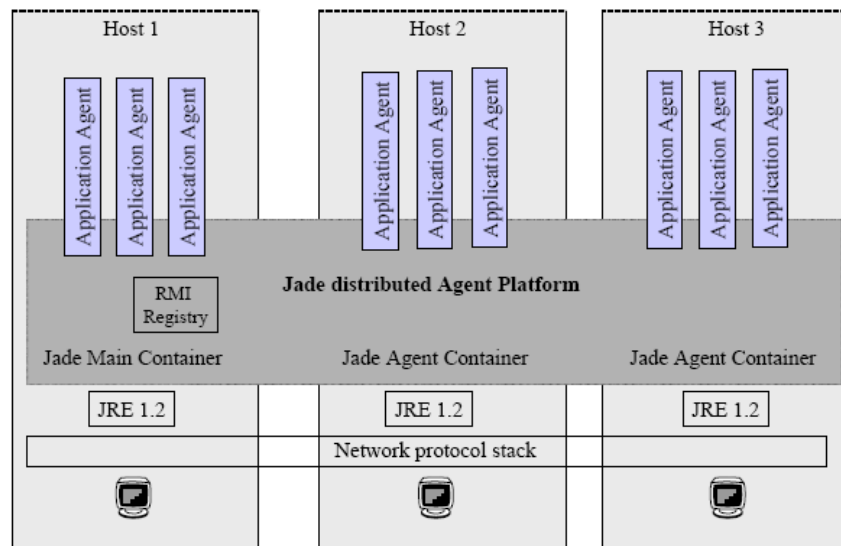


Figure 8 - JADE Agent Platform distributed over several containers.

The main container has two special agents activated on startup, namely the Agent Management System and the Directory Facilitator.

Agent Management System (AMS)

The AMS ensures that each agent in the platform has a unique name. It also represents the authority in the platform, it can for example create, activate or kill agents on any container in the platform.

Directory Facilitator (DF)

The DF provides a Yellow Pages service by means of which an agent can find other agents. An agent can publish the services it provides with the DF so that other agents can find and successfully exploit them.

Agent behaviours

A JADE agent carries out its tasks by means of behaviours. These behaviours can be added to the agent's pool of active behaviours when the agent starts up, or can be added later (e.g. by another behaviour after a specific message has been received). There are three types of behaviour: one-shot behaviours execute an operation once, cyclic behaviours perform an operation repeatedly and generic behaviours keep track of a status variable and perform different operations depending on the status.

Behaviours can be temporarily blocked, for example to wait for a specific message to arrive. Whenever a message is added to an agent's message queue, the agent resumes all blocked behaviours. Depending on the desired functionality, a behaviour can block again if the received message is not the right one. Once a behaviour is finished, it is removed from the pool of active behaviours. Figure 9, taken from the JADE tutorial, shows the way behaviours are handled by JADE agents. More details on agent behaviours can be found in the JADE Tutorials [26].

Agent message queue

The JADE runtime automatically posts messages in the receiver's private message queue when they arrive. Agents can pick up messages from their queue with their *receive()* method. If there are messages in the queue, this returns the oldest message in the queue and removes it from the queue. A template can be used to select only messages of a specific kind.

ACL messages

Messages exchanged by JADE agents are formatted according to the ACL language defined by the FIPA international standard for agent interoperability [24]. This format comprises a number of fields such as the message sender, its receivers, its performative and the message content. The performative indicates the intention of the message, examples are REQUEST and ACCEPT_PROPOSAL. JADE provides a set of skeletons of typical interaction patterns to perform specific tasks, such as negotiations, auctions and task delegation [27]. JADE also provides a special wrapper class that allows Java objects to be sent in an ACL message. More details can be found in the JADE tutorial [26] and the FIPA website mentioned above.

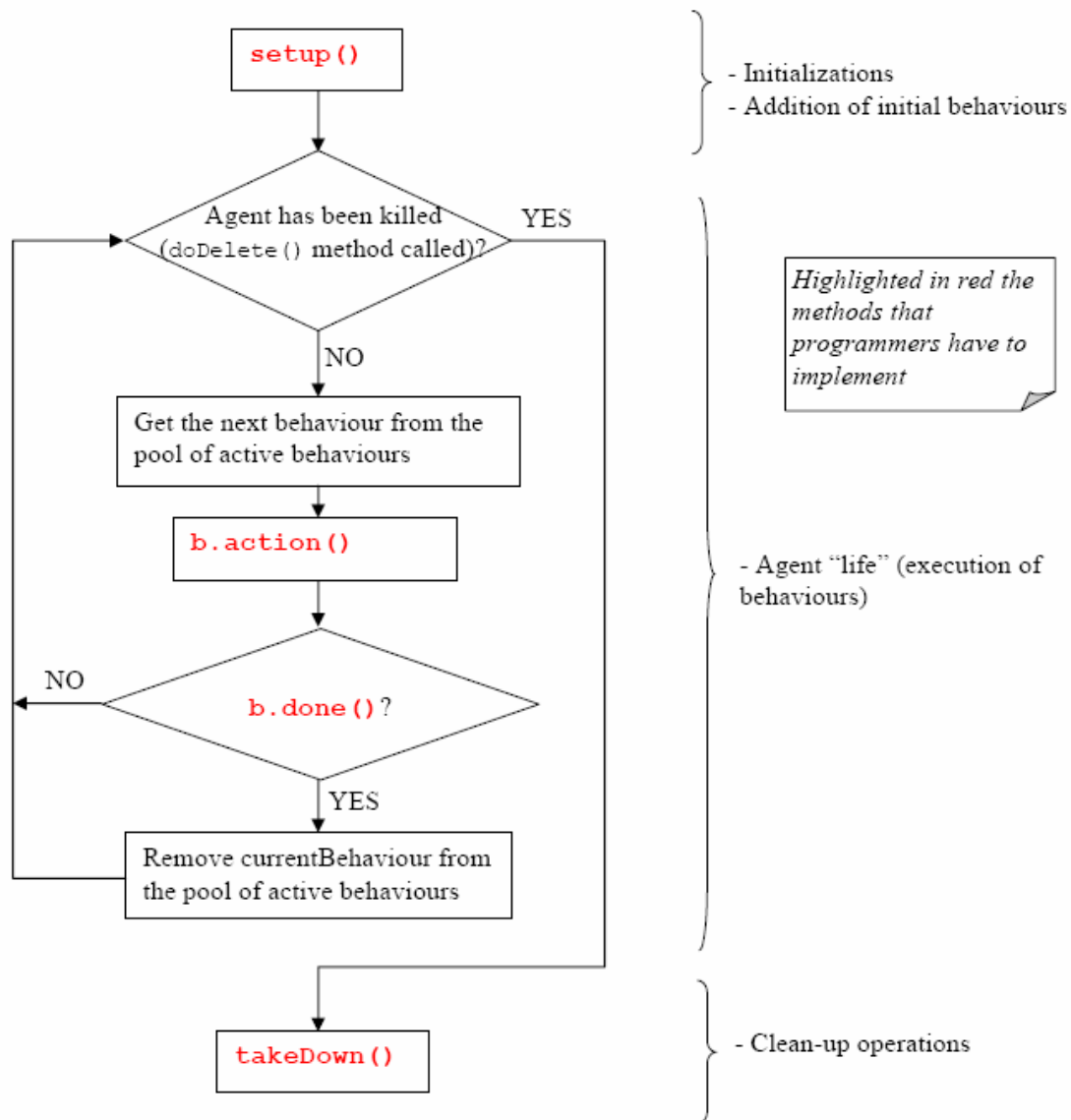


Figure 9 - Agent Thread path of execution as in the JADE programming tutorial.

Using JADE as a basis, agents have been created that provide a client server architecture with a blackboard like function. The blackboard architecture forms the basis of the system. It provides the basic functions required by the IMICS application, such as sending and receiving updates as described in the next sections. More details on JADE can be found in the tutorials and documentation available at [17].

5.2 Task layer

The task layer provides the actual functionality the system is designed for. For IMICS, it provides the actual blackboard like functions. It processes the information available and determines what information is distributed and to whom and whether data is stored locally or at the server. Globally, the system can be divided into two separate systems, a

client system, providing the functionality to the users, and a server system that serves as the agent management system (AMS) and directory facilitator (DF) for the JADE framework and provides the interaction with external systems. Ordinary users never interact with the server directly. In fact, for the proof of concept, the server is completely automated and does not provide any means for users to interact with it directly. Naturally, for practical use of the system, such an interface must be provided to support tasks such as maintenance by system administrators.

As Section 4.4.3.2 discussed, the system design provides a number of features to improve incident management. Table 12 gives an overview of the functions from Chapter 3 that the user could perform by using the system. It also shows whether the functions have been implemented, or are just envisioned for a future implementation. Section 5.3 discusses the user interface of the designed system and briefly discusses the design of the parts that have not been implemented.

Table 12 - The tasks a user could perform using IMICS.

Function	In design	Implemented
Request resources	Partially	No
Allocate, delay or deny resources	No	No
Report and update situation	Yes	Yes
Analyse situation	Yes	Yes
Edit, organize, and summarize information	Yes	Yes
Maintain resources (logistics)	Partially	Partially
Acquire more or new resources	No	No
Oversight review, consult, advise	Yes	Yes
Alert all with a need to know	Yes	Yes
Assign roles and responsibilities when needed	Yes	No
Coordinate among different resource areas	Yes	Partially
Priority and strategy setting	Yes	No

The next sections discuss the basic functionality of both the server and the client part of the IMICS system.

5.2.1 Server system

The server system would normally run on a fast server computer, connected to the internet with a fixed line. In the current implementation, the server system simply runs the JADE main container, facilitating the AMS and DF and allowing other agents (client systems) to connect to this container. The only agent running on the server is the serverAgent. The serverAgent is responsible for receiving updates, updating its central database, and forwarding these updates to the right users. It could also connect to data from external sources.

A central server makes the system vulnerable since if the server fails, the entire system fails. For that reason, JADE can duplicate the server to another system, reducing the risk of failure [28]. Although this is an option that simply has to be activated on JADE

startup, it has not been implemented and tested for this proof of concept. The design of IMICS, however, assumes a duplicate server is always active.

As the server has no user interface, it can only be accessed indirectly by an IMICS client. If future additions to the system require a direct interface with the server system however, this can be implemented on top of the server agent without requiring drastic changes to the existing part. The interaction with this new user interface can be implemented by adding new behaviours to the server agent or by creating a new agent that runs on the JADE framework independent of the server agent. The favoured option depends on the kind of functionality required. For example if an interface is needed to access old incident data for evaluation purposes, a completely new agent would be the best choice, since this function is independent of the operational incident management facilitated by the serverAgent. If for some reason an overview of the sent and received messages for a specific incident is needed, it would probably be best to simply add a behaviour to the existing server agent since it is responsible for handling these messages.

5.2.2 Client system

The client system provides the actual interaction with the user. It can be run on a desktop machine in the emergency control room, or it can be run on a mobile device at the incident location, for example a PDA or a laptop in a police car. Depending on the device it runs on, it can connect to the server over a fixed line or through a wireless internet connection. Before a user can access the client system, he has to provide authentication credentials. After the user has successfully logged on to the system, the client system starts a JADE container on which the client agent is run. The client agent asks for the server name or IP and registers itself with the directory facilitator of the server and sends a notification requesting the current incident data. After receiving this data from the server and storing it locally, a list of the currently open incidents is shown. In a complete implementation of the IMICS system, this list should only contain the incidents in the region the user is assigned to, but in the current implementation it simply shows all incidents open on the server. The user can now select an existing incident from the list or create a new incident.

Creating a new incident

Creating a new incident shows a number of empty forms in a tabbed layout in which the user should enter as much information as available to him. After this has been done, the user should save the data, after which the client agent takes care of sending the update to the server, which in turn forwards the update to all users in the region. The new incident is now shown in the incident list for all users assigned to the region and can be opened by the users assigned to this specific incident.

Opening an existing incident

Users can open the incidents they are assigned to. Opening an existing incident shows the same forms as creating a new one, but the information already in the system is already filled in. The user can now either look up or modify existing data or enter additional data

into the forms, after which saving will send the information to the server as with a new incident. If an update is received in the meantime, the local data is updated immediately and directly shown in the interface. As will be discussed in detail in Chapter 6, synchronization is an important issue here. As the current implementation is a simple proof of concept, it does not have an advanced synchronization mechanism, but simply overwrites all data with the latest update. The current system is therefore prone to losing data when an update is received when entering data or when an old update is sent after a lost connection has been re-established.

User assignment

In the implemented proof of concept, users are all automatically assigned to all incidents. In a full implementation however, users can only be assigned to an incident by other users that are already assigned. Although this has not been implemented, the control room is automatically assigned to all incidents in its region as it is responsible for the coordination of the response and is usually responsible for providing the most information in the system. Although all involved users can do it, the control room is usually also responsible for assigning persons to an incident. An exception to this is when a field officer is the first to discover an incident and creates a new incident. He is then automatically assigned to the incident. The control room should also be able to temporarily assign users from other regions if required (e.g. in case of a severe incident or when an incident is located near the border between two regions).

In the next section, details of the interface and the kind of interaction the user has with the system are described.

5.3 Presentation layer

The presentation layer constitutes the user interface of the system. It provides clear and unambiguous forms with which the user can interact. It presents the available information in a well structured and understandable way and allows the user to fill in and save new information.

The server and client parts as presented so far have been implemented for the proof of concept (as is discussed in more detail in Chapter 6). In a complete implementation of IMICS however, keeping track of the incident data is only part of the system, as the design of IMICS comprises more functions. In this section, the interface of the complete design of the envisioned system is discussed. Functionality that has not been implemented is briefly explained, accompanied by sketches of a possible interface. The interface of the implemented part of the system is discussed using actual screenshots.

As discussed above, the server part of the system does not have a user interface and cannot be accessed by the user directly. Therefore this section treats the client system only.

General interface client system

The client system will run as a stand alone application that communicates with the server system. The client system can run on both a desktop computer and a PDA. It provides a clear overview of the information available in a well structured manner. Figure 10 shows a possible global lay out of the system.



Figure 10 - A possible division of the screen.

This lay out has been worked out in more detail to give an idea of the possibilities of the system. Figure 11 shows the design for the main window of the client application. Clicking on the buttons on the left opens a new window that shows detailed information about the department indicated by the text and icons. The tasks button opens an overview of the personal tasks the user has and the progress towards completion of these tasks.

Geographic information system (GIS)

The map in the centre of the screen (actually a screenshot taken from Google Maps) shows the location of a hypothetical incident. This map could be part of a geographic information system (or shortly GIS, see Appendix A). Depending on the overlays activated by the icons on the right, events, persons, vehicles and other items of interest can be displayed in the screen. This part is not implemented for the prototype since it is not the focus of this project.



Figure 11 – Interface design for the IMICS system.

Communication system

The lower part of Figure 11 shows a few tabs that contain different information. They can for example contain a simple mail client or contact information of the persons involved in the incident currently shown. This part is not implemented. Considering the limited screen area of PDAs, the different tabs could be opened in a new window or hidden for the user until selected. Once more, this design is only meant to give an idea of the possibilities of the system.

Task overview

A general task overview is available to all users involved in an incident. Most incidents involve a sequence of standard tasks and procedures. Some tasks can only be started after

certain other tasks have been completed and some tasks can be fulfilled by people from different organisations. The task overview allows users to check on progress on a certain task, or to reassign a task to another user (possibly from another organisation). It gives a complete overview of the tasks of all organizations directly involved. This is also not implemented for the proof of concept. Figure 12 shows a possible lay-out for this screen. Note once more that this is only a sketch to indicate the possible functions provided by the system.

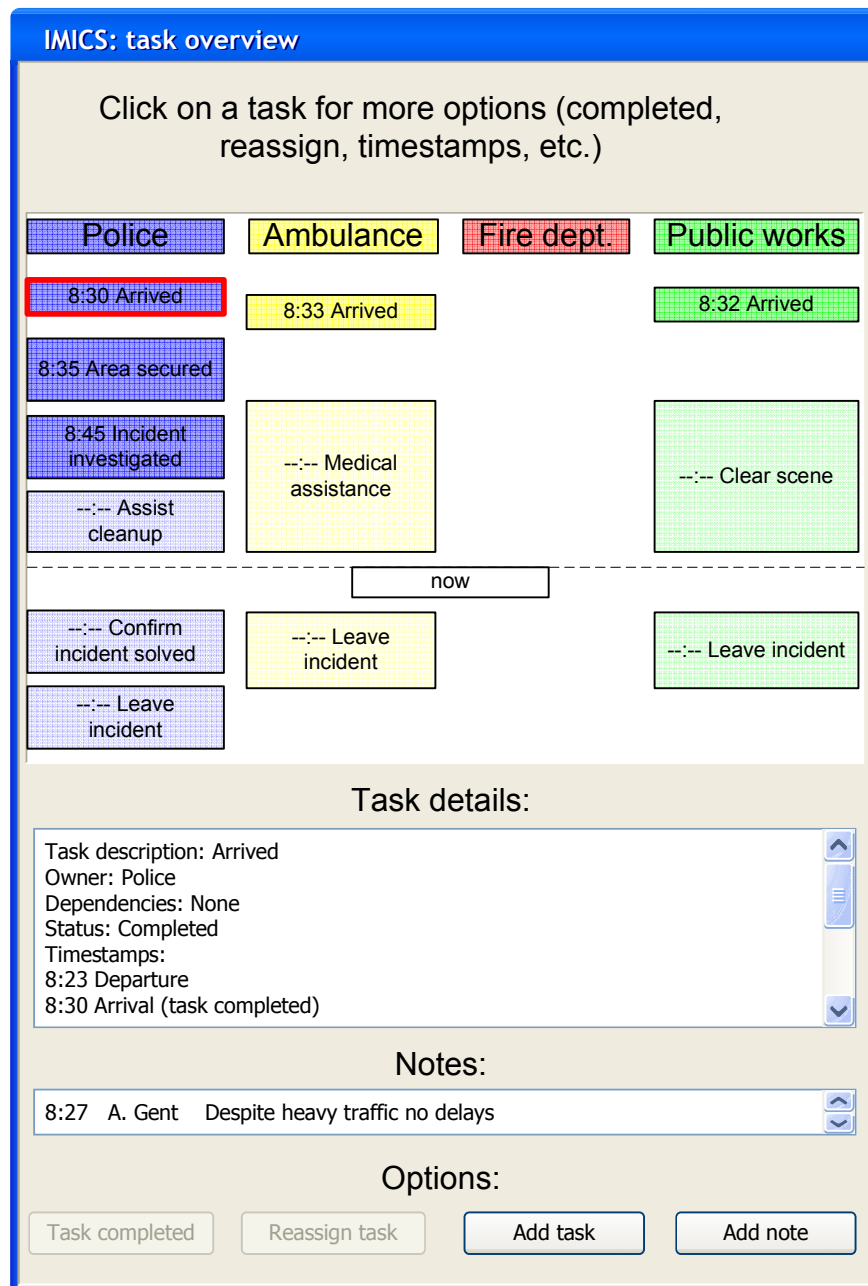


Figure 12 - Interface design of the task overview.

All tasks are grouped in columns representing the different organizations. The vertical axis depicts time, starting from the first people being dispatched at the top. The completed tasks are given a dark colour, while incomplete tasks are given a light colour. (Active tasks could be given a different colour to highlight them) The dashed horizontal line shows the current time. Although the basic tasks involved in all incidents are already there from the creation of a new incident in the system, new tasks that are required by the special nature of an incident can be added. Tasks can be selected by clicking them (as indicated by the red line around it). The details of the selected task are shown in the text box in the middle of the screen and below it user notes can be read. On the bottom of the screen the user can indicate that the selected task is completed, or the user can reassign a task to a different organization if necessary. (Note that in Figure 12, since the selected task is already completed, both these options are no longer available). It is also possible to add a new task or a note here.

Organization overview

The system also provides a specialised screen for each primary stakeholder. Here the involved organisations can access information that should not be available to people outside of their organisation. For example, information could be retrieved from their own private servers through this part of the system. The buttons on the left of Figure 11 open new windows with detailed information about the organization selected. This is also not implemented. An impression of the kind of information that could be found here is given in Figure 13.

The personal tasks of each officer and the general tasks of every organization can be found here. Other relevant information such as information about supervisors or contact information can also be found here. The branch of authority gives a compact overview of the persons involved in the incident and their roles. Once more, it must be stressed that not every organization would like this kind of information to be available to everyone. Especially the police do not want personal information to be accessible by other organizations. Depending on the stakeholders' desires, information on the involved organizations could be shared with other organizations or be accessible by own personnel only.

The interface is titled "IMICS: police overview" and contains the following sections:

- Involved police officers:** A list box containing "R. Deniro" (highlighted), "A. Gent", and "H. Westbroek".
- Role:** A list box containing "Senior officer", "Assistant", and "Assistant".
- Selected:** A text label displaying "Selected: R. Deniro, senior officer".
- Phone number:** A text label displaying "Phone number: 06-11111111".
- Completed tasks:** A list box containing "10:34 Arrival", "10:38 Secure area", and "10:45 Incident scene investigation".
- Remaining tasks:** A list box containing "Approve cleanup" and "Clear road".
- Superiors:** A list box containing "Z.E. Bigboss".
- Subordinates:** A list box containing "A. Gent" and "H. Westbroek".
- Branch of authority:** A tree view showing a hierarchy where "Z.E. Bigboss" is the parent and "R. Deniro", "A. Gent", and "H. Westbroek" are children.

Figure 13 – Interface design of the police interface.

Detailed incident information

One of the most important aspects of IMICS is improving situational awareness. For this reason, the system maintains a well structured overview of the details of the incident. Exact information on, for example, the amount and type of vehicles or injuries involved can be found and modified here. Together with the client and server architecture, this constitutes the actually implemented part of the system.

As mentioned in Section 4.2.1, the stakeholders normally gather information through the same procedure. The information they need largely overlaps [9]. To provide this information, the system has a part dedicated to sharing this information with all users in a well structured manner. This part of the system can be accessed by all users involved in a specific incident. All users can read the information available and add or update it if required, as will be discussed in the next section. The ‘shared memory’ created in this way will improve the reliability of the information since all users can correct mistakes [10]. However, to keep the information in the system consistent when all users are occupied by their relief work, there should be a clear responsibility for maintaining an overview and keeping information up to date. This will be discussed in more detail in Section 5.4.

Figures 14 to 20 show actual screenshots of this overview as it is implemented in the prototype. Before any information can be accessed, the user has to select the incident in which he is involved, or create a new incident in the system. Figure 14 shows the interface where incidents can be selected.

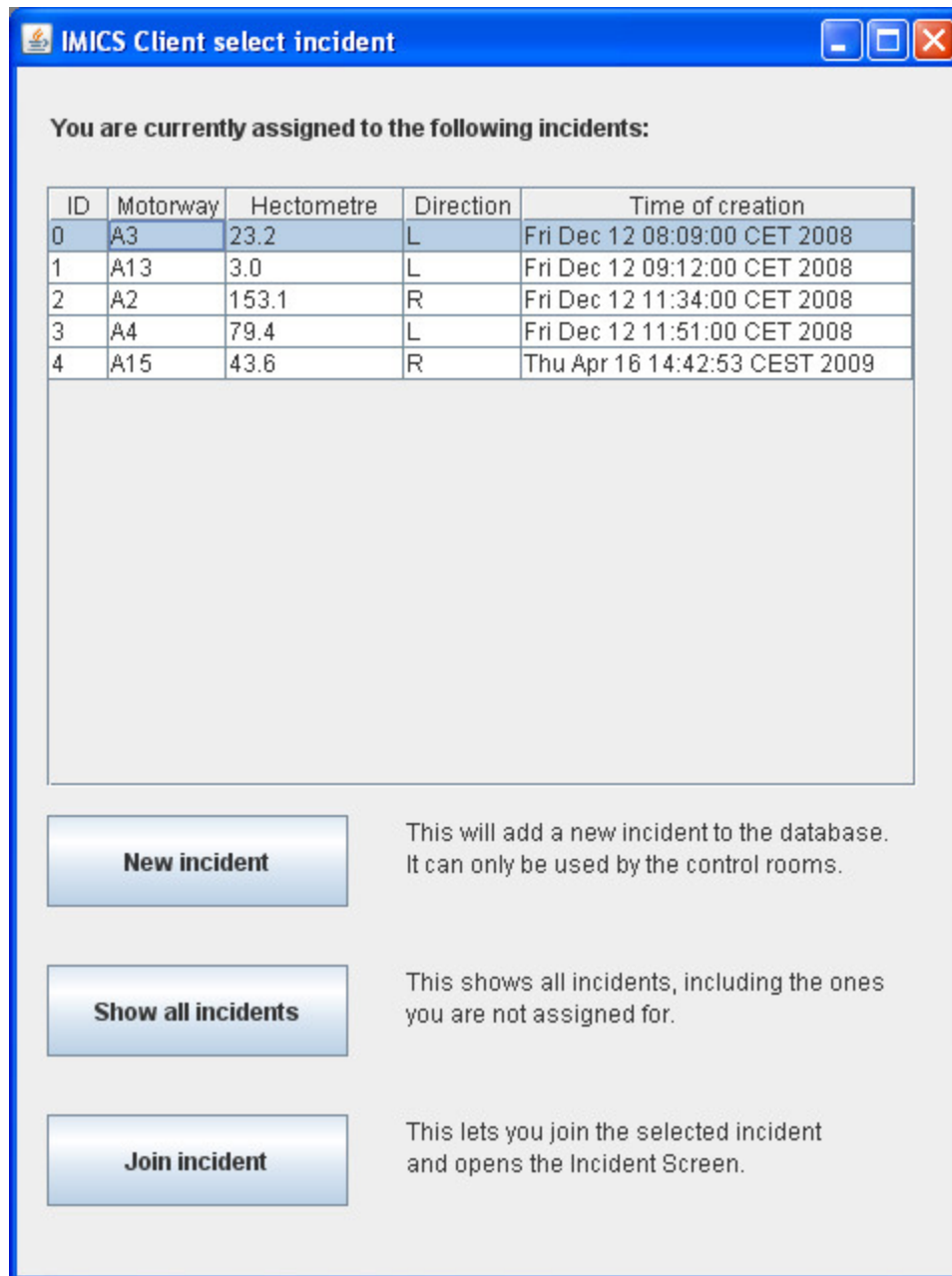


Figure 14 - Screenshot of the list of incidents in the prototype.

As can be seen, incidents are identified here by a unique id, but also by their exact location and the time the incident was first entered into the system. This list normally only shows incidents in the safety region the user is assigned to. It is possible to show all incidents currently active, this could be necessary when a large incident requires assistance from an adjacent safety region.

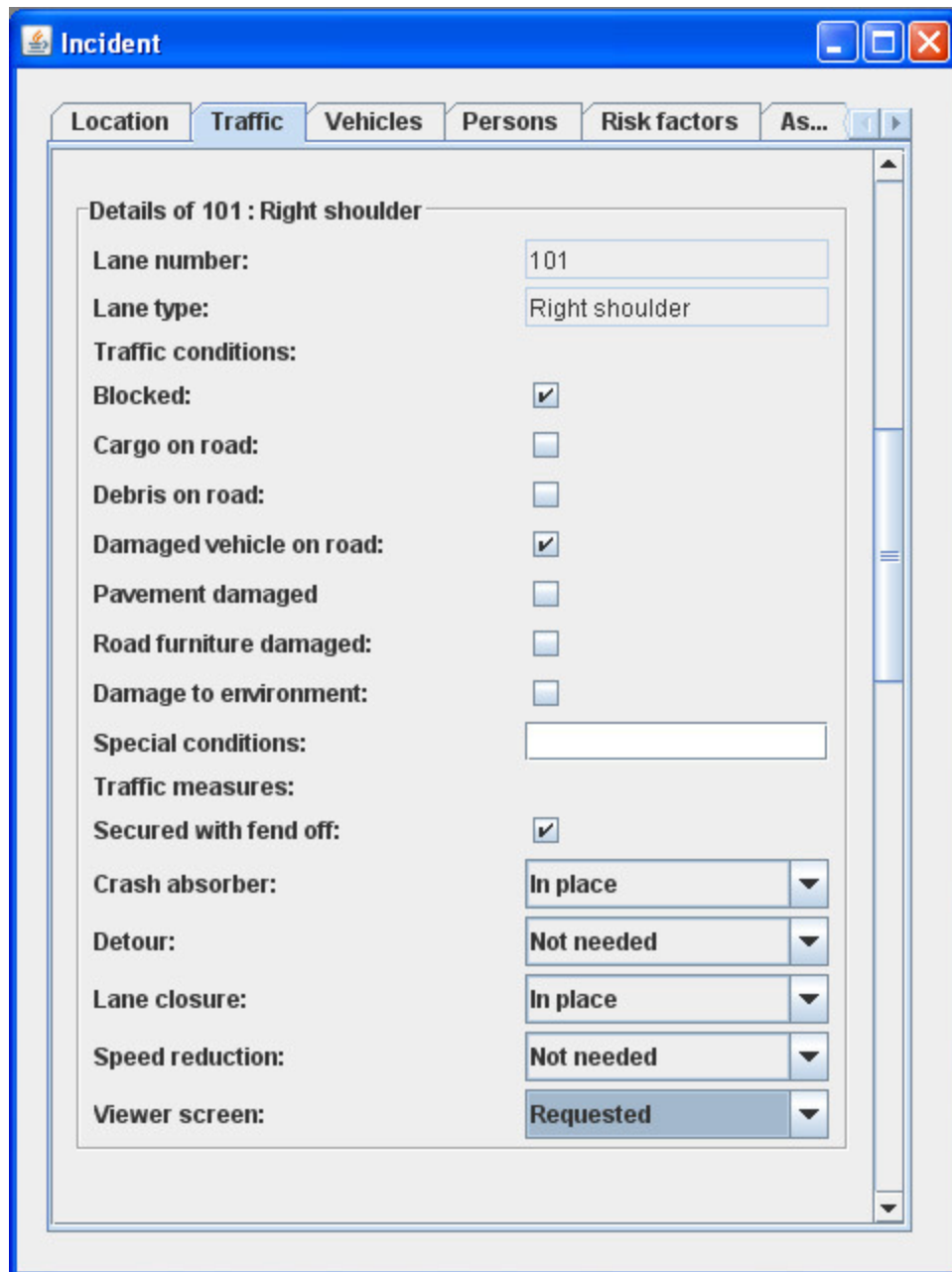
Once an incident is selected or created, the incident data is opened. The incident data is structured in different sections such as vehicles and persons. These sections are displayed in Figures 15 to 20.

The screenshot shows a software window titled "New incident" with a blue header bar. Below the header is a tabbed interface with six tabs: "Location", "Traffic", "Vehicles", "Persons", "Risk factors", and "As...". The "Location" tab is currently selected. The main area of the window contains a form with the following fields and controls:

- Motorway:** A text input field containing "A13".
- Hectometre:** A text input field containing "14.3".
- Direction or letter indication:** A text input field containing "R".
- Start point:** A text input field containing "Rotterdam".
- Destination:** A text input field containing "Den Haag".
- Amount of lanes:** A text input field containing "3".
- Left shoulder present:** A checkbox that is currently unchecked.
- Right shoulder present:** A checkbox that is currently checked.
- Save:** A large, light blue button with the text "Save" centered on it.

Figure 15 - Screenshot of the location tab of a specific incident.

In the location tab, information on the exact location of the incident is found. Currently, all information has to be entered by hand. IMICS could be coupled to a database of the Dutch road network to automatically determine the amount of lanes and shoulders at a given location (for example the system in use at the Dutch traffic control rooms). Naturally this information is important for all involved stakeholders.



Incident

Location Traffic Vehicles Persons Risk factors As...

Details of 101 : Right shoulder

Lane number: 101

Lane type: Right shoulder

Traffic conditions:

Blocked: ☒

Cargo on road: ☐

Debris on road: ☐

Damaged vehicle on road: ☒

Pavement damaged: ☐

Road furniture damaged: ☐

Damage to environment: ☐

Special conditions:

Traffic measures:

Secured with fend off: ☒

Crash absorber: In place ▼

Detour: Not needed ▼

Lane closure: In place ▼

Speed reduction: Not needed ▼

Viewer screen: Requested ▼

Figure 16 - Screenshot of the traffic tab in the prototype.

The traffic tab shows details about the road and traffic conditions. Per lane detailed information is stored on traffic conditions, road status and traffic measures required and in place. Although important to all stakeholders, this information is especially essential for the Department of Public Works and the traffic control rooms.

Figure 17 - Screenshot of the vehicles tab in the prototype.

The vehicles tab shows detailed information on all vehicles involved in the incident. Is it a car, or a lorry carrying hazardous cargo, can the vehicle move on its own and where exactly is the vehicle located. Although all involved organizations use this information, this tab will mostly be used by the fire brigade and in future extensions by the salvage companies.

Figure 18 - Screenshot of the persons tab in the prototype.

The victims of the incident are stored in the persons tab. If there are no significant injuries, just the name and contact information of the victims are stored. In case of heavy injuries, this tab can be used to provide additional details. Naturally, this information is most important for the ambulance service.

The screenshot shows a software window titled 'Incident' with a blue title bar and standard Windows window controls. Inside the window, there are several tabs: 'Location', 'Traffic', 'Vehicles', 'Persons', 'Risk factors', and 'As...'. The 'Risk factors' tab is currently selected. The main area of the window contains a list of risk factors, each with a label and a corresponding input field or checkbox:

- Fire:** ☐
- Fire description:**
- Weather conditions:** ☒
- Weather description:**
- Hazardous materials:** ☐
- GEVI-code:**
- Substance type:**
- Hazard type:**
- Other indications:**
- Safe distance:**
- Wind direction:**

At the bottom of the form area, there is a large, light blue button labeled 'Save'.

Figure 19 - Screenshot of the risk factors tab in the prototype.

Certain complications can make work at the incident scene more difficult or even dangerous. Especially hazardous materials can hamper the incident response significantly. Although these risks do not occur very often, if they do their impact on all involved stakeholders is large. For this reason, the design incorporates a special tab dedicated to them.

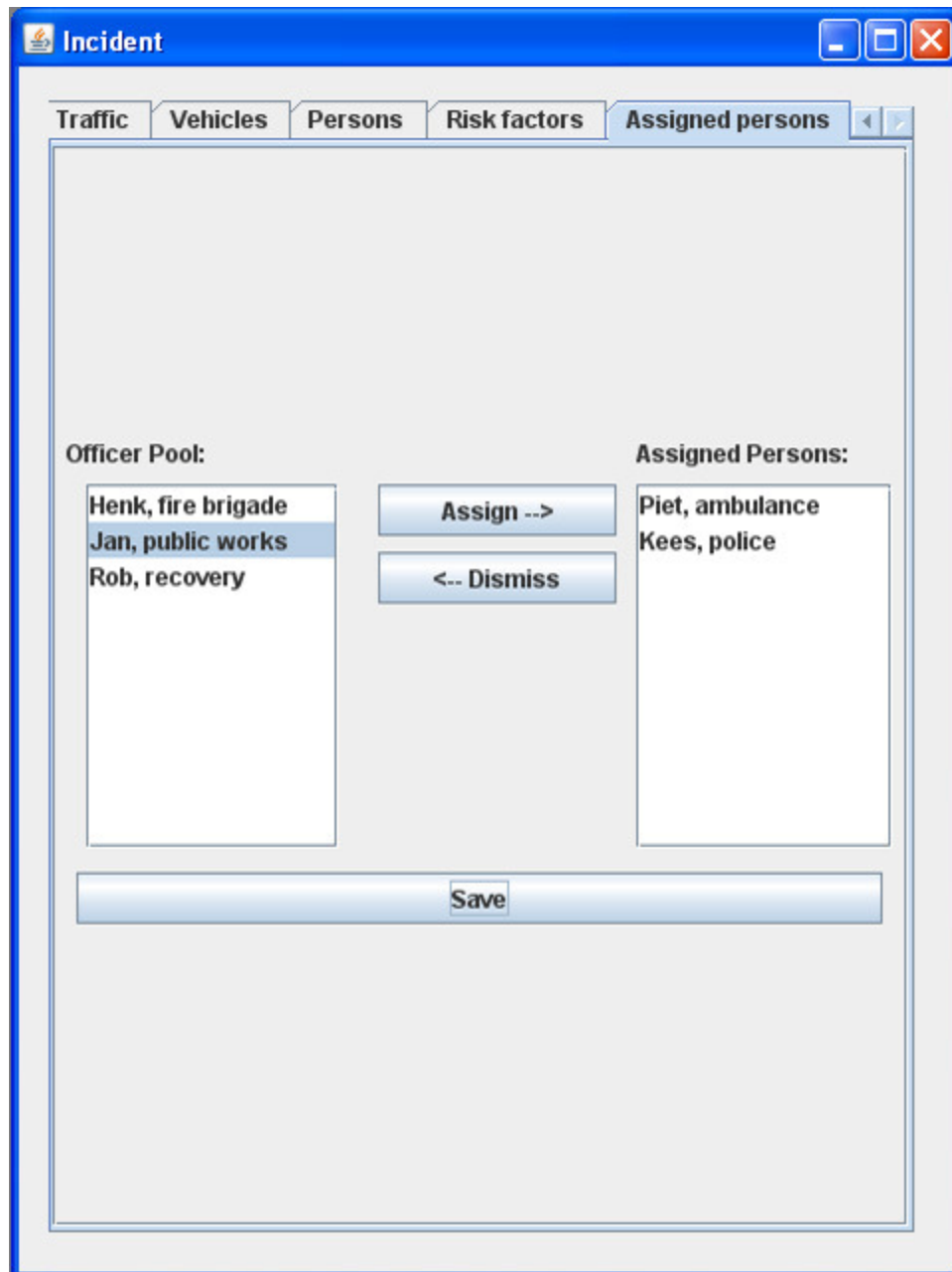


Figure 20 - Screenshot of the assigned persons tab in the prototype.

Finally, in the assigned persons tab, all personnel involved in the incident are shown. It is possible to add persons from the officer pool when they arrive, or when a user has completed his task, he can be dismissed. This will mostly be used by the control rooms as they dispatch people to the incident site, or when users have finished their tasks.

Depending on the user and status of the incident, accessing and changing data may be disabled. The incident must not be closed and a user must be assigned to the incident as discussed above (Section 5.2.2).

A detailed description of the data stored in the system and the way it is structured is described in Section (6.3.3 Ontology).

5.4 User roles

The system users will be the field officers engaged in the incident response and the people in the emergency control room working to resolve the incident situation. The field officers are primarily people from the police, fire brigade, ambulance service and the Department of Public Works. In the current design of IMICS, all users can read the data in the system for the incidents they are assigned to (except for the specialized sections that can only be accessed by personnel from a specific organization as discussed in Section 5.3). Entering information is somewhat less straightforward though. The system's functions are divided in different roles, with different tasks and rights. In the current design, the system supports the three roles shown in Table 13.

Table 13 - The different roles supported by IMICS.

Field officer

A field officer is any person involved with the incident response at the incident site. Field officers can access the information in the system to get a clear picture of the incident. The system also keeps track of the tasks of each field officer and the progress made in the recovery process. Depending on the type of access rights a field officer could also add data to the system (this is discussed in more detail below). All field officers could for example update their progress in the system by indicating which tasks have been completed.

Control room

The control room is where the first information on an incident is received. The control room is generally responsible for creating a new incident instance in the system and for assigning and dispatching the right persons to the incident. The control room has complete read and write access to all incidents in the region, as it is automatically assigned to them.

Director

The director can be a field officer, usually of high rank, or a person at the control room, depending on the type and scale of the incident. The director is responsible for the appropriate handling of the incident. His task is coupled with the monitoring and coordination of the activities at the incident site.

Usually, the director role is given to the one in charge of the incident response. If there is a fire involved, this will be the superior of the fire brigade, in other cases it will usually be the senior police officer. In the IMICS system, the director is responsible for the correctness and up-to-dateness of the information in the system. When certain tasks take too long to complete, he inquires into the reason and tries to speed up the work. The director regularly checks if field officers have updated the status of their tasks in the system and is able to update the data for them.

The director role can be reassigned during incident management (e.g. when the first respondent on scene has to fulfil the direction role until the responsible person has arrived at the scene). The director has full read and write access, the director role must always be fulfilled and there can never be more than one director. Since many incidents involve material damage only (UMS), often only the road inspector is involved. In this case, the director role is very limited and may be fulfilled by the single field officer at the site or by the control room.

In the design discussed so far, the details of the access rights and responsibilities of the roles are not yet fully established. A few alternative options have been discussed with people from the field (see Appendix B). The different options and their advantages and disadvantages will now be discussed in more detail before a final choice is made.

Option 1: The director is responsible for updating data in the system.

In this scenario, the director is the only role with write rights. All other users could suggest updates to the director, but not directly manipulate data in the system themselves.

Advantages:

- 1 Since all data is updated by one person, it is clear who is responsible for the data. (This also helps with liability).
- 2 The other field officers can continue their jobs with little interruption.
- 3 System administration and synchronization is simplified because there is only one role that can update information.
- 4 The director role could be fulfilled by specially trained persons. Their training and experience make them enter new data into the system more efficiently and with less ambiguity.

Disadvantages:

- 1 The director has to check up on every field officer's progress.
- 2 If the incident is complex, involving many field officers, managing the information in the system will become a very daunting task. The director could easily be overwhelmed by the amount of updates he has to keep track of and enter into the system.
- 3 Having an extra person work at the scene, only to keep the information in the system up-to-date, is a drastic (and expensive!) change to the current procedures. It would be more realistic to combine the director task with the coordination of work at the scene. In this case keeping track of all information could easily come in the way of the coordination process or the other way around.

Option 2: The shared control room is responsible for updating data in the system.

In this scenario, the shared control room is responsible for updating the information in the system. Other users can suggest updates and users can be contacted by the control room for up-to-date information, but no one at the incident scene can directly manipulate data in the system.

Advantages:

- 1 The people at the control room are in a controlled environment; they are not distracted by events in their surroundings as much and have the time to input information into the system.
- 2 The control room has access to desktop computers, which means the means to input data are much less restricted.
- 3 The control room has a secure wired connection to the server, which means that if a PDA loses its connection to the network, the data input to the system will reach all other PDAs without delay. (However it still has to reach the control room first).
- 4 Since the shared control room is also responsible for creating the incident instance, the emergency control room should have the possibility to insert data into the system anyway.
- 5 Since all data is updated by the control room, it is clear who is responsible for the data. (This also helps with liability).
- 6 System administration and synchronization is simplified because all information is updated at one location.
- 7 The other field officers can continue their jobs without interruption.
- 8 The control room personnel could receive specialized training. Their training and experience make them enter new data into the system more efficiently and with less ambiguity.

Disadvantages:

- 1 No one at the incident scene can directly input new data to the system.
- 2 The information has to be communicated from the incident location to the control room before being updated in the system by the control room. This extra communication step is one of the things the system intended to eliminate in the first place to reduce communication delays and errors. The distance between the control room and the incident location could mean new information takes longer to get into the system. Since the people at the emergency control room only have an indirect view of the incident, their view of the situation, and therefore the information in the system, could be incomplete or worse, incorrect.

Option 3: Everyone can update information in the system.

This scenario allows everyone to update information in the system.

Advantages:

- 1 Making people responsible for their own information keeps them involved.
- 2 The people that update the information are also the most aware of the status of the process, so the information provided is generally the most accurate in this scenario.
- 3 Since the work load is spread over all people involved, the risk of people getting overwhelmed by the updating of data is greatly reduced.
- 4 This approach does not require any extra persons to get involved.
- 5 Since everyone is involved in keeping track of the information in the system, errors are potentially found and corrected earlier.

Disadvantages:

- 1 It may be unclear who is responsible for updating certain information.
- 2 Synchronization is more complex, since everyone can modify all information.
- 3 Especially field officers may be too occupied with their primary tasks to keep track of their progress in the system. (When a person is in critical need of medical attention, bookkeeping has low priority).

Final solution: A compromise between the above.

The approaches mentioned above are very one-sided. In practice a well balanced compromise could enhance the system's usability and performance. Although the most desirable approach may differ in each incident, a solution that is generally acceptable can be devised.

First a distinction can be made between general incident information (e.g. location, number of vehicles involved, etc.) and task information (e.g. has the incident site been secured, have the wounded been treated?). The proposed solution is to make every user responsible for the status of his tasks and the incident data these tasks require primarily. Although users can supply any information if they so desire, they are only responsible for information on the tasks they are involved in. In this way, facts in the system are generally supplied by the people most aware of them, but can be supplied by others if the situation requires it. To ensure this does not interfere with the tasks themselves, it must be very simple to mark the completion of a task and entering information into the system must take as little time as possible.

The director will have the final responsibility for the task information, this means the director should monitor progress and if a user is unable to enter his progress into the system, the director will complete the information in the system. In this way, the work

load is divided over the users without spreading responsibilities too much. The users can skip updating the information if the incident situation requires direct attention and the director is not overwhelmed by the amount of information he has to manage. The system could be extended with automated alerts when for example progress of a user appears to be slow, or required data is missing. Chapter 9, Future work, briefly describes techniques that could help to identify points of attention.

In practise, much information is received by the control room since most incidents are reported here first. It therefore makes sense to make the control room responsible for the general information on the incident. Although additional information is usually first perceived at the incident site, the personnel at the control room have the time to keep an overview and to contact the right persons when extra information is required.

These options should be tried in a field test before the system is actually adopted. This was not possible in the scope of this thesis and therefore the options were discussed with experts from the field. The system is also designed in such a way that adapting the roles and their rights will be relatively simple if required.

5.5 Summary

Summarizing this chapter, the global design of IMICS has been completed, and partially implemented. As the basis of the system, a server and client agent have been implemented that run on the JADE framework. Multiple client agents can connect to one server, forming a blackboard like system.

Multiple subsystems have been designed to run on the blackboard infrastructure. The IMICS design includes a task overview, organization overviews for the primary stakeholders, a geographic information system, a communication part and a general incident information part. As a complete implementation is beyond the scope of this project, only one part has been implemented. The incident information part has been implemented since it provides a good overview of the incident situation. This part of the system is also the most general, providing the same information and functionality to all users.

There are three roles in the IMICS system: field officers, control rooms and the director. All IMICS users are all responsible for keeping track of their own tasks. The director is also responsible for keeping track of the other users assigned to the same incident, and should intervene if required by the situation. The control rooms are generally responsible for creating a new incident in the system, entering incident information in the system and assigning persons to an incident. All user types can perform all these tasks in the system for the incidents they are assigned to though, so the distinction is mainly in responsibility. All users can only access the incidents they are assigned to, except for the control rooms. The control rooms are automatically assigned to all incidents in their region.

Chapter 6

Detailed design

As discussed in Chapter 5, the IMICS system currently consists of a client part and a server part. The functionality of these parts is provided by a client agent and a server agent. These agents run on a JADE container that has to be activated first. To run the IMICS application, the server application and at least one client application must be started. The following sections describe the inner workings of the client and server system and how they and the JADE container are initialized.

6.1 Server system

The IMICS server application can be run by running the *imicsServer.jar* executable, or by running 'java imicsServer.java' from the command line. When the server is started it runs *Main.java*. From here, the *LoginDialog* is shown where the user must enter correct authentication credentials as is shown in Figure 21.



Figure 21 - The login dialog.

After the user has successfully signed in, the main class calls *MainApp.java*. Here, the JADE runtime is initialized, and the *ServerAgent* is created. After JADE and the agent have been initialized, the agent is activated in the *run()* method.

Server Agent

The server agent is responsible for registering new client agents with the directory facilitator of the JADE framework and sending them the up-to-date incident data. It is the central point of the IMICS system through which all messages and updates are sent. The server agent is therefore responsible for forwarding updates to other clients. The server also serves as central storage, keeping track of all incidents currently open and storing information about assigned persons. All received data is also stored in a log. Currently, the server system has no further user interface as all interaction with the users is provided by the client system.

Although creating a new incident is initiated at the client side, the server is responsible for creating the actual incident object and creating a new unique incident ID. This new ID is then sent to the client requesting the new incident. In this way, the risk of inconsistent data is reduced. This remainder of this section describes the implementation details of the server agent.

Initialization

When the server agent is started, it first initializes its required variables. The server agent keeps track of a number of variables. First of all, it keeps track of the different agents connected to the server. These agents are identified in JADE by a specialised class *AID*. This class maintains a globally unique name and a number of addresses with which agents on different platforms can be found. These are stored in an array called *clientAgents*.

The server also keeps track of the different incidents that are currently active in an ArrayList called *incidents*. Incidents are stored in the *Incident* class that is part of the ontology that has been specifically designed for IMICS. Details of this class are described in Section 6.3.3.

Finally, the server keeps track of the persons that are available in the officer pool. The personnel of the emergency services are stored in a class called *Professional*. These are stored in an ArrayList called *officerPool*.

Figure 22 shows the class diagram of the server agent.

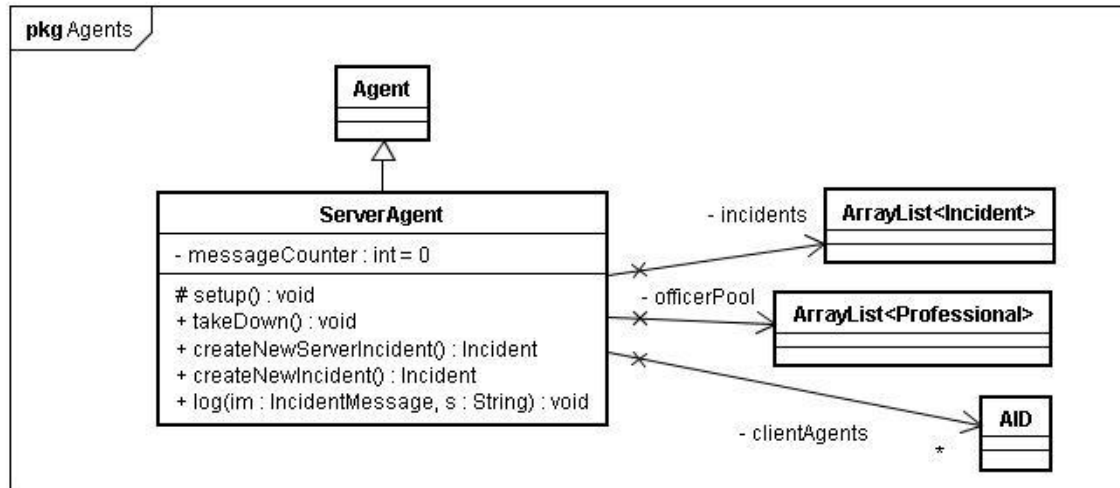


Figure 22 - The server agent class diagram.

After initialization, the server agent runs its *setup()* method. Here it fills the officer pool and the list of incidents with the data stored previously. For the proof of concept, this is actually fictive data, hard coded into the system. In a fully implemented system, the server should never shut down (or at least one server should always be available). The officer pool should be kept up to date by tracking personnel and the incidents that are not yet resolved are automatically tracked when personnel updates information. When an incident is resolved, it must be closed by the responsible person (usually the director). This option is not implemented for the proof of concept though. After the required data has been initialized, the server agent registers as “serverAgent” with the JADE Directory Facilitator so that other agents can find it.

Activation of behaviours

In the final part of the setup procedure, the actual server behaviour is activated. This is done by adding the behaviour *ReceivingBehaviour* to the agents list of active behaviours.

Behaviours description

The different behaviours are described below.

ReceivingBehaviour

The *ReceivingBehaviour* is activated on startup of the server agent. It is a cyclic behaviour that remains active until the agent is shut down. It checks the type of messages received by the server, writes the received data to the server log and activates the proper response. Table 14 shows the kind of messages the *ReceivingBehaviour* responds to and what kind of response is activated.

Table 14 - The types of messages received by the server and its response.

Subject	Response
<i>ClientExistenceNotification</i>	The server activates the <i>SendInitialUpdateBehaviour</i> with the original sender as parameter.
<i>ClientToServerUpdate</i>	The server activates the <i>ForwardUpdateBehaviour</i> with the received update message as parameter.
<i>ClientToServerOfficerPoolUpdate</i>	The server activates the <i>ForwardOfficerPoolUpdateBehaviour</i> with the received update message as parameter.
<i>RequestNewIncident</i>	The server activates the <i>NewIncidentBehaviour</i> with the received message as parameter.
Other	The server displays the error message: "IncidentMessage subject not understood" if a different subject is received.

SendInitialUpdateBehaviour

This behaviour is activated every time a *ClientExistenceNotification* is received. This means a new client has registered with the server. In response, the server first updates its list of client agents. It then sends the *InitialUpdate* type of message, that contains a copy of the current incident data at the server and a *ServerOfficerPoolUpdate* message that contains a copy of the current officer pool data at the server.

ForwardUpdateBehaviour

The *ForwardUpdateBehaviour* is activated when a *ClientToServerUpdate* is received from one of the clients. It first reads the updated data from the received message and stores it at the server. The server then updates the list of active clients to make sure it is up to date before forwarding the received data in a *ServerUpdate* message to all client agents except the original sender of the update.

ForwardOfficerPoolUpdateBehaviour

This behaviour responds to *ClientToServerOfficerPoolUpdate* messages. It is basically the same as the *ForwardUpdateBehaviour*, except that the update does not contain incident data, but the officer pool data. It first reads the officer pool data from the received message and stores it at the server. It then updates its list of active clients and forwards the received data in a *ServerOfficerPoolUpdate* to all client agents except the original sender of the update.

NewIncidentBehaviour

After receiving a *RequestNewIncident* message, this behaviour creates a new *Incident* object at the server and sends it to the original sender in a *NewIncident* message.

Agent shutdown

Although a server agent should always be available, it is possible that a server must be shut down (e.g. for maintenance). When a JADE agent is shut down, its *takeDown()* method is run before it is terminated. Here, the server agent deregisters itself with the

JADE DF. Although this has currently not been implemented, a check should be performed here that at least one server agent is still operational. Should the server agent be the only one active, before it can be shut down, a new server agent must be started to guarantee an active server is available. If the entire server computer must be shut down, this new server agent must be activated on a different JADE container on a different computer.

Future additions

In Section 6.4 a number of features is discussed that should be implemented before the system is deployed. In Chapter 9, Future work, additional functionality that could be added to the server system to further improve the system's impact on incident management is discussed.

6.2 Client system

The IMICS client application can be run by running the *imicsClient.jar* executable, or by running 'java imicsClient.java' from the command line. When the server is started it runs *Main.java*. From here, just as with the server, the *LoginDialog* is shown (see Figure 21).

After the user has successfully signed in, the main class calls *MainApp.java*. The *IPDialog* as shown in Figure 23 is now displayed. Here the user must provide the IP address (or the local name if used in a private network) of the server computer to connect to. If desired, the user can supply a port number through which the connection should be made, but this is optional. The default port number is 1099.

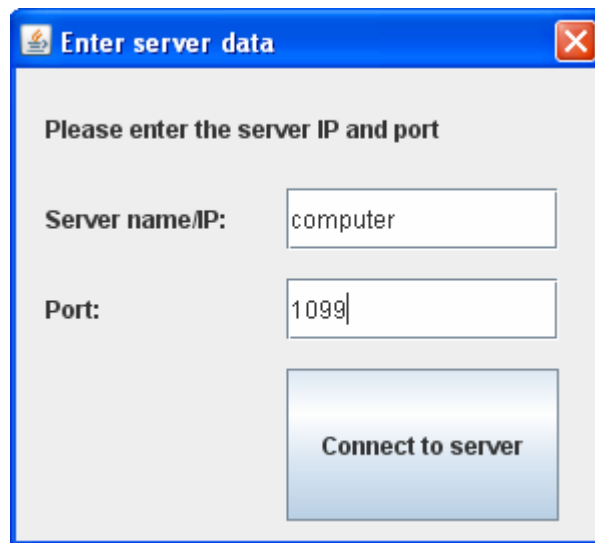


Figure 23 - The IP Dialog.

The JADE runtime is now initialized, connecting to the JADE main container at the IP address and port supplied, and the *ClientAgent* is created. After JADE and the agent have been initialized, the agent is activated in the *run()* method.

Client Agent

The client agent registers with the server, receives the latest incident data and provides a user interface in which the user can read or modify data. The implementation details of the client agent are now described.

Initialization

Just as the server agent, the client agent keeps track of a number of variables. First of all, it keeps track of the active servers (in the proof of concept, there is only one server). The server too, is identified by the *AID* class. The servers are stored in an array called *serverAgents*.

Just as the server agent, the client agent keeps track of the active incidents in an *ArrayList* holding *Incident* objects called *incidents*. The data in this list is received later from the server agent. When used in practice, the client should only keep track of the incidents the user is assigned to, reducing the use of system resources. For the proof of concept, the amount of incidents in the system is limited though and the complete list of incidents is copied.

The same holds for the officer pool. Just as the server, the client agent has an *ArrayList* of *Professional* objects called *officerPool*. This data is also received from the server agent later.

The client agent also provides a user interface to enable interaction with its users. The interface is initiated in a class called *SelectIncidentFrame* that is an extension of the Java *Frame* class. The client agent holds a reference to this interface to be able to access the different fields provided by the interface and to be able to respond to events generated by the interface. This reference is stored in a variable *selectIncidentFrame* of type *SelectIncidentFrame*.

The class diagram of the client agent is shown in Figure 24.

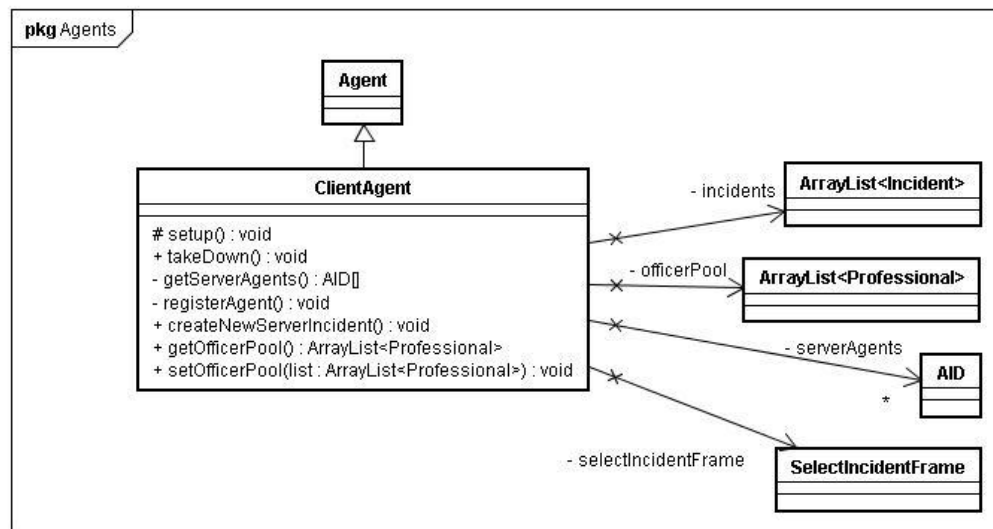


Figure 24 - The client agent class diagram.

After initialization, the *setup()* method creates an instance of the *SelectIncidentFrame* with a reference to the *clientAgent* and the *incidents* object, and displays it on the screen. The agent then registers with the server DF through the *registerAgent()* method.

Activation of behaviours

The *NotifyServerBehaviour* is activated to notify the server of the activation of a new client agent and to receive the latest incident data and officer pool. Finally, the *ReceivingBehaviour* is activated to handle incoming messages.

When the user saves information in the system using buttons in the user interface, one or both of the update behaviours are activated, depending on the information the user wishes to store.

Behaviours description

The behaviours of the client agent are described below.

NotifyServerBehaviour

This behaviour is executed once, when the client agent starts. It updates the list of servers and sends a *ClientExistenceNotification* to them to ensure all servers are aware of the new agent and to receive the latest data from them.

ReceivingBehaviour

At the client, the *ReceivingBehaviour* has the same purpose as at the server. It is activated on startup of the client agent and remains active as a cyclic behaviour until the agent is terminated. The *ReceivingBehaviour* checks what kinds of messages are received by the client, writes the received data to the client log and activates the correct behaviour in response. Table 15 shows the type of response to each type of message.

Table 15 - The types of messages received by the client and its response.

Subject	Response
<i>InitialUpdate</i>	The incident data received from the server is stored locally by the client and the user interface is refreshed to display it correctly.
<i>GUIUpdate</i>	The client activates the <i>ClientToServerUpdateBehaviour</i> with the current incident as parameter.
<i>GUIOfficerPoolUpdate</i>	The client activates the <i>ClientToServerOfficerPoolUpdateBehaviour</i> .
<i>ServerUpdate</i>	The client checks if the incident received in the update is already stored locally. If so, it is updated with the new data, if not, it is added to the local list of incidents. Finally the user interface is updated.
<i>NewIncident</i>	The client reads the received incident from the message and adds it to the local storage.
<i>ServerOfficerPoolUpdate</i>	The client updates the local officer pool information.
Other	The client displays the error message: "IncidentMessage subject not understood".

ClientToServerUpdateBehaviour

This behaviour is activated in response to a *GUIUpdate* message generated by the clients user interface. It updates the list of servers and sends a *ClientToServerUpdate* message containing the updated incident data to the server so the server can forward it to the other clients.

ClientToServerOfficerPoolUpdateBehaviour

This behaviour is similar to the former. When the client's user interface generates a *GUIOfficerPoolUpdate* message, the list of servers is updated and a *ClientToServerOfficerPoolUpdate* message is sent to the server containing the updated officer pool data.

createNewServerIncident

When the user creates a new incident with the user interface, a *RequestNewIncident* message is sent to the server. The server in response creates a new incident object and sends its information to the clients. This is actually a method of the *clientAgent* class. The client system automatically opens the new incident. Other users can not see the new incident until it has been saved.

Agent shutdown

When a client agent is shut down, its *takeDown()* method simply deregisters with the DF and the agent terminates.

Future additions

Once more Section 6.4 discusses some required features that were not implemented. The client agent could also be improved or extended in the future. Some ideas are provided in Chapter 9, Future developments/additions.

6.3 Agent communication

As discussed above, the IMICS agents respond to messages received from other agents and send messages themselves. This section takes a close look at these messages. Section 6.3.1 describes what kinds of messages are sent by the client and server agents, to whom they are sent, and what their general content is. How the information is processed so JADE can handle them is discussed in Section 6.3.2. Finally, Section 6.3.3 provides a detailed view of the information that is sent in the messages and the way this information is structured.

6.3.1 Agent messages

Tables 16-26 show the IncidentMessage objects sent by the IMICS agents. These messages are all wrapped in an ACL message sent with the INFORM performative.

Table 16 - The ClientExistenceNotification message.

Message Subject:	<i>ClientExistenceNotification</i>
Sender	Client
Sent in behaviour	<i>NotifyServerBehaviour</i>
In response to	Client startup
Receiver	Server
Content	None

Table 17 - The ClientToServerUpdate message.

Message Subject	<i>ClientToServerUpdate</i>
Sender	Client
Sent in behaviour	<i>ClientToServerUpdateBehaviour</i>
In response to	<i>GUIUpdate</i>
Receiver	Server
Content	1 Incident object

Table 18 - The ClientToServerOfficerPoolUpdate message.

Message Subject	<i>ClientToServerOfficerPoolUpdate</i>
Sender	Client
Sent in behaviour	<i>ClientToServerOfficerPoolUpdateBehaviour</i>
In response to	<i>GUIOfficerPoolUpdate</i>
Receiver	Server
Content	An arrayList of Professional objects (the officer pool)

Table 19 - The RequestNewIncident message.

Message Subject	<i>RequestNewIncident</i>
Sender	Client
Sent in behaviour	<i>createNewServerIncident*</i>
In response to	Creating a new incident in the client GUI
Receiver	Server
Content	None

* createNewServerIncident is not a behaviour, but a method

Table 20 - The InitialUpdate message.

Message Subject	<i>InitialUpdate</i>
Sender	Server
Sent in behaviour	<i>SendInitialUpdateBehaviour</i>
In response to	<i>ClientExistenceNotification</i>
Receiver	Client
Content	An arrayList of Incident objects (all incidents)

Table 21 - The ServerOfficerPoolUpdate message sent to a newly connected client.

Message Subject	<i>ServerOfficerPoolUpdate</i>
Sender	Server
Sent in behaviour	<i>SendInitialUpdateBehaviour</i>
In response to	<i>ClientExistenceNotification</i>
Receiver	Client
Content	An arrayList of Professional objects (the officer pool)

Table 22 - The ServerOfficerPoolUpdate sent to all assigned clients after an update.

Message Subject	<i>ServerOfficerPoolUpdate*</i>
Sender	Server
Sent in behaviour	<i>ForwardOfficerPoolUpdateBehaviour</i>
In response to	<i>ClientToServerOfficerPoolUpdate</i>
Receiver	Client
Content	An arrayList of Professional objects (the officer pool)

* This type of message is sent by two different behaviours in response to different messages

Table 23 - The ServerUpdate message.

Message Subject	<i>ServerUpdate</i>
Sender	Server
Sent in behaviour	<i>ForwardUpdateBehaviour</i>
In response to	<i>ClientToServerUpdate</i>
Receiver	Client
Content	1 Incident object

Table 24 - The NewIncident message.

Message Subject	<i>NewIncident</i>
Sender	Server
Sent in behaviour	<i>NewIncidentBehaviour</i>
In response to	<i>RequestNewIncident</i>
Receiver	Client
Content	1 Incident object

Table 25 - The GUIUpdate message.

Message Subject	<i>GUIUpdate</i>
Sender	Client GUI**
Sent in behaviour	Activated by client GUI
In response to	Saving data in the client GUI
Receiver	Client
Content	1 Incident object

** These messages are created by the GUI (*IncidentFrame.java* and *NewIncidentFrame.java* from package IMICS.GUI) and put directly into the client agent's message queue.

Table 26 - The GUIOfficerPoolUpdate message.

Message Subject	<i>GUIOfficerPoolUpdate</i>
Sender	Client GUI**
Sent in behaviour	Activated by client GUI
In response to	Saving data in the client GUI
Receiver	Client
Content	An ArrayList of Professional objects (the officer pool)

** These messages are created by the GUI (*IncidentFrame.java* and *NewIncidentFrame.java* from package IMICS.GUI) and put directly into the client agent's message queue.

6.3.2 The IncidentMessage object

JADE Agents communicate using ACL messages. The receiving agent has to check what kind of message is received and respond accordingly. Although an ACL message has fields such as receiver, performative and content, there is no general field for a subject or message type. JADE does, however, support topic based messaging. Topics can be seen as a sort of channels with a certain subject. Agents can publish messages to a channel and subscribe to a channel to receive all messages published to it.

Unfortunately I was unable to get topic based messaging to work properly. To cope with this problem, I have created an *IncidentMessage* class. An *IncidentMessage* can contain all incident information (i.e. the list of incidents and the officer pool) and a subject. In this way, the agents can filter the messages on the basis of their subject and perform the correct behaviour corresponding to the type of message received. Figure 25 shows the *IncidentMessage* class.

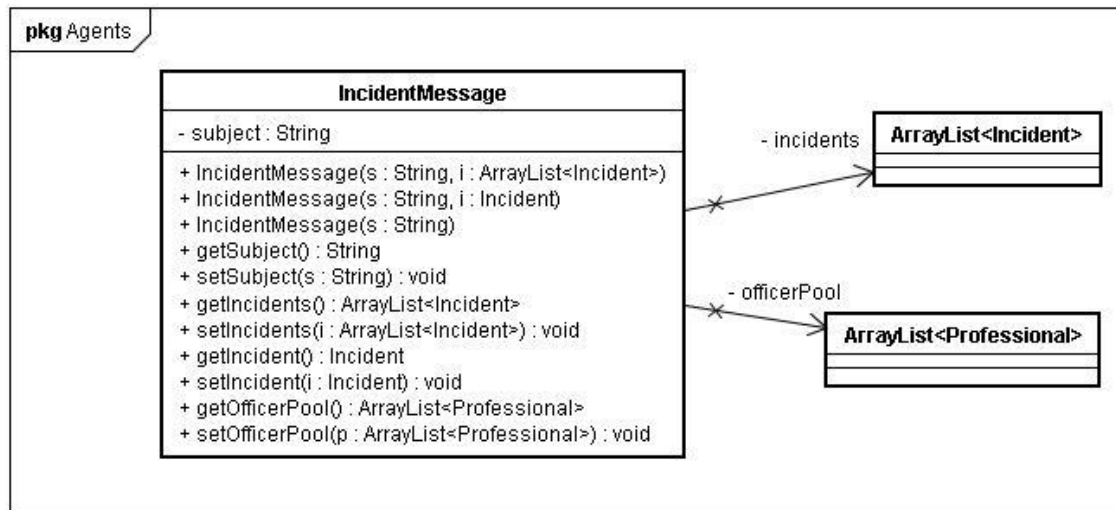


Figure 25 - The IncidentMessage class.

The ACLMessage class, as provided by JADE, provides a wrapper class with which Java objects can be sent as the content of an ACL message. Using the *setContentObject()* method, an *IncidentMessage* object can be directly communicated within an ACL message.

6.3.3 Ontology

Though the text above explains how information is sent by means of ACL messages, it does not describe what incident information is sent. As mentioned in Section 5.3, the information need for most involved stakeholders is quite similar. To provide access to this information in a well structured way, an ontology has been designed.

Let's first define what an ontology is:

“An ontology is an explicit specification of a shared conceptualization that holds in a particular context.” [29].

Or in another definition: *“An ontology is a fundamental form of knowledge representation about the world, or any part and domain of it. Ontology defines the basic constituents and elements of reality: entity kinds, categories, or classes with their constituent relationships, inherent properties and possible instances.”* [30].

So an ontology describes basic concepts in a domain and defines relations among them. The basic building blocks of ontology design are shown in Table 27.

Table 27 - The basic components of an ontology.

1. Classes or concepts
2. Properties of each concept describing various features and attributes of the concept (sometimes called roles)
3. Restrictions on properties (sometimes called role restrictions)

An ontology provides a common vocabulary for experts who need to share information in a specific domain. An ontology together with a set of instances of its classes constitutes a knowledge base. Although ontologies have other uses, in this project an ontology has been created to share common understanding of the structure of information among people and software agents in incident management. In future implementations, automated agents could use the ontology to reason about the incident situation, providing functions such as relevance inference and decision support.

The information requirements of the incident management stakeholders have been derived from the information in [9]. Based on these requirements, the ontology has been designed. It has been created using Protegé, a free, open source, ontology editor and knowledge-base framework [31].

Once the ontology had been designed, usable Java code was generated using the Bean Generator [32]. Some minor adjustments were made to the code to make it easier to use with the user interface of the IMICS agents.

Using a free diagram making tool, JUDE, a complete class diagram of the ontology has been created. Because of its size, the ontology is presented in separate parts. As a result, not all relations can be shown in the diagrams. For this reason, in Figures 27 and 29, if an association with a class outside of the figure is present, this class is shown only by name. The full relations can also be found in the attributes of the classes themselves.

The ontology can be found in the *IMICS.ontology* package. The basis of the IMICS ontology is the *IMICS* object, it implements the *Concept* class as required in a JADE ontology. The classes *Incident* and *Involved_Object* extend the *IMICS* object as can be seen in Figure 26 (fignr checken).

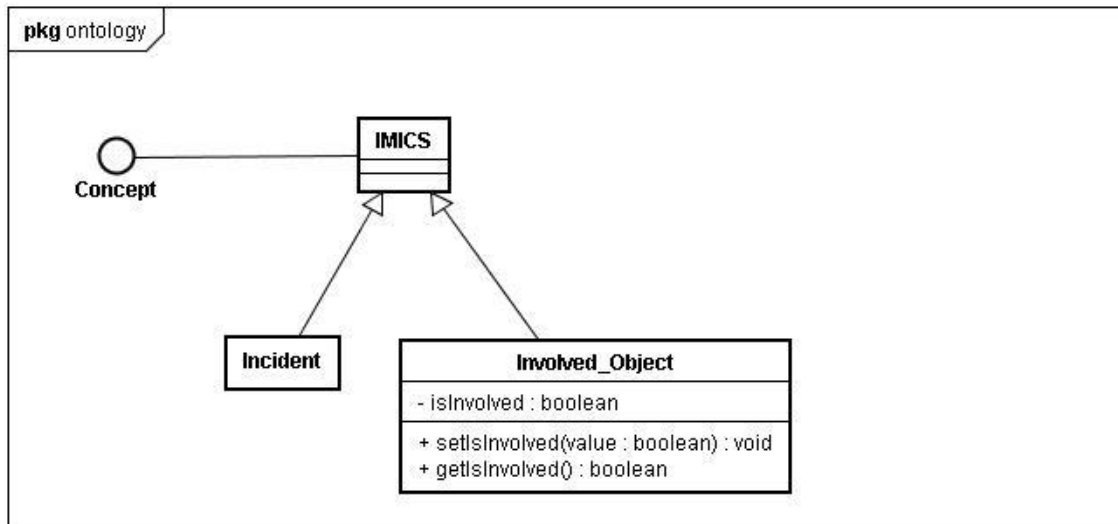


Figure 26 - The IMICS object.

The *Involved_Object* class represents all objects that can be involved in an incident. (These are the *Risk_Factor*, *Cargo*, *Traffic_Measures*, *Location*, *Lane*, *Vehicle*, *Person*, *Hospital*, *Injury*, *Task* and *Role* objects, and their subclasses. The *Involved_Object* class has no relation to the *Incident* class.) It has been created to add an extra feature, to indicate whether an object is still involved in an incident. This feature is not yet used in the current implementation, but could be used for future extensions of the system. It could also be extended with timestamp information to track how long certain events take independent of the tasks in the system.

Incidents

The *Incident* object represents an actual incident in the system. Its class diagram and its relations are shown in Figure 27. As shown, an *Incident* object has a unique *incident_id* by which it can be identified by the client and server agents. To simplify identification of incidents by IMICS users, an incident also has a property called *incident_description*. This field holds a string describing the incident by its location and time of creation in the system. The boolean *incident_is_resolved* is set to true when all work at the incident scene has been completed and the situation has been returned to normal. The *is_closed* variable is set to true when the control room has closed the incident. This should be done after the incident has been resolved and all information has been correctly entered into the system so it can no longer be changed. Note that the latter two values are not yet used in the implemented proof of concept.

The *Incident* object also holds references to the persons and vehicles involved, the incident location and any specific risk factors involved.

Persons

Although both persons, the system discriminates between *Professionals* assigned to resolve the incident, and *Civilians* involved as a victim of the incident. The complete class diagram of the *Person* object and its related objects is shown in Figure 28.

Persons are stored in the system with their personal details as shown in the class diagram above. The *person_ID* is a unique identifier that is created at the server, the *person_number* identifies persons within an incident. The distinction between the *Civilian* and the *Professional* can be clearly seen. Where a civilian's personal information is stored to support medical care and to enable future reference (e.g. for insurance, law cases etc.), the system requires more information about the professionals. The tasks and roles taken by the involved personnel are stored in the system for reference. The further specification in *Ambulance*, *Fire_Brigade*, *Police*, *Public_Works* and *Recovery* can be used to assign tasks and roles and can also be used for managing access to parts of the system as described in Section 5.3.

Resolving an incident involves certain tasks that have to be divided over the personnel involved in the incident. Examples are extinguishing a fire or securing the incident site. Therefore, the *Professional* class has a reference to the *Task* class. Tasks are identified by a description and have a field for the time the task was started (*time_started*) and for the

time the task was finished (*time_finished*). They also have a boolean field *is_finished* to indicate the task is in fact finished.

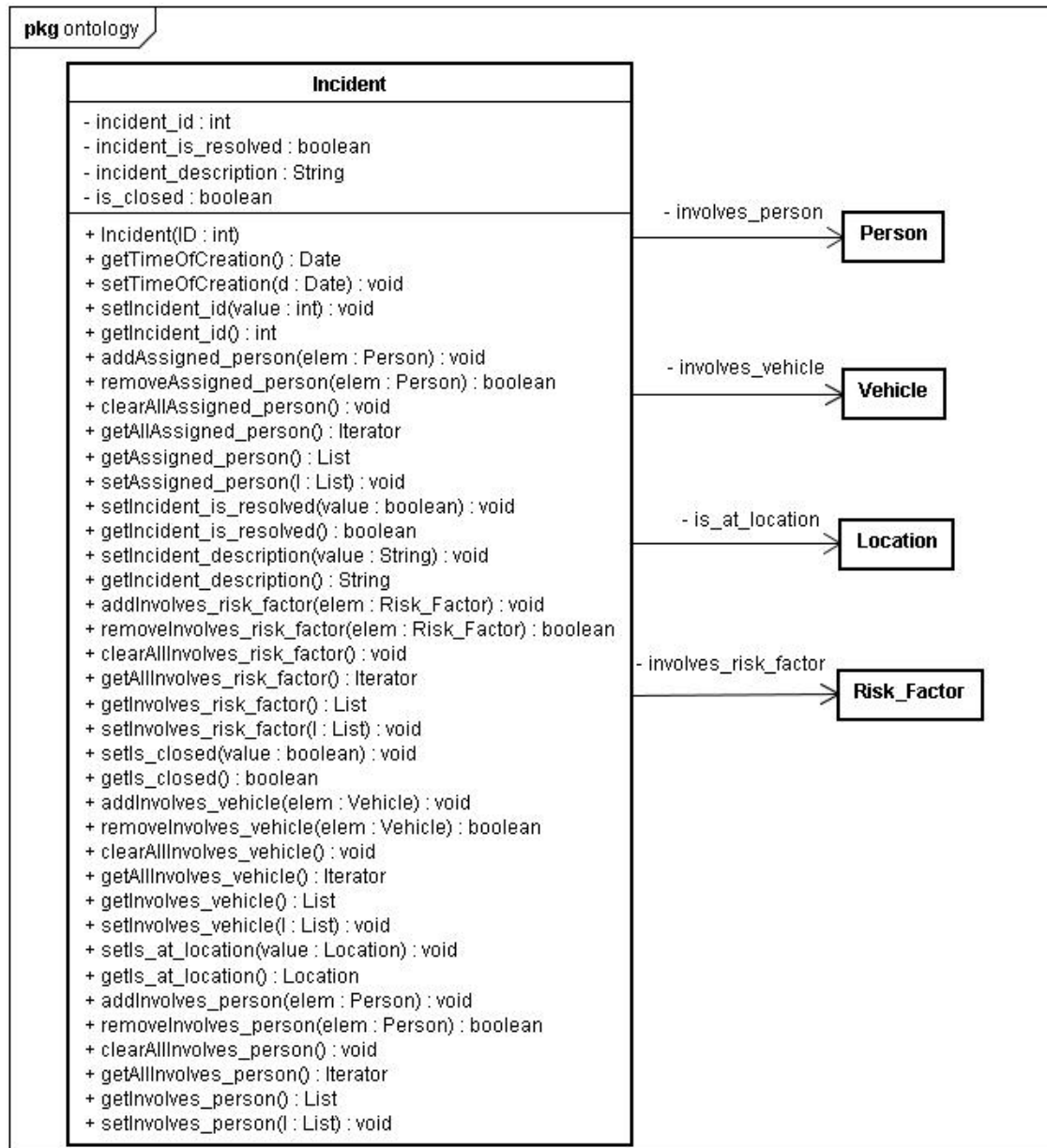


Figure 27 - The incident class.

Professionals can also be given a specific role. One person (usually the one highest in command) will be given the director role as described in Section 5.4. In the current design, the *Role* object only has a description. For the proposed hierarchy overview for each emergency service, as described in Section 5.3, the *Role* object could be extended though. (For example, in the current IMICS design, the control rooms should always be assigned. This could be simply implemented by a specialised role for the control room.)

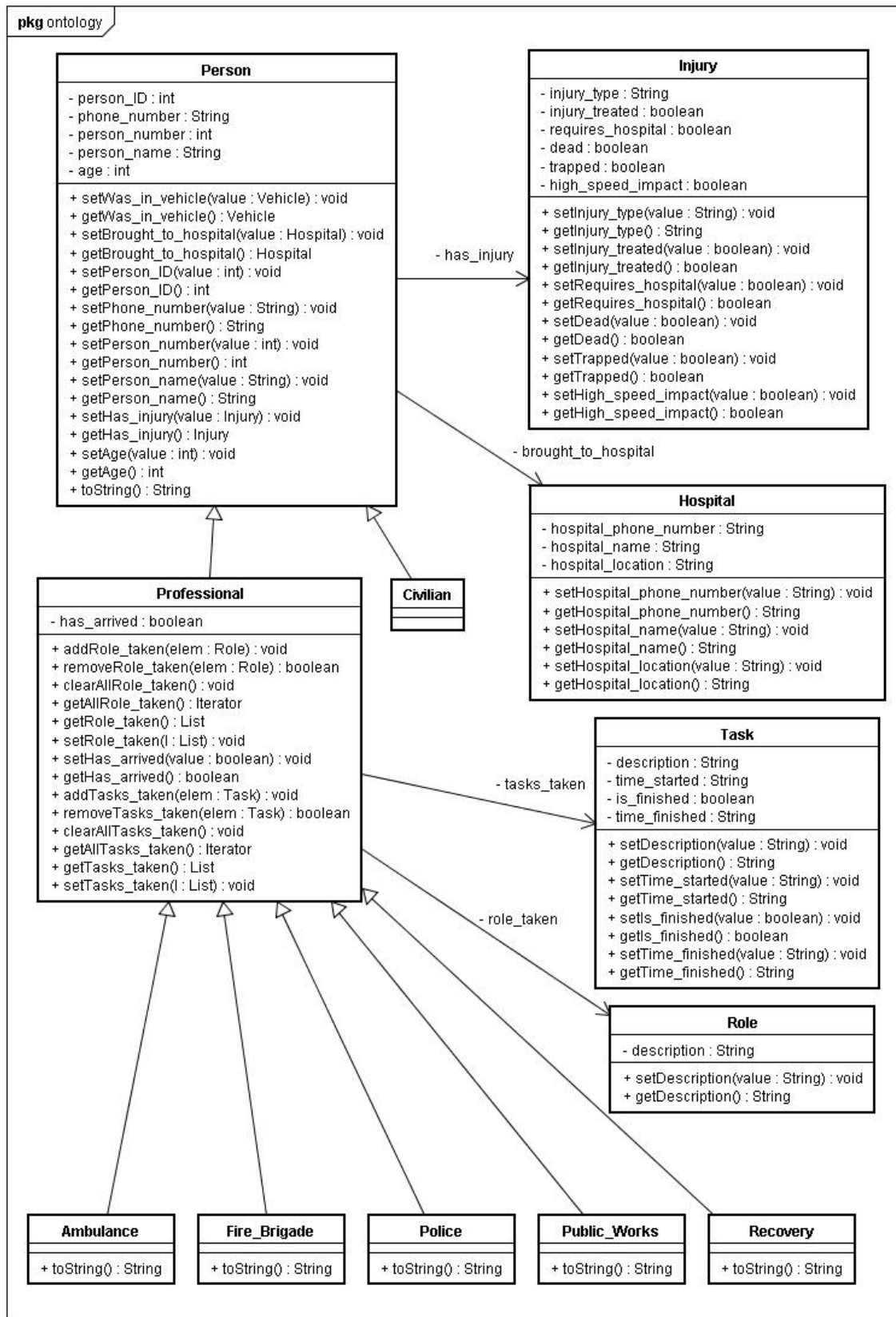


Figure 28 - The person class and its relations.

For the involved civilians their injuries can be described in the system if required. A professional can also get injured at the incident site though, therefore, the *Injury* class is directly linked to the *Person* class. It provides a textual description of the injury in the *injury_type* property. However, it also provides boolean values to quickly give an indication of the type and severity of the injuries. The *injury_treated* field is set to true when the injuries require no more direct attention and the *requires_hospital* is set to true when an injury is so severe that the victim must be taken to a hospital. The *dead* and *trapped* fields are set to true when a victim is deceased or is trapped in a vehicle. Finally, the *high_speed_impact* field is set to true when the incident involved a high speed collision, which usually means severe injuries.

If a victim is taken to a hospital, this is stored in the system. The *Hospital* class stores the hospital name and location, and its phone number.

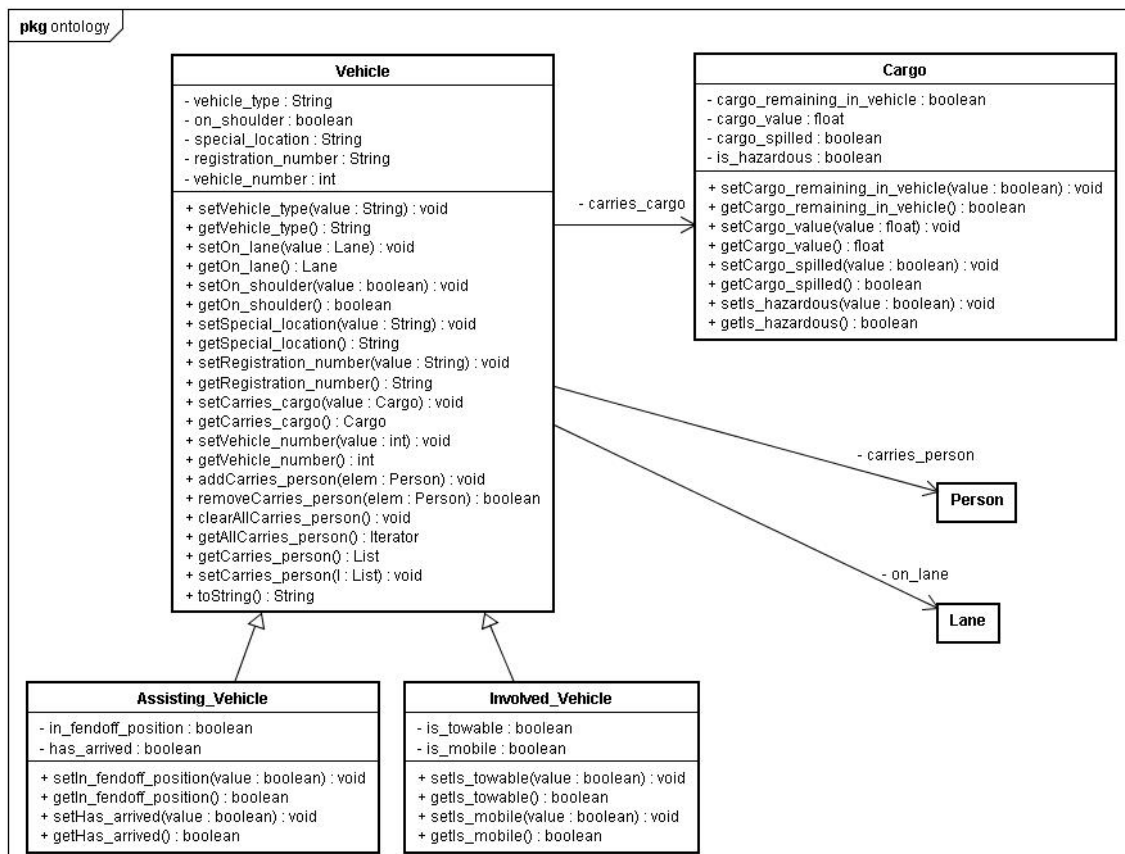


Figure 29 - The vehicle class and its relations.

Vehicles

The *Vehicle* class represents the vehicles involved in an incident. As can be seen in Figure 29, they are identified in the system by a unique *vehicle_number*, they are also identified by the registration number on their licence plate in *registration_number*. The *vehicle_type* field holds the type of vehicle, such as car or lorry. If the vehicle is not at one of the driving lanes, its location is further specified by two fields, *on_shoulder* determines if the vehicle is on the road shoulder and *special_location* is used to enter

unusual or difficult locations, such as at the other side of the guard rail, in a ditch or at the bottom of a slope.

The vehicles can be further divided into vehicles that were involved in the actual incident and assisting vehicles of the emergency services.

Assisting vehicles are represented by the *Assisting_Vehicle* class. They have two extra fields, *in_fendoff_position* determines whether a vehicle is in the fend off position, securing the incident site from approaching traffic [2], *has_arrived* determines whether an assisting vehicles has arrived at the incident scene.

Vehicles involved in the actual incident are represented by the *Involved_Vehicle* class. The field *is_mobile* determines whether a vehicle can still move on its own, *is_towable* determines whether a vehicle can simply be towed, or requires specialised equipment to be removed from the site.

Since vehicles carry persons, they have a list of references to the *Person* objects representing the persons in the vehicle.

If the vehicle is located at one of the driving lanes, a reference is kept to the *Lane* object representing that lane.

If the vehicle was holding cargo, a reference is kept to a *Cargo* object representing it. If there is still cargo remaining in the vehicle, this is indicated by the *cargo_remaining_in_vehicle* field. If the cargo is extremely valuable (e.g. medical equipment), this may give saving the cargo a higher priority, therefore its value is stored in the *cargo_value* field. If the cargo is spilled a result of the incident, this is indicated by the *cargo_spilled* field. If the cargo involves hazardous materials, the *is_hazardous* variable is set to true.

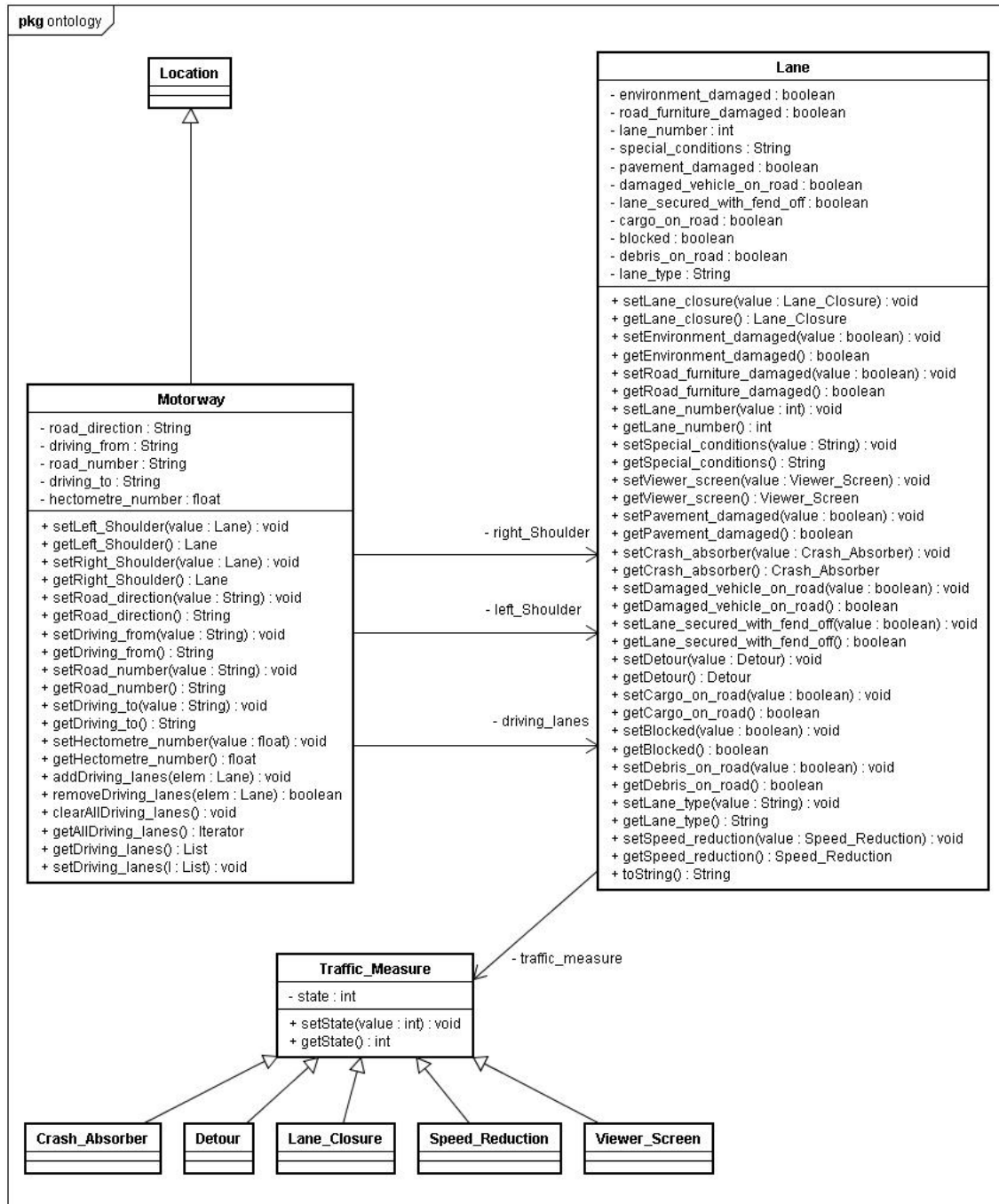


Figure 30 - The location class and its relations.

Locations

As Figure 30 shows, the incident location is represented by the *Location* class. As mentioned in Section 2.2, the Dutch incident management procedures have only been implemented on the motorways. For this reason, although incidents can happen at different types of location, the focus of this thesis, and therefore the IMICS system, is on the Dutch motorways. Therefore the only subclass of *Location* currently available is the *Motorway* class. The ontology could easily be extended to include other locations.

The *Motorway* class represents a real motorway and therefore has fields that represent the road properties. The *road_direction* field indicates what side of the road, so in what direction, the incident is located. This is indicated by either an L or an R (for left and right) as can be found on the hectometre signs at the side of the road. The *driving_from* and *driving_to* fields describe the point of departure and the destination of the road. They are available to check that the direction indication is correct. The *road_number* field holds the motorway number that identifies it. The exact location on the motorway is indicated by the *hectometre_number* field, which holds the number of the nearest hectometre sign. A motorway also holds references to the *Lane* objects that represent the actual lanes of the motorway. If a left or right shoulder is present, the motorway also holds another reference to a *Lane* object for them, since shoulders are modelled as a special type of lane.

The *Lane* object represents the lanes and shoulders of the actual motorway. The *lane_type* field indicates whether the lane is an actual driving lane or a hard shoulder. The *environment_damaged*, *pavement_damaged* and *road_furniture_damaged* fields indicate if there is any damage to the environment, the pavement, or the road furniture respectively. Just as the motorways themselves, lanes are identified by a number, stored in the *lane_number* field. The fields *damaged_vehicle_on_road*, *lane_secured_with_fend_off*, *cargo_on_road*, *debris_on_road* and *blocked* indicate whether a lane has a damaged vehicle on it, if it is secured by a vehicle in the fend off position, if there are debris on the road and if it is blocked. Any specifics that cannot be entered in the other fields can be stored in the *special_conditions* field. A lane can also have references to special traffic measures if they are in place.

The *Traffic_Measure* class represents the traffic measures mentioned above. Traffic measures can be not needed, requested, in place and removed after its use is no longer required. These properties are stored in the *state* field, where 0 corresponds with 'not needed', 1 with 'requested', 2 with 'in place' and 3 with 'removed'.

The different kinds of traffic measure are represented by the subclasses of the *Traffic_Measure* class. The different measures are: crash absorbers to stop traffic safely if necessary, detours to lead traffic around the incident site, lane closures to prevent uninvolved traffic from reaching the incident site, speed reductions to reduce the difference in speed between traffic and the vehicles involved in the incident and viewer screens that prevent other traffic from being distracted (reducing the risk of secondary accidents and traffic jams).

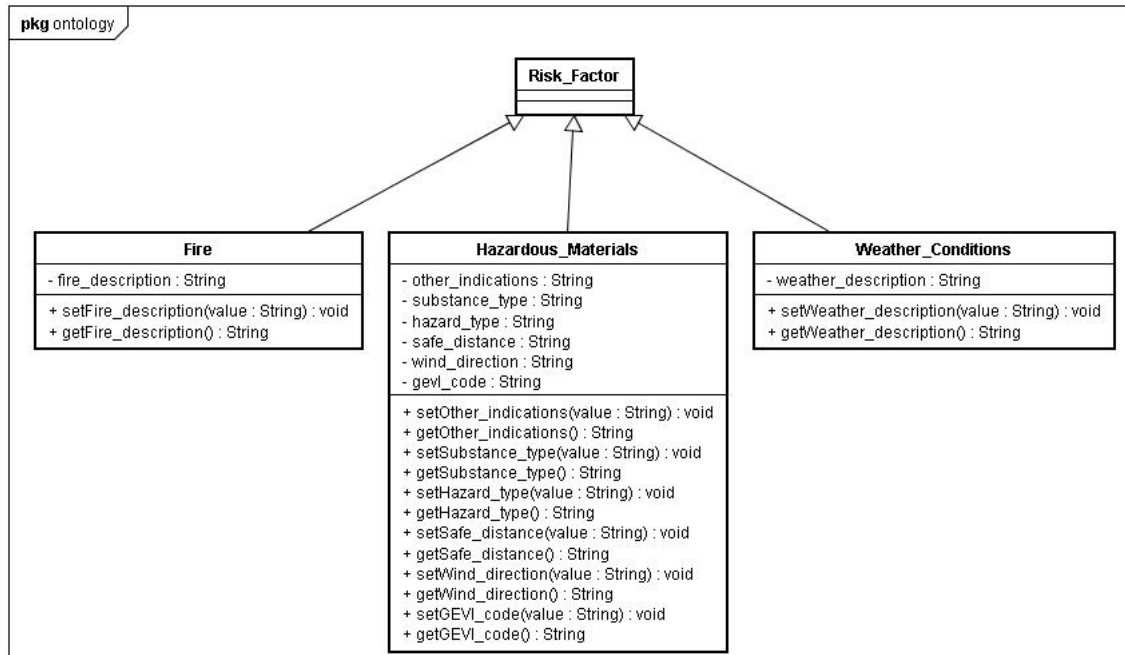


Figure 31- The risk factor class and its subclasses.

Risk factors

The *Risk_Factor* class represents three special situations that pose an increased risk to the involved persons and may require a different approach. Figure 31 shows the way these risk factors are represented in the ontology.

If a vehicle is on fire, this is indicated by a *Fire* object that holds a description of the fire.

If there is a risk that hazardous materials are released, for example when a lorry carrying chemicals is involved, this is represented by the *Hazardous_Materials* class. Since there are many types of hazardous substances, with varying risks associated with them, as much details as possible are stored in the system. First of all, the substance type and hazard type are stored in their respective fields. The *gevi_code* field holds the special GEVI code that identifies the involved substance and its dangers. The *safe_distance* and *wind_direction* fields are used by the emergency services to determine the safety zone that must be closed to make sure no one is exposed to the hazardous substance. Finally, the *other_indications* field can be used to describe other clues, such as a suspicious odour or oddly coloured smoke.

The *Weather_Conditions* class represents weather conditions that could pose a risk to the involved personnel. The *weather_description* field holds a description of the conditions (e.g. heavy showers or snow).

6.3.4 The IMICS log

Both the server and client maintain a log of the messages they received. The log file is created when an agent receives his first message. The file name is composed of the agent name and date. Each time a message is received, it is written to a log file in a readable format as is shown in Figure 32.

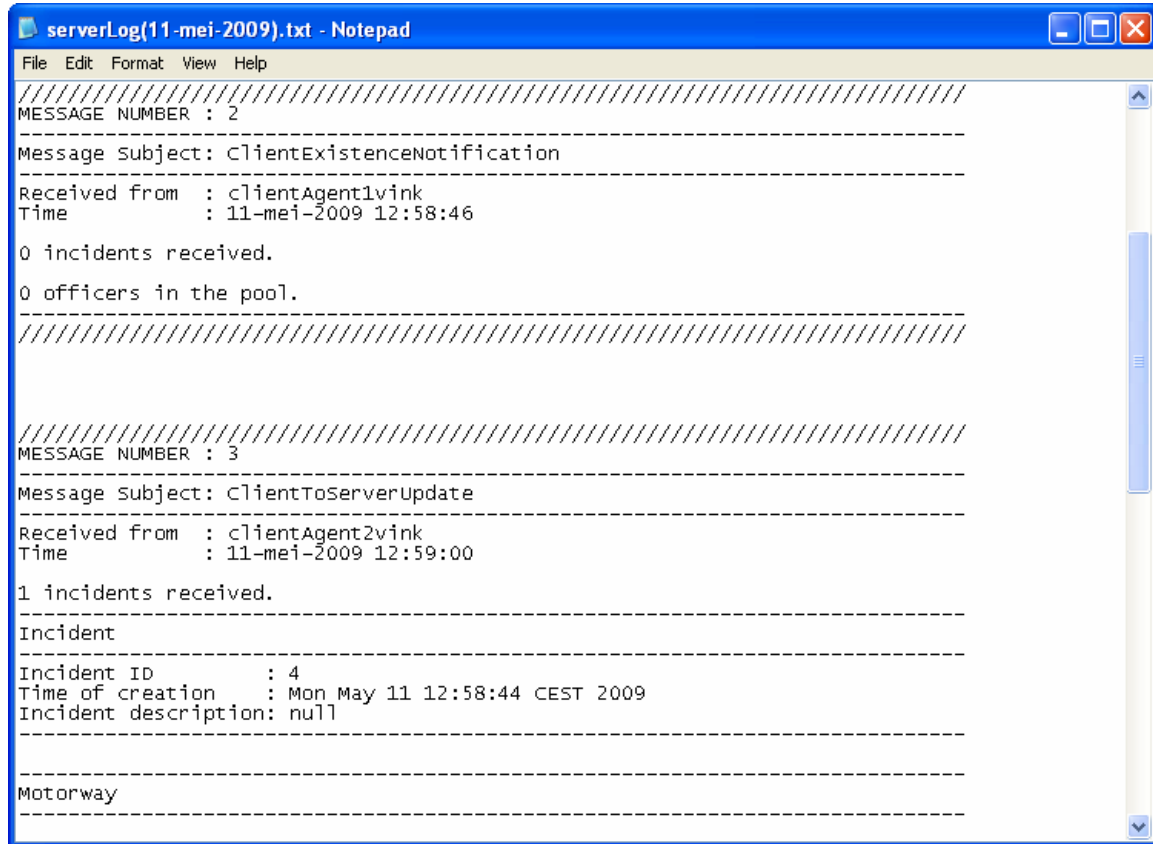


Figure 32 - A part of the log created in one of the system tests.

This log is mainly used for debugging and evaluation. In the testing phase however, it also serves to verify the results (i.e. does the system work as is expected).

For each message a few fields are stored as summarized in Table 28.

Table 28 - The fields stored in the log for each message.

1. A counter to indicate the amount of messages received.
2. The subject of the message.
3. The sender of the message.
4. The time the message was received.
5. The incident objects contained in the message.
6. The officer objects from the officer pool contained in the message.

The most important goal of the log, evaluation, could be improved by providing a specialised interface. It could be augmented with for example search methods, statistics

and the change of information during the incident as mentioned in Section 9.2. This has not been implemented for the proof of concept though.

6.4 Features to be implemented

One of the project goals, as stated in Section 1.2 was to implement a first prototype as proof of concept. Next, refinements have to be made. There are a few general areas in which improvements will have to be made before the system is ready for practical use.

User interface

The user interface of the client system allows for testing of the system in a controlled environment. There are some features that have not been implemented that would greatly reduce the risk of errors and improve system reliability though.

Type checks

At the moment, no type checks are performed for the data provided by the user. This means that if for example when a user provides a string where an integer value is expected (for example a person's age), an error is encountered, and the data is not correctly saved. These checks could be performed either when a field loses focus after a user has updated a field, or when a form is saved. If incorrect input is found, this could be indicated by a pop-up message or a change of colour, prompting the user to change his input before continuing.

Indicator for critical actions

In the current implementation it is possible that a critical action is skipped by a user. For example a required data field has not been filled out, or the user closes a screen without saving the updated data. To prevent such mistakes, a pop-up message could ask the user to save the updated data or to fill out a field before saving.

Indicate updated data

When a user updates and saves data, it is automatically sent to all other users. Currently, the fields are updated with no notification to the other users. Once more, a pop-up message could be shown combined with a warning sound, or in case of low priority data, a simple change in colour could be displayed so the user is not distracted too much.

Synchronisation

The message queue of the JADE agents provides basic synchronisation. Messages are handled one after another on a first in first out (fifo) basis, so the newest update is always shown. JADE can be setup to use persistent messaging. This means that when JADE is unable to deliver a message to an agent, it will store the message and try again. This is especially useful when connections are lost regularly. Persistent messaging is turned on in the IMICS system.

At the moment however, no advanced synchronisation takes place between the server and clients. When 2 updates are received at the server, the last update simply overwrites all data from the first. This could lead to data loss in the system when two updates are

received shortly after each other. For a small scale test in a controlled environment, this is not a real problem. In small scale incidents in practice, few people will be using the system simultaneously, so this would not be a large issue either. For scalability however, this is very critical. One can imagine that in a large scale incident, not having critical data available because it got lost during simultaneous updates would not only lead to much frustration, but could put the lives of victims at risk.

Currently, an update of incident information by an IMICS client simply contains a complete incident object. The complete incident data is overwritten with the new information. This could be improved by comparing and updating each information item separately. By comparing the time an item was updated and using the latest update, most data loss could be prevented. If two users simultaneously update the same data field, the problem still occurs though. More advanced synchronisation techniques could take care of this. Making one users responsible for a specific part of the system also reduces the risk of simultaneously updating the same data.

Multiple instances of one incident

In current incident management practices, it is possible that an incident is reported many times. Sometimes it is unclear that multiple reports describe the same incident. In IMICS, although a new incident would usually be entered at the shared control room, it is possible that a patrolling police officer discovers an incident and wishes to enter in into the system. If the control room has opened a new incident but has not yet saved the new incident to the system, it is not yet visible to the other users, which could lead to two instances of the incident in the system. Although the IMICS system has an overview of all incidents in the system and provides a description containing the time of occurrence and an exact location, it is still possible a user overlooks the right incident instance and creates a new incident in the system. To prevent this kind of ambiguity, if a user still tries to create an incident very close to an existing incident, the system could ask the user if the existing incident is the same as the user wishes to report. The system could also provide a procedure to merge two incident instances. These features were not implemented for the proof of concept though.

Security

As noted in Section 4.3.2, for the scope of this project, security is not very important. For practical use however, since the system could contain sensitive information (especially the organization overview), security is very important. Although JADE has built in options to encrypt sent data and IMICS requires users to provide authentication credentials, these are only a few basic measures. Before the system is put to use, the system should be augmented with strong security measures and tested by security experts.

Server robustness

The centralized server is a critical component for the IMICS system. Therefore, the IMICS design includes a backup server to improve the robustness of the system. JADE can be setup to automatically create a duplicate main container to serve as a backup. This requires only one parameter to be added in the code to start up the JADE platform (and a

second server computer), and should always be in place in practical use. As mentioned before, the server system should be equipped with a user interface to allow system administrators more control over the system.

Users

In the current implementation, all users are treated equally. All users are currently allowed to access all incidents. Although the current system does keep track of the officer pool and the users assigned to an incident, it does not yet check if a user is assigned and allows all users to see and join all incidents. In practice, users should only be allowed access to the incidents they are assigned to. The users at the control rooms should always be able to access all incidents in their region, since they are responsible for assigning other users to an incident. Also, the users at the control room have larger screens at their disposition, allowing them to view multiple incidents at a time. They also have the task of closing incidents, adding new users to the system and checking the logs when something went wrong. These functions have not yet been implemented.

Priorities

Different tasks have different priorities depending on the situation. The same holds for the information in the system and the update messages sent by the users. The system could be extended to allow priorities to be given to certain information. All objects and data fields from the ontology could be given a priority property by giving the `involvedObject` and `Incident` classes a property priority. The user interface could be extended with checkboxes indicating high priority data or messages. The IMICS system could also be extended with relevance inference techniques that automatically determine a message's priority. In Chapter 9, Future work, more suggestions for future extensions of the IMICS system are presented.

Responsibilities

Although the task part of the system is responsible for keeping track of users' responsibilities, this should be coupled to the data in the general incident overview. Users with certain tasks are responsible for entering specific information into the system. It should be visible who is responsible for a certain piece of information and who has provided the last update of it. Since the task part has not been implemented, this also has not been implemented in the incident overview.

Chapter 7

Evaluation and testing

The different software components have been tested after they were completed during the implementation phase. After all components of the IMICS prototype were implemented, the system was tested using a fixed test setup. In this setup, three scenarios were tried using the prototype. Before this test and its results are discussed, in Section 7.1 the system is compared to the theory from Chapter 3. Section 7.2.1 describes the test setup. The used scenarios are described in Section 7.2.2. Finally, in Section 7.3 the results of the tests are presented.

7.1 Comparison with theory

The theory in Chapter 3 is about incident management in general, which is much broader than traffic incident management. Specifically the scale of traffic incidents is usually much smaller than general incidents or disasters. This means that some suggestions made in the theory may seem a bit overdone. However the basic principles are meant to improve communications and situational awareness, which basically are the goals of this thesis.

Design premises

Turoff et al. first present a number of design premises [10]. These are now compared with the implemented system.

1. Prevent information overload (show relevant information only)

The system design is clearly separated in parts with different functionality. In each part, the data relevant to a certain task is shown. In this way, theoretically, the user should not be overwhelmed by the data in the system.

2. Improve situational awareness (provide and review up-to-date information and context)

As all information in the system is provided in a clear structure, relevant information should be easy to find. Since all data in the system is directly made available to all users they should be aware of new facts quickly.

3. Support configuration (adaptable priorities and filtering)

JADE provides a way to automatically publish agents as services, allowing a service oriented architecture to be created including the IMICS functionality. In this way, users or administrators can select the services they would like to use. At the moment however, this is not used. The user interface itself can currently not be modified by the user. On the other hand, the relatively limited scope of traffic incidents, combined with the fact that information is grouped into a well defined structure that follows incident management tasks and procedures, should enable users to work efficiently using IMICS.

4. Support role and task transfer (monitor and manage tasks and availability of personnel)

Although it has not been implemented for this proof of concept, the IMICS design incorporates task management. The task part of the design allows tasks to be transferred within and across organization boundaries.

Roles that should be supported

After the design premises, Turoff et al. give a list of roles that should be supported in order to improve incident management [10].

1. Request resources (people and equipment)

In the implemented part of the system, it is possible to assign persons to the incident, so they can be given access to the incident. This could also be used to dispatch personnel to the incident scene. In the current situation the emergency services themselves are responsible for dispatching units. In practice, the best solution would probably to couple IMICS to the existing dispatch system. In that situation, IMICS would forward requests to this system. So the IMICS design does support this role, but only indirectly at the moment.

2. Allocate, delay or deny resources

Allocation of resources is currently handled by the emergency services own dispatch systems. Just as with requesting resources, IMICS could be coupled to these systems, but does not facilitate this feature directly.

3. Report and update situation

The main functionality of IMICS is presenting incident information and keeping it up-to-date. This feature is fully supported by IMICS.

4. Analyze situation

The clearly structured information presented in IMICS enables analysis of the incident situation. This analysis could even be partially automated (see Chapter 9, Future work).

5. Edit, organize, and summarize information

All users are able to edit information in the system. This information is organised in a standard structure that follows the tasks and procedures of traffic incident management. It is organised into different aspects of the situation and the different organizations have their own subparts in which only information relevant to their tasks is found. In future versions, users could be allowed to adapt the layout of the information to their own

insight, but the current proof of concept does not support this. The information is limited to key concepts, so it does not need to be summarized in the system.

6. Maintain resources (logistics)

Maintaining resources is mainly done by the different emergency services themselves. IMICS does provide an overview of the people assigned to an incident and the people available in the officer pool. An overview of the traffic measures taken and their status is also available within the system.

7. Acquire more or new resources

Except for persons and traffic measures, extra resources cannot be acquired directly using IMICS. When extra resources are required, this is usually due to the unusual nature of an incident. In these cases, an expert must often be consulted first. At the moment this must be done by phone. This will generally be the responsibility of the director or the control room.

8. Oversight review, consult, advise

Everyone with access to the system can get a complete overview of the information of the incidents he is assigned to. In this way, experts could read the information from a different location and consult with or advise other users via the (not yet implemented) communication part of IMCIS or via telephone. The communication part holds the contact information of important persons.

9. Alert all with a need to know

The shared control room is responsible for assigning the right persons. Once these persons are assigned, they automatically receive all updates through IMICS. Although this is not yet implemented, people could be notified of an important update as mentioned in Section 6.4 (hfstnr checken).

10. Assign roles and responsibilities when needed

The tasks part of the IMICS design completely supports this feature.

11. Coordinate among different resource areas

It is possible to connect to a server of a different safety region. In this way, a user can help with an incident outside of his own safety region just as he normally would. The local shared control room has the responsibility to assign a user from a different region if required. Once this has been done, the user is treated as a local user and the incident is coordinated as usual. Other resources, such as equipment are managed by the people and IT systems in use at the emergency services' stations. IMICS could be coupled to such existing systems in future versions.

12. Priority and strategy setting (e.g., command and control)

Although the priorities and strategy are laid out in the standard procedures, the COPI team, and the director in particular, is responsible for any deviation from these procedures. The task part allows changes to the approach to be put in effect directly.

Design principles

Following the design premises and the required roles, a number of design principles was given. These were compared with the system.

1. System Directory

The current design provides a hierarchical structure for all data in the form of the ontology. At the moment, no text search is provided. Since all relevant data for the primary stakeholders is provided in a clear structure, this should not pose a problem in normal situations. In very large scale incidents however, scrolling through the list of involved vehicles, for example, might become a bottleneck. For this reason, a simple text search, as can be found in for example most internet browsers, should be incorporated in a fully deployed system.

2. Information Source and Timeliness

This is provided for by the designed ontology (as discussed in Section 6.3.3). At the moment however, the time of occurrence and original source of information is stored in the incidentMessage object for the entire message. This means that only the time and sender of the latest information is stored locally. All messages are stored in the log though. More direct access to the time and provider of each separate information item would require the server to compare each updated item with the old item separately. This is part of the advanced synchronisation that has not been implemented for the proof of concept, as mentioned in Section 6.4.

3. Open Multi-Directional Communication

Although traffic incident management is not as chaotic as disaster management, open communication is still critical for successful incident management. This is supported in many ways by the system design. First of all, it is provided by the incident information that is shared with all assigned users. Secondly, all users can communicate freely using the communication part of the system. The PDA the design is based on can also be used to call a person directly and in practice, much of the communication is direct and personal at the incident site.

4. Content as Address

In the current design, most information about an incident is of direct interest to all users involved. An incident in the system and its assigned users can be seen as a group of common interest. In this way, this principle is provided by IMICS.

5. Up-to-Date Information and Data

This principle is fully supported by the system as data is automatically updated in the system as discussed in Chapter 6. Notifications and priorities have not been implemented yet, but are required as discussed in Section 6.4.

6. Link Relevant Information and Data

The design of the ontology and the incidentMessage object link all data as a single unit of information. As an update always contains this complete set of information, IMICS complies with this design principle.

7. Authority, Responsibility and Accountability

As Section 5.4 discusses, the director is responsible for keeping an overview of the situation. This role can be taken by either the control room or a senior officer at the incident scene. Although the task part of the system is not implemented, its design, combined with the director role, follows this design principle.

8. Psychological and sociological factors

People from the different emergency services in a safety region receive training together and cooperate often in practical work, they are aware of the others' tasks and way of working. IMICS should become part of their training sessions to ensure users are familiar with the functionality and the kind of interaction the system provides. Social interaction is supported by the different means of communication the system provides. Information overload is reduced by the well structured information in the ontology. Personal priorities and adaptations to the user interface are not yet implemented as discussed in Section 6.4.

Finally Turoff et al. mention that communication contains a lot of ambiguity [10]. The designed ontology, in combination with the incident management procedures clarify the objects and relationships in a traffic incident setting and should reduce ambiguity.

7.2 System test

Now that it has been determined that in combination with the existing systems the basic requirements have been met by our design, it is important to perform a practical test to see if the system does indeed do what it is designed for.

7.2.1 Test setup

The test is run with a server computer and one or more client computers. On the server computer, the IMICS server is started. Since the amount of available computers is sometimes limited, after the server has started correctly, an IMICS client is started on this computer as well and connects to the server at the local machine. On the other computers, only an IMICS client is started. It is given either the IP address or the network name of the server computer, and optionally a port number to be used, and connects to it.

When the server and clients have been set up, the users can start using the system to work through the predefined scenarios as in a real incident situation. The users of the different computers serve as the different field officers involved in the incident. Not all stakeholders have access to the IMICS system, and those that do, do not always provide information for the IMICS system. In the user tests, only the stakeholders that provide information in IMICS are represented. For the third scenario, users were only given the general storyline and the data they had to enter into the system to make the test more realistic.

7.2.2 Scenarios

The system is tested with 3 scenario's, with increasing complexity. The first scenario is a simple incident that only involves material damage and one field officer from the department of Public Works. The second scenario involves a trapped and heavily injured driver and a blocked lane and requires the cooperation of multiple organizations. The third and most complex scenario involves a truck carrying hazardous materials. This scenario requires a well coordinated cooperative effort and should benefit the most from the IMICS system.

The scenarios are described step by step. The different events are identified by the time of occurrence and, if applicable, the organization they involve. The numbers in brackets indicate events that result in new information. As will become clear, not all of these events can be stored in the currently implemented part of IMICS. The bold numbers can be stored in the system, but other numbers are events that should be stored in not yet implemented parts of IMICS (such as the task part or the organization part). The bold text following a bold number in brackets is the information that has to be entered into the system. As mentioned above, during the third test, this information was only given to the stakeholder that has to enter this data.

Scenario 1: Material damage only

A driver is too close to the car in front of him and is unable to stop in time when the driver before him has to use his brakes.

Step by step description:

16:30 A car hits the car in front of him with limited speed, resulting in material damage only. Both drivers are uninjured and are able to move their vehicles to the hard shoulder without assistance.

Department of Public Works

16:35 While the drivers are getting their insurance forms, an inspector from the Department of Public Works patrolling the area notices the incident and stops his car at the hard shoulder in front of the damaged vehicles. He places his vehicle in the fend off position and keeps his flashing lights activated to warn incoming traffic. The inspector checks if the incident is present in the system and creates a new incident in the IMICS system and enters the location **(1). The incident is at the A13, near hectometre sign 26.2 R on the route from Den Haag to Rotterdam. There are 3 lanes and a right shoulder present.**

When he finds the incident involves material damage only and both vehicles are able to move on their own he suggest to drive to the nearest rest area.

16:40 When the drivers arrive at a fuel station, the inspector asks if they require assistance filling out the insurance forms. The drivers agree they will be able to complete the forms themselves.

Department of Public Works

16:45 The inspector enters the vehicles' registration numbers and the drivers names into the IMICS system (2). **The vehicles are both normal cars and are both mobile. The first car has registration number xx-xx-xx and the second has registration number yy-yy-yy. The driver of the first car, with license number xx-xx-xx is A. Anders, is 21 years old and his phone number is 06-11. The driver of the second car is the 22 year old B. Berends, with phone number 06-22.**

The inspector verifies the data and closes the incident. The inspector leaves the fuel station.

Scenario 2: Two cars with a blocked lane and a trapped, injured driver

At 8 o'clock in the morning, during heavy traffic, a car crashes into another car at high speed at the right driving lane. Only these two cars are involved in the incident. Although the front driver is only lightly injured, the back driver is heavily injured and must be taken to hospital as soon as possible. He is trapped in his car however and is unable to get out on his own. The vehicles completely block the right lane and cannot move to the hard shoulder themselves, leaving only two lanes available for traffic.

Step by step description:

Control Room:

8:01 The emergency response centre receives the first report of the incident. A driver saw the incident and called the emergency number 112 immediately. The driver is connected to the local control room. During this call, as much information as possible is obtained about among others the location and the amount of people, vehicles and injuries involved through the standard procedures (1).

Control Room:

8:03 Multiple notifications of the incident reach the control room. They are processed and new information is stored (2). **The incident has taken place at the A4, near hectometre sign 76.5 R on the route from Den Haag to Delft. There are 3 lanes and a right shoulder at the location. Lane 3 is blocked since there is a vehicle on the lane. There are two vehicles involved. The first car with registration number xx-xx-xx is no longer mobile and towable. It is at lane 3 and only contains the driver. The second car, with registration number yy-yy-yy is also no longer mobile and towable, is also at lane 3 and also holds only one person.**

So the incident involves 2 persons, the driver from car xx-xx-xx, who is injured in the high speed impact, and the driver from car yy-yy-yy who is also injured by the high speed impact.

A police surveillance unit that is close to the incident location is dispatched to the incident site (3).

Control Room:

8:04 Since the incident occurred at high speeds, an ambulance is dispatched since injuries are suspected (4)

Control Room:

8:05 The driver of the front car also calls 112 and reports that he is barely injured, but the other driver is trapped in his car (5). **The driver of car xx-xx-xx is barely injured, the driver of car yy-yy-yy is trapped and badly injured.**

Control Room:

8:07 Because one of the drivers is trapped in his vehicle, an emergency assistance vehicle from the fire brigade is sent to the scene (6).

Control Room:

8:07 Another control room employee contacts the regional traffic control centre to verify the exact location, close the middle and right lane and reduce the allowed speed at the left lane (7). **Request lane closure for lane 2 and 3, speed reduction for lane 1.**

Regional Traffic Control Centre:

8:08 The traffic control centre verifies that the location is correctly entered into the system and closes lanes 2 and 3 and reduces the speed limit at lane 1 (8). **Lane closure in place for lane 2 and 3, speed reduction in place for lane 1.**

8:09 An inspector of the Department of Public Works is called and dispatched to the site (9).

8:10 The CMI is called to have the vehicles towed away from the incident location (10).

CMI:

8:12 The CMI calls the recovery service company (11).

Recovery service company:

8:14 The recovery service company dispatches two vehicles to the incident. Unfortunately there is only one recovery vehicle available because of another incident and a second vehicle has to come from a location further away.

Police:

8:14 The dispatched surveillance unit was nearby and is the first to arrive at the incident location. The police driver secures the location by placing his vehicle in the fend off position while the other police officer provides first aid to the victims. It turns out the front driver requires no medical attention, but the trapped driver must be taken to hospital. Meanwhile the police driver joins to assist and provides supplementing information about the trapped driver to the control room (12). **Lane 3 secured with fend off. The driver from car xx-xx-xx is 24 year old A.**

Anders, the driver from car yy-yy-yy is the 32 year old B. Berends and requires a hospital.

Department of Public Works:

8:15 The inspector from the Department of Public Works arrives at the incident location. It appears there is no significant damage to the road and road furniture, but there are glass and oil on the road (13). **No damage to road, some debris and oil on road**

He has to wait for the police to gather evidence at the scene before he can start cleaning the road, so in the meantime he assists providing first aid.

Police:

8:17 One of the police officers starts gathering evidence.

Ambulance:

8:18 The ambulance arrives at the incident scene. They treat the trapped driver as well as possible and also check the front driver. They confirm he does not need medical attention and can be brought home (14). **Driver car xx-xx-xx treated.**

Fire brigade:

8:20 The fire brigade arrives at the scene with special tools to free the trapped driver. They consult with the ambulance personnel how to free him.

Police:

8:21 The police has finished gathering evidence, so the cars can be removed from the scene as soon as possible.

Department of Public Works:

8:21 Although the police has finished gathering evidence, freeing the trapped driver has first priority, so the inspector decides to wait until this is finished before cleaning the road.

Fire brigade:

8:21 The fire brigade starts cutting free the trapped driver. They need about 10 minutes to free the driver while the ambulance personnel keeps an eye on his status.

Recovery service company:

8:26 The recovery worker arrives at the location. They quickly decide to tow the front car from the right driving lane to the hard shoulder. Because the driver of the other car is still trapped, this car cannot be towed away yet. Because clearing the right driving lane has higher priority than the hard shoulder they consult with the police whether to start towing the front car or wait to clear the driving lane as soon as possible (15). Since a second recovery vehicle has almost reached the

incident location, as indicated by the tracking and tracing software the recovery service company uses, they decide to start towing away the front car.

Ambulance:

8:31 The driver has been freed from his car and is quickly taken to hospital by the ambulance. The ambulance informs the control room to what hospital the driver is taken while driving there.

Department of Public Works:

The inspector has to wait until the second car has been towed away until the road can be completely cleaned

Fire brigade:

8:32 The fire brigade starts clearing materials.

Recovery service company:

8:35 The second recovery unit arrives and immediately starts removing the remaining car wreck.

Department of Public Works:

8:38 Now that all wreckage has been removed, the inspector can start clearing the last debris and oil from the road.

Fire brigade:

8:38 The fire brigade assists in clearing the road.

Department of Public Works:

8:48 The road has been completely cleaned and the inspector from the Department of Public Works leaves the incident scene **(16). Road cleared.**

Fire brigade:

The fire brigade has completed its tasks and leaves the incident location **(17).**

Police:

8:49 The police is the last to leave the location. They notify the traffic control centre that the situation has returned to normal and the speed reduction and lane closures can be removed and return to their station **(18). All traffic measures being removed.**

Traffic control centre:

8:50 The traffic control centre removes all traffic measures **(19). All traffic measures not used.**

Scenario 3:

At 4 a.m. a truck loaded with hazardous materials has toppled over, leaving the truck on its side and its driver badly injured. The truck is blocking the right driving lane and the adjacent hard shoulder. The truck is damaged badly and part of the dangerous load is leaking onto the road. Despite the late hour, the incident is reported fairly soon.

Step by step description:

Emergency response centre:

4:03 A passing driver sees the toppled truck and calls the emergency number 112 (1). He is quickly connected to the local shared control room.

Control room:

4:04 The control room opens a new incident in IMICS. Following standard procedures, the control room asks about the location and vehicles involved in the incident. Since a truck is involved, they ask if there is any suspicion that hazardous materials are involved. The reporting driver informs them he has not recognized any names or orange warning signs, but he noticed a penetrating odour (2).

The incident has taken place at the A2 motorway, on the route from Amsterdam to Utrecht. The nearest hectometre sign indicated 27.3 R. At the location, the motorway has three lanes and a right shoulder. The right shoulder and the right driving lane (lane 3) are blocked by the toppled truck. (Cargo at lane 3, right shoulder).

The involved vehicle is a truck with registration number zz-zz-zz and since it is on its side, is not mobile or towable. Because of the penetrating odour, it is suspected that its cargo is at least partially spilled. It is assumed for now that some of the cargo is still left in the truck and that it could be a hazardous substance. The reporting driver thinks the truck driver was still in his vehicle. Considering the situation, they suspect the driver is injured. Because of the potentially dangerous load, a safety distance of 100 meters is kept. The hazard type is currently unknown and the only indication is a penetrating odour.

Control room:

4:07 The control room asks the KNMI for the wind direction, which turns out to be west (3).

Control room:

4:08 The control room contacts the regional traffic control centre and asks the preferred route to the incident with the current wind direction and the hazardous materials that may have been released. As a precaution, they ask the traffic control centre to close the road for the time being (4). **All lanes (except shoulders): lane closure and detour requested.**

Control room:

4:08 In the meantime, the control room decides to send a police surveillance unit to the location (5). Since they suspect the truck driver may be injured, they decide to send an ambulance to the incident location (6). Since hazardous materials are suspected to be involved the fire brigade is required and will be in charge of the incident relief (7). The CMV is called to salvage the truck (8).

Regional traffic control centre:

4:10 The traffic control centre checks the location with the camera's at the motorway and confirms the location and the fact that a truck is on its side. They also apply the requested traffic measures (9). **Traffic control centre joins the incident. All lanes closed and detour in place.**

Police:

4:12 The police surveillance unit has left for the incident location (10).

Ambulance:

4:13 An ambulance is dispatched since they expect the truck driver to be injured (11). **The ambulance service joins the incident.**

Regional traffic control centre:

4:13 Because of the risk of hazardous materials, traffic is rerouted using the dynamic route information panels (DRIPs) (12). An inspector of the Department of Public Works is dispatched to the incident scene (13). He will receive the phone number of the Lorry Salvage Consultant as soon as this is available.

Fire brigade:

4:14 An assistance vehicle is sent to the incident location (14). **The fire brigade joins the incident.**

Since hazardous cargo is potentially involved, the regional officer hazardous substances (ROGS) is called (15). A consultant hazardous substances (AGS) is also called (16). They are in charge of exploring the incident scene and releasing it to the other organizations when the danger has been mitigated.

Department of Public Works:

4:15 The inspector starts driving towards the incident location (17). **The inspector from the Department of Public Works joins the incident.**

IM lorry recovery dispatch centre (CMV):

4:15 The CMV notifies a salvage company (18) and a lorry salvage consultant (STI) (19). They also call the Department of Public Works to get the contact information of the involved inspector (20) and to give him the name and phone number of the lorry salvage consultant (21). This information is also given to the control room together with the information of the recovery service company that is dispatched to the incident (22).

Recovery service company:

4:17 The recovery service company dispatches heavy equipment to salvage the truck (23).

Lorry salvage consultant:

4:20 The lorry salvage consultant starts driving towards the incident (24) and gives an estimate of his time of arrival to the CMV (25) and the Department of Public Works (26).

Fire department:

4:22 The fire department is the first to arrive at the incident location (27). Because of the risk of hazardous materials, they keep a safety distance of 100 metres from the truck. Since they are the first to arrive, they put their vehicle in the fend off position. They immediately supply the control room with additional information about the exact location (the hard shoulder and the right driving lane), the involved vehicles (the truck and its registration number), and the codes of the orange warning signs on the truck indicating the type of substance the truck carries (28).

Lane 3 and the right shoulder are now secured with the fend off position. The GEVI code at the warning signs is 33-1203

They also call the ROGS and AGS to inform them of the involved substance (29).

Police:

4:23 The surveillance unit arrives at the scene (30). They take over the fend off position from the fire brigade (31). They consult with the fire brigade at the scene and keep a safe distance of 100 metres from the toppled truck until the situation is deemed safe (32).

Incident management lorry recovery dispatch centre:

4:24 The IM lorry recovery dispatch centre contacts the owner of the truck (33). The owner has a truck available and dispatches it to the incident site to remove the remaining cargo.

Fire department:

4:25 After consulting with the AGS the cargo turns out to be a flammable, but otherwise fairly harmless substance (34).

Substance type: Gasoline, Hazard type: Flammable, Safe distance: 100m

They decide to cover the spilled substance with foam (35). For security it is decided that the other organizations keep their distance for now (36). Wearing protective gear the firemen approach the truck and cover the substance. At the same time, they find that the truck driver is heavily injured, but because of the risk of fire he cannot be helped yet (37). They also discover that the leak is only small and can be closed relatively easy (38).

Department of Public Works:

4:25 The inspector from the Department of Public Works arrives at the scene (39) and for now remains at a safe distance. After consulting with the fire brigade (40) he contacts a specialized cleaning company to clear the spilled substance (41).

Ambulance:

4:28 The ambulance arrives at the scene, but has to remain at a safe distance for now (42).

Fire department:

4:35 After the leak has been closed and the spilled substance has been adequately covered with foam (43), the ambulance personnel is allowed to approach the truck (44). In the meantime the firemen provide first aid to the driver who has regained consciousness but is unable to get out of the truck alone.

Ambulance:

4:35 The medical personnel rush toward the victim, whose injuries turn out to be significant, but not life threatening (45). He is stabilized and prepared to be moved to hospital as quickly as possible. In the meantime they ask the victim his name and age the ambulance service enters this information into IMICS and closes the system (46). **The victim is a 52 year old man, called H. Groen. He is taken to the UMC hospital in Utrecht. The ambulance service closes the system.**

Police:

4:35 Although part of the evidence is covered in foam, the police start identifying the remaining evidence (47).

Recovery service company:

4:37 The recovery unit arrives at the scene (48). Before the truck can be towed however, the remaining cargo will have to be removed and the spilled cargo will have to be cleaned.

Lorry salvage consultant:

4:45 The lorry salvage consultant arrives at the scene (49).

Specialized cleaning company:

5:10 The cleaning company arrives at the scene (50).

Transport company:

5:20 The new truck sent by the trucks owner arrives at the scene (51). In cooperation with the fire brigade, the lorry salvage consultant and the recovery worker, they start pumping the cargo into the new truck. (52)

5:45 The remaining cargo has been completely moved to the new truck and the truck leaves the incident scene (53).

Fire brigade:

5:45 The fire brigade enters the fact that no cargo is remaining into the system (54).

Recovery service company:

5:45 After consulting with the lorry salvage consultant, the inspector from the Department of Public Works and the fire brigade, it is decided the incident site is now safe to start salvaging the toppled truck, however they decide to keep the detour in place so the recovery worker can use the entire motorway (55).

6:30 The truck has been salvaged, and the recovery service company leaves the incident scene (56).

Department of Public Works:

6:30 The inspector from the Department of Public Works updates the system, indicating that the truck is removed from the right lane and shoulder and these lanes are no longer blocked (57).

Fire department:

6:30 The fire department, the Department of Public Works and the cleaning company start cleaning the road after a quick consult (58).

Department of Public Works:

6:30 The inspector determines the pavement at the right lane and at the hard shoulder is damaged. It requires repairs and the inspector contacts a specialized contractor (59).

Cleaning company:

7:05 The road has been cleaned and the cleaning company leaves the incident site (60).

Department of Public Works:

7:05 The inspector of the Department of Public works indicates no debris and cargo remain on the road (61).

Fire department:

7:05 The fire brigade has completed its tasks and leaves the incident site (62). **They close the IMICS system.**

Lorry salvage consultant:

7:05 His task is completed and he leaves the incident scene (63).

Department of Public Works:

7:08 After consulting with the police, the inspector calls the regional traffic control centre. The lane closure and detour for the left and middle lane (lane 1 and 2) are no longer needed, but they a speed reduction is requested (64).

Regional traffic control centre:

7:09 The left and middle lane (lane 1 and 2) are opened for traffic, but with reduced speed due to the repairs that have to be made at the right lane. The detour for all lanes is removed (65).

Contractor:

7:40 The contractor arrives at the scene and starts repairing the road (66). Aside from the DRIPs above the motorway, they place warning signs to inform traffic of the road repairs (67).

Police:

7:45 Since the location has now been properly marked with warning signs, the police leave their fend off position and leave the scene (68).

Contractor:

9:20 The road has been repaired and the contractor leaves the scene (69).

Department of Public Works:

9:20 The inspector finds that all is back to normal and leaves the scene. He informs the traffic control centre that the signs can be returned to normal, opening the right lane and removing the speed reduction. He then closes the IMICS system **(70). All remaining traffic measures are being removed. No one is in the fend off position anymore and there is no more damage to the pavement. The inspector closes the IMICS system.**

Regional traffic control centre:

9:21 The traffic control centre removes all remaining traffic measures and enters this into the system, after which they close the IMICS system for this incident **(71). All traffic measures are back to normal. The traffic control centre closes the IMICS system.**

Control room:

9:22 The incident has been completely resolved and all involved parties have left the incident scene. Therefore, the control room closes the incident **(72). The control room closes the IMICS system for this incident.**

7.3 Test results

In this section the test results of the three scenarios are discussed. Since the consecutive scenarios are increasingly complex, it is expected that the later tests lead to the most interesting results.

7.3.1 Scenario 1

The first scenario concerns a small incident and does not really require any involvement from the emergency services. The inspector from the Department of Public Works only enters basic information about the vehicles and drivers and leaves the scene.

In the test, all information was stored correctly and was accessible by other users should it have been necessary. Figure 33 shows a part of the log containing the incident data. With all entered data correctly written into the log at the server, the system allows information to be stored for future reference.

```

serverLog(Jun 17, 2009) - Notepad
File Edit Format View Help

-----
Involved_vehicle
-----
Vehicle number      : 0
Vehicle type        : car
Registration number  : xx-xx-xx
Is on shoulder       : false
Special location    :
Is mobile           : true
Is towable          : false
At lane number      : 101
Passengers          : 1
Carries person      : null
-----
Involved_vehicle
-----
Vehicle number      : 1
Vehicle type        : car
Registration number  : yy-yy-yy
Is on shoulder       : false
Special location    :
Is mobile           : true
Is towable          : false
At lane number      : 101
Passengers          : 1
Carries person      : null
-----
Person
-----
Person number : 0
Person ID     : 0
Phone number  : 06-11
Name          : A. Anders
Age           : 21
In vehicle    : 0 car xx-xx-xx
was injured   : no
Brought to hospital : no
-----
Person
-----
Person number : 1
Person ID     : 0
Phone number  : 06-22
Name          : B. Berends
Age           : 22
In vehicle    : 1 car yy-yy-yy
was injured   : no
Brought to hospital : no
-----
Risk factors
-----
Assigned persons :
0 persons assigned.

0 officers in the pool.
-----
///////////////////////////////////////////////////////////////////

```

Figure 33 - A log file containing incident data.

7.3.2 Scenario 2

The second scenario is more involved, and requires cooperation of multiple organizations. The stakeholders using IMICS are the shared control room, the police, the ambulance service, the fire brigade, the traffic control centre and the Department of Public Works. Since at the time only three computers were available, this scenario was tested with three persons. Each person had to take the role of two stakeholders on one instance of IMICS. On the first computer both the IMICS server and an IMICS client were activated. The client system was used for the role of the control room and the ambulance service. The second user played the role of the police and the traffic control centre and the final user played the role of the fire brigade and the Department of Public Works as shown in Table 29.

Table 29 - The roles as distributed over the different computers during the second test.

Station	Programs running	Roles taken
PC 1	IMICS client, server	Control room, ambulance service
PC 2	IMICS client	Police, traffic control centre
PC 3	IMICS client	Fire brigade, Dept. of Public Works

In this test, all information reached the other client systems correctly. There was one problem though: at one occasion an update was not displayed correctly by the IMICS client that created the new incident (the shared control room). This client's log showed that the update message was properly received, so the issue was related to updating the interface. Closing and reopening the incident on this system fixed the problem since the data was properly received. In practice however, this is not an option, so the problem had to be resolved. As it turned out, the bug was a result of the way the client automatically opens a new incident after it is created at the server and received by the client. By changing the way a new incident is opened, the problem has been addressed completely and updates are now displayed correctly. Further tests have shown no incorrect behaviour.

7.3.3 Scenario 3

The third and most complicated scenario requires the cooperation of many organizations. Since not all involved parties have been included in the design of IMICS, those not in the design will have to communicate using the old means in practice and have not been included in the test. This scenario was run three times to see how the IMICS system would perform in practice. In these tests, all involved stakeholders were represented on a separate computer running an IMICS client. Since the police do not enter information in IMICS in this scenario, they were not represented by a user during the test. The IMICS server was run on a separate machine that could not be directly accessed by the users, as would be the case in practical use. Table 30 shows the stakeholders included in this user test.

Table 30 - The roles and computers used in the third test.

Station	Programs running	Role taken
PC 1	IMICS server	Automated server
PC 2	IMICS client	Ambulance service
PC 3	IMICS client	Shared control room
PC 4	IMICS client	Fire brigade
PC 5	IMICS client	Dept. of Public Works
PC 6	IMICS client	Traffic control centre

In an early practical test, it turned out there were some minor shortcomings to the scenario. These shortcomings are listed in Table 31.

Table 31 - Shortcomings to the third scenario corrected after the first test.

1. The scenario did not include any information about when stakeholders should join or close the incident. This information was added to the scenario.
2. It was not clear to the test persons that it should be indicated that debris and cargo were on the driving lane. Although this would have been clear to trained emergency services, this was stated more clearly in the scenario for the responsible stakeholder (control room).
3. At the end of the scenario, the Department of Public Works called off the traffic measures, but the scenario did not include the actual removal of the traffic measures by the traffic control centre. This final step was added to the scenario.

The scenario was updated to take care of these shortcomings and two more tests were performed. These tests showed once more that the basic system works. The updates are all received correctly by all users. After the bug encountered in the second scenario was addressed, no unexpected behaviour was found.

Since the test subjects were unfamiliar with the project and the field of incident management, it was expected they also might find shortcomings in the user interface or the used ontology. As the tests showed, a few issues were indeed found. Some of the suggestions made by the test persons were already known, such as the lack of an indication that data has been saved or updated (as described in Section 6.4). The other shortcomings found are shown in Table 32.

Table 32 - Shortcomings to the IMICS prototype found during the test.

1. In the traffic tab, the hard shoulders are displayed above the driving lanes. When a change had to be made to the first and second lane, users occasionally made these changes to the two lanes shown first, which in this case were the right shoulder and the first lane.
2. The system currently uses checkboxes to indicate if certain conditions are true or not. However, if it is not yet known if a condition is true, it is currently also unchecked. This could be misleading.
3. The status indications for the traffic measures were unclear at times. They were:
 - “Not needed” if the traffic measure was not used (the starting position),
 - “Requested” if the traffic measure was requested but not yet applied,
 - “In place” if the measure was applied and
 - “Removed” if the traffic measure was no longer needed but not yet removed.
 The use of the “Not needed” and “Removed” descriptions caused some ambiguity in one of the tests.

The third issue was simply a matter of using less ambiguous names. To improve clarity, they have been changed to:

- “Not used” if the traffic measure was not used (also the starting position),
- “Requested” if the traffic measure was requested but not yet applied,
- “In Place” if the traffic measure was applied and
- “Being removed” if the traffic measure was no longer needed, but not yet removed.

The first and second issues have not been addressed in the implementation. The ambiguity of the different lanes and hard shoulders could be addressed by separating the shoulders from the normal driving lanes or by displaying a graphical interpretation of the different lanes. The checkboxes could be replaced by a three state selection, where the third option indicates the fact that this information item is not yet known.

7.3.4 System performance

During the testing phase, the system performance was compared to the design goals as stated in Section 1.2. This section discusses the test results for all the sub goals except the design, implementation and testing of the system.

Improved availability of data

During the tests, all saved data was automatically and immediately distributed to all involved users. As the test was performed in a lab setting, it could not be tested whether the data would be available at a true incident scene. However, as long as a network connection to the server computer was available, the IMICS client could always connect to the server and did receive all updates correctly. The availability of context information

has not been implemented. By adding a few hidden fields to every data field and extending the user interface, this would be completely supported though. Overall, the goal of improved availability of data is completely supported by the IMICS design.

Improved reliability of data

Each time a user saved data to the system, the interface on all clients was automatically updated. As mentioned in Section 7.3.2, an issue was found where the interface did not update correctly, but this issue was corrected and only the latest data was always displayed.

A few mistakes were made during the tests, and most of them were noticed by other users and corrected. It can be concluded that the large user group does indeed help correcting mistakes.

Most information has been entered into the system correctly and users generally quickly found where data had to be entered, so in general the used ontology was quite clear. The tests did reveal some ambiguity though, as mentioned in Section 7.3.3. One of the issues has been improved upon, but two issues remain. In section 8.2 a solution to these issues is suggested.

In the tests, the information in the system was generally very reliable as very few mistakes were made and most were corrected by the users. An extended field test will have to determine if this also holds in practice and if the system really improves the reliability of data in comparison with the current procedures. For the user test however, it can be concluded that the goal of improved reliability of data is supported by the system.

All information and communication joined in one device

As the IMICS system was not fully implemented, not all information was shared through IMICS. In the scenarios described in Section 7.2.2, the numbers in brackets indicate events that should be stored in the system. Only the numbers in bold could actually be stored in the currently implemented part of IMICS. In the system design however, all information can be stored and shared.

Since the IMICS system is a lightweight Java based application, it should run on any modern PDA supporting Java. In this way, the system joins IMICS and phone calls in one device.

Logging of data and events

As the user tests showed, all updates were correctly written to the log at the server. The client systems also keep a log of their received updates, so if something goes wrong it can be determined where the error took place. As the logs are stored in a readable format, they can be accessed directly for evaluation.

Availability of context of information

As each update is stored in the log with information such as the time it was received and the provider of the information, the context of the information is available.

7.3.5 General remarks

Entering data into the system, especially during the first phase of the incident where most new information is received can take a few minutes. Users that are unfamiliar with the system take even longer. In one particular case, it took a test participant over a minute to find where to enter one specific bit of information. Although the separate tabs impose a clear structure on the information in the system, users that are not familiar with the system or the incident management domain and its procedures may have trouble finding the correct fields quickly.

During the tests, it became apparent that users quickly became familiar with the IMICS system, performing much faster when they participated in a second or third test. This shows that training and knowledge of the system is important in practical use. As mentioned in Section 4.1.3, users should receive proper training and use the system on a regular basis.

As was expected, creating a new incident and entering its first information is the most time consuming part of the incident response concerning IMICS. As the control room is currently responsible for collecting this information and processing it, it seems once more that the shared control room is where the majority of the data is entered into the system as was anticipated in the design.

7.4 Summary

Before formulating conclusions in Chapter 8, this section shortly recapitulates the results discussed above.

Theory

Although not everything has been implemented, the design premises from Turoff et al. [10] have all been taken into account in the design of IMICS. If the not implemented improvements mentioned in Section 6.4 are included in the design, most envisioned roles are supported too. The lack of a summary as described in the fifth role is compensated by the clarity of the ontology. Roles 1, 2, 6, 7 and 11 all involve resource management. These tasks are carried out by the control rooms and the emergency services themselves, and they have specialised software in place to support this. IMICS could be coupled to these systems, but currently does not provide this functionality itself as this is easily accomplished by these systems. The remaining roles 3, 4, 8, 9, 10 and 12 mainly comprise coordination and the sharing of information and are fully supported by the IMICS design.

Most of the design principles from the theory are supported by the IMICS design. Not all features have been fully implemented though. The first principle is supported except for a text search feature, which should be implemented in a future version. Principle 2, 5 and 8 are supported by the design, but require some extra features to be implemented as described in Section 6.4. Design principle 7 is supported by the design of IMICS, but has not been implemented. Principles 3, 4 and 6 are fully supported.

Test results

The tests revealed a few shortcomings to the system. First of all, an important bug was found that prevented newly received data to be displayed correctly after a new incident had been created. This bug was corrected. Some other issues with the interface have been found as discussed in Table 32. The names of the states of the traffic measures were slightly ambiguous and have been improved. A solution to the other two issues found is suggested in Section 7.3.3.

The tests showed that in a lab setting, the IMICS concept provides the functionality it was designed for. The system design offers improved availability and reliability of data and joins information and communication in one device. By logging all data and its context, analysis and evaluation of the incident response is supported and old incident data can be used for training personnel.

Chapter 8

Conclusions and recommendations

The IMICS system is designed to improve situational awareness of the emergency services during the handling of traffic incidents on the Dutch motorways and to improve evaluation of the incident management procedures. It is a blackboard like system in which incident data can be shared between users from different organizations at different locations. For clarity, the goals of this thesis as stated in Section 1.2 are repeated in Table 33.

Table 33 - The project goals as stated in Section 1.2.

Improve communication and situational awareness by:
1 Improved availability of data
2 Improved reliability of data
3 All information and communication joined in one device
Improve analysis, training of personnel and evaluation by:
4 Logging of data and events
5 Availability of context of information
Design, implement and test a working prototype
6 Create system design
7 Implement prototype
8 Test prototype

In order to be able to test the system design, a prototype has been implemented. Based on the existing JADE framework, a system infrastructure has been implemented providing blackboard like functionality combined with a centralized server. On top of this infrastructure, an interface has been designed that uses a specially designed ontology to provide its users with unambiguous information. Together, these components form a fully functional prototype that has been tested in a lab setting.

Although the created proof of concept is not a complete system and some refinements will have to be made, all persons interviewed during this project were very enthusiastic

about the general system design. As the previous chapter discusses, by sharing information with all stakeholders, the system improves one of the biggest bottlenecks of current incident management.

In this chapter, the general conclusions drawn from the results in the previous chapter are presented. First, in Section 8.1 the project goals are evaluated. After a short reflection in Section 8.2, a number of recommendations for the further development and deployment of IMICS is given in Section 8.3. Finally, Section 8.4 gives a short summary of the conclusions and recommendations.

8.1 Conclusions

As the previous chapters discuss, a design for IMICS has been made and partially implemented and tested in a lab setting as stated in the project goals. When the theory from Chapter 3 is compared to the current situation, it turns out that to a certain extent most design principles from theory are already supported by the procedures and systems in use. This is not surprising as the current incident management procedures are based on years of practical experience. Still, IMICS does provide a few advantages when compared to the current procedures. We will now evaluate the project goals.

1: Improved availability of data

With the implemented prototype, all users automatically receive updates from other users everywhere. The lab tests discussed in the previous chapter showed all updates are handled correctly, making new information available to the users assigned to the same incident. It can be concluded that this project goal has been reached.

2: Improved reliability of data

The prototype always displays the latest data only, ensuring that users are kept up to date. As the tests showed, the fact that multiple users can access the same data helps to correct mistakes and improves reliability of the data in general. The use of a fixed ontology that follows the incident management procedures reduces ambiguity. The user interface also shows what data is required, serving as a reminder to its users. This goal has been reached.

3: All information and communication joined in one device

The IMICS prototype is designed to run on a PDA. In this way no extra communication equipment is required. Sharing data by phone or through the IMICS system can both be facilitated by a PDA, effectively joining communication in one device and fulfilling this project goal.

4: Logging of data and events

The IMICS system provides a complete log of the data sent and received by both the server and the clients. The lab tests showed the logs to work correctly. Since IMICS combines all information into one system, its logs contain the shared data

of the stakeholders. This means that the complete incident response can be evaluated for all stakeholders. This goal has been reached.

5: Availability of context of information

As all information is stored with context information such as the time it was received and the provider of the information, this goal has been reached.

6: Create system design

As chapters 5 and 6 discuss in detail, a system design has been made that allows users to share incident information in a clear and unambiguous way. A general design has been made and the part specifically aimed at incident information has been worked out in detail, fulfilling this project goal.

7: Implement prototype

The part of the system design that has been worked out in detail has been implemented in a prototype. It consists of three components. The first is a supporting infrastructure provided by an existing framework called JADE. The second component is the task layer that provides the blackboard like functionality that allows data to be shared among IMICS clients and the IMICS server and an ontology describing the kind of information that is communicated. The third component is the presentation layer that provides a clear user interface. The latter two components have been implemented for this thesis specifically, providing a running prototype and fulfilling this project goal.

8: Test prototype

The implemented prototype has been tested in two ways. First of all the IMICS components have been tested during the implementation phase to ensure their correct behaviour. Secondly, the system design has been tested in a lab test. Several scenarios, involving different stakeholders, have been tried with the system in a controlled environment. The tests showed that the IMICS design delivers the intended functionality and that the project goals have been reached in a lab setting.

8.2 Reflection

The main advantage of the proposed IMICS system is the fact that most functions are coupled in one system that is available to those that need it everywhere and at any time. IMICS can be seen as a spider in the web, connecting employees from the emergency services and sharing information among them.

The system design was discussed with the Program Manager Incident Management at the Dutch national traffic control centre, mr. Eeltje Hoekstra. According to him, in the current situation, little information is directly shared between the stakeholders. A test project from the Department of Public Works, called the Man In the Middle, allowed users to post a short message to a shared server, so that other involved users could read it. As this test showed, sharing information is seen as a great improvement by all

stakeholders. (As noted by both Eeltje Hoekstra and Peter Grinwis, see Appendix B). As mr. Hoekstra indicated during our interview, one of the main risks to the system is privacy. The involved organizations may be reluctant to share information, and they are especially reluctant to allow access to their private networks. The IMICS system could be coupled to the existing systems such as the shared control room system (GMS), but it can also be used separately, allowing the stakeholders to enter relevant information into the IMICS system without compromising the privacy and security of their network. Whether information is shared automatically or entered in the IMICS system separately, the IMICS system provides a means to share information between stakeholders easily.

The goals of this thesis have all been reached by the IMICS design in a lab setting as is discussed above. As users enter information into the system, the data is automatically received by all signed in users and other users can check and update this data. All updates are stored in the log for future reference. It has not been proven that the subgoals actually lead to the main goals in practice however. The user tests took place in a controlled environment and a few assumptions were made. First of all, the functionality not provided by IMICS was assumed to be taken care of in the tests. Secondly, in the scenarios, all required data was readily available. Though the system follows the standard procedures for obtaining information and shows what information is required for each step, in practice not all data may be as easily obtainable. Moreover, since users may be occupied by their main tasks, data that is available may not always be entered into the system. Since a test in a lab setting cannot simulate these facts realistically, a field test is required to determine if the results found in this thesis hold in practice. Summarizing, proving that the results hold in practice would require thorough practical evaluation, which is unobtainable in the scope of this thesis. In theory and in a lab setting however, the concept of IMICS works to improve communication, situational awareness, training, analysis and evaluation.

8.3 Recommendations

Following the test results and conclusions, a few recommendations can be made. These recommendations are divided in two categories: recommendations regarding the further development of the IMICS system and recommendations regarding practical deployment of the IMICS system.

8.3.1 *Further development*

The current implementation of IMICS provides the basic framework on which the entire system runs and a part of the envisioned functionality. A number of tests in a lab experiment have shown that the concept of IMICS is indeed a valuable addition to incident management. This section discusses in which way the system should be developed further.

Discuss with stakeholders

As the interviewed people were very enthusiastic about IMICS, the next step is to get interest and support from the emergency services and the Department of Public Works

for a follow-up project. As this project is a mere proof of concept, its results must be discussed with the intended stakeholders. Demands on functionality, security and privacy should be fully satisfied and further refinements may have to be made before the system can be put to practical use.

Refine implementation

The improvements suggested in Section 6.4 should be implemented. As discussed in Section 7.1, a text search function should also be added to IMICS. The remaining shortcomings to the user interface found in the user tests, as discussed in Section 7.3.3, should also be addressed.

Conduct field test

A field test should be conducted to see if the proposed system really works in practice. The impact of the envisioned system to the work of its stakeholders must be analysed. IMICS should not distract from work or hamper the incident response in any way and its positive impact on the incident response should be evaluated. Security, scalability and robustness should also be evaluated in the field.

8.3.2 IMICS deployment

As functions are added, IMICS could eventually replace existing communication and information systems used in incident management. However, since 100% network reliability can not be guaranteed, backup facilities must always be available. The current procedures, in which field officers communicate with the control rooms by phone or C2000, seem very suitable since they are fully supported and proven in the field. In its current form however, IMICS has a primarily supporting role as a spider in the web. The emergency services have their own systems in use for tasks such as tracking and tracing and resource management and IMICS simply provides a shared repository that allows its users to build a correct mental image of the incident situation, improving situational awareness.

Include IMICS in training and exercises

As it is important that using IMICS does not occupy users for too long and distract from other tasks, the system should be familiar to its users. For that reason, if IMICS is adopted in the field, the use of IMICS must be included in training and exercises.

Legal and financial aspects

It is very important that the privacy of the users and the victims of an incident is sufficiently protected. Also the costs of developing and deploying IMICS must be weighed against its advantages. For example, when a next generation of C2000 phones is bought, it could be more expensive when it has to be able to support IMICS. (However, since IMICS is a fairly lightweight Java application, most modern PDAs support it).

8.4 Summary

The IMICS system provides an improvement to the current incident management procedures. As the previous sections discuss, a system design was made and a running prototype was implemented and tested in a lab setting. Table 34 summarizes the advantages of the IMICS system.

Table 34 - Summary of the advantages of IMICS.

- | |
|---|
| <ol style="list-style-type: none"> 1. Improved availability of data. (Goal 1) <ol style="list-style-type: none"> a. IMICS serves as a spider in the web, sharing information and functionality with users from different organisations anywhere at any time. b. Sharing information and its context across organizations helps improve situational awareness. c. Data is automatically distributed to all involved users. d. Data is available at all locations. e. The context of information is available. 2. Improved reliability of data. (Goal 2) <ol style="list-style-type: none"> a. Only the latest data is shown. b. Shared access increases the chance of errors being corrected. c. The ontology provides a clear and well structured overview of the situation, reducing ambiguity. d. The ontology follows the information needs of the stakeholders, reminding users of the information required by other users and providing the necessary data. 3. One device is used for both IMICS and telephone. (Goal 3) 4. The availability of all updates in the log improves evaluation of procedures and personnel for all stakeholders. (Goal 4) 5. The availability of context information in the log could explain certain decisions made and improves evaluation. (Goal 5) |
|---|

As the results in Section 7.3 show, the goals of this thesis as stated in Section 1.2 have all been reached and in a lab setting. It has not been proven that the goals are achieved in practice however. A realistic field test is required to show whether these results can be reproduced in practice. Following the conclusions, a number of recommendations were made in Section 8.2. Table 35 summarizes them.

Table 35 - Summary of the recommendations following the conclusions.

- | |
|--|
| <ol style="list-style-type: none"> 1. The IMICS implementation must be refined in cooperation with the stakeholders. <ol style="list-style-type: none"> a. Discuss with stakeholders and gain support for follow-up project b. Refine IMICS design in cooperation with stakeholders 2. The IMICS implementation must be extended to include additional functionality. <ol style="list-style-type: none"> a. Security and advanced synchronization b. Search function c. Improvements to the client interface d. Server interface 3. The IMICS system must be tested in a realistic field test <ol style="list-style-type: none"> a. Test research goals and security, scalability and robustness in practice. |
|--|

Chapter 9

Future work

In the previous chapter, recommendations were made for the completion and deployment of the IMICS system. This chapter, on the other hand, presents additional functionality that could be added to the system to improve its impact on incident management.

9.1 Technical improvements

Some improvements could be made to the system design that increase flexibility. The following suggestions may require some adaptations to the implemented part of IMICS.

Advanced storage

The current system stores incident data in local memory and writes it to a log file on the server hard disk. Although simple storage systems will do for early development and testing, more sophisticated solutions will be needed to take care of synchronization and consistency when the system is fully operational. A data or information repository would be a good solution, taking care of synchronization automatically.

Web services

JADE has an interesting add-on; The Web Service Integration Gateway allows automatically exposing agent services registered with the DF as Web Services. This add-on can be found on the JADE website [17]. Although this is a very useful feature, especially when linking IMICS to other systems, the agents in the current implementation have not been implemented with this option in mind and may have to be adapted for this to work.

Custom forms

Some users may only need a very limited subset of the information available in IMICS. It is possible to extend the system so that users can create their own customized interface. For example, the client agent could be published as a web service that provides the information and a specialized service that provides the customized user interface.

9.2 Extended functionality

The functionality provided by the current IMICS system could be further extended in many ways. A few suggestions are made here.

Include other stakeholders

First of all, IMICS is currently designed for the most important stakeholders only. Future versions should be extended to allow access to other incident management stakeholders.

Although this was deliberately not done to limit the scope of the project, including the other stakeholders in the design would increase the advantage the system offers. It should at least be possible to keep track of their tasks and estimated time of arrival in the system. This does not necessarily require these stakeholders to have access to the system themselves. However, since the system offers the biggest advantage in complex incidents with many involved organizations, access to the system by the other stakeholders is recommended. It is very important to analyse the information needs and tasks of these stakeholders to provide optimal functionality to all of them.

Include secondary road network

As currently the Dutch incident management procedures are only applied to the motorways, so is the current IMICS design. IMICS could be extended to provide the same functionality for incidents on the secondary road network. This would require the ontology to be extended with different types of location. For the rest of the IMICS design it may have little impact, as the main difference would be in the procedures used by the emergency services, and not in the need for cooperation and information. However, this would have to be carefully analyzed.

Extended logs and evaluation tools

The current logs are basic text files in which received updates are simply concatenated to the end of a file (a new file is created every day). These text files may become quite large and finding specific data or a specific update can be cumbersome. A special user interface could be created to provide search functions or to show the changes in time in a more elegant overview. Furthermore, the system could be extended with statistical tools and graphs to offer improved evaluation. Such systems could for example compare incidents and find bottlenecks in the incident management process.

Include safety checks

Since safety is regarded the most important aspect of incident management (see Chapter 4.2.1 Priorities), it might be desirable to include safety procedures in the system. The system could provide a simple checklist with safety procedures ordered on basis of priority. When dangerous substances are involved, a warning could be issued to all involved personnel. The system could also provide prioritized task lists and issue an alert message when priorities or tasks should be adapted due to changes in the situation.

9.3 Connection to other systems

The IMICS system could be linked to several systems in use or under development. The following overview gives a number of interesting examples IMICS could be connected to or integrated with.

The shared control room system

The control rooms shared by the Dutch police, fire brigade and ambulance service have recently taken a new IT system into use. This is called the *Gemeenschappelijk Meldkamer Systeem* (GMS) in Dutch. This system actually provides much of the functionality offered by IMICS to the control room. Combined with IMICS, this would form a powerful tool that provides all stakeholders with the necessary information at any location.

The national datawarehouse

The national datawarehouse is a data repository that contains up-to-date road and traffic information. It could provide emergency services with the latest traffic information and provide alternative routes in case of congestion. The other way around, the emergency services could provide information of blocked lanes to IMICS and through the coupling, directly to the datawarehouse.

Road details

At the traffic control centres, an overview of the complete Dutch motorway network is kept in an advanced IT system. It also keeps track of traffic conditions and traffic measures. The amount of lanes and shoulders present at a location now have to be entered into IMICS by hand. Linking to this system could allow the amount of lanes and an overview of the traffic situation to be generated automatically after entering the motorway, the hectometre number and the direction.

Vehicle information systems

The Dutch 'Rijksdienst voor het Wegverkeer' (shortly RDW) is responsible for vehicle registration in the Netherlands. Information about the vehicles involved in an incident could be automatically obtained by connecting IMICS to the IT systems in use by the RDW. Currently, vehicle information can be obtained by providing the vehicle's registration number at their site [33].

eCall

Systems are currently being developed that could automate some of the information provision. An example is the eCall system [34]. Cars equipped with the eCall system would automatically send a notification to the emergency control room when involved in an incident. It could automatically provide details about things such as the car type and registration number, the amount of persons in the car and impact speed.

Theoretically, in the nearby future, a vehicle could keep track of the persons in it and automatically send their details, including medical details (for example through the electronic patient file) to the ambulance service when an incident occurs. This could lead

to privacy issues though, so a solution may be found in providing medical information on a voluntary basis.

Geographic information systems

As suggested in the global design of the IMICS interface (see Section 5.3) IMICS could be integrated with a geographic information system (GIS). Such systems present an overview of the incident location and could be coupled with tracking and tracing, route information and driving time estimation as is offered by certain existing systems (examples are TOMTOM and the ANWB navigation systems).

To take things even further, the GIS systems could be extended with traffic simulation as for example in [35].

Photos and maps

The system could also provide an option to store pictures of the incident situation. As most PDAs are equipped with a camera, this could be easily included in the system.

Lately, a test project has been developed by the Department of Public Works in which users were able to post pictures and GPS coordinates of an incident to a website. According to the Program Manager Incident Management at the Dutch national traffic control centre, Eeltje Hoekstra, the project's results were promising and it was generally received very well by the emergency services.

Bibliography

- [1] L.H. Immers, “Visie Incident Management”, TNO 2007.
- [2] “Directive Initial Safety Measures for Incidents on Motorways in the Netherlands”, Verkeerscentrum Nederland, 2005.
- [3] http://en.wikipedia.org/wiki/OODA_Loop
- [4] Berenschot, “Eindrapport Landelijke Evaluatie Incidentmanagement”, Utrecht, 2002.
- [5] R. Iannella, K. Henricksen, “Managing Information in the Disaster Coordination Centre: Lessons and Opportunities”, *ISCRAM 2007*.
- [6] J.R. Steenhuisen, M.M. de Weerd, C. Witteveen, “Enabling Agility through Coordinating Temporally Constrained Planning Agents”, *ISCRAM 2007*.
- [7] H. Cramer, “Adaptive Information Distribution to Support Human Collaboration”, *ISCRAM 2007*.
- [8] <http://nl.wikipedia.org/wiki/Veiligheidsregio>
- [9] “The Roles of the Emergency Services in Incident Management in the Netherlands”, Verkeerscentrum Nederland, 2007.
- [10] M. Turoff, M. Chumer, B. Van de Walle, and X. Yao, “The Design of a Dynamic Emergency Response Management Information System (DERMIS)”, *The Journal of Information Technology Theory and Application (JITTA)*, 5:4, 2004, 1-35.
- [11] R.R. Dynes and E.L. Quarantell, “Organizational Communications and Decision Making in Crises,” Disaster Research Center Report 17, January 1977, Ohio State University.
- [12] D. Benyon, P. Turner, S. Turner, *Designing Interactive Systems*, Addison-Wesley 2005.
- [13] “MultiDisciplinair slotsscenario C2000, Eindrapportage”, Veiligheidsregio Rotterdam-Rijnmond, 2006.

- [14] F. De Rosa, V. Di Martino, L. Paglione and M. Mecella, "Mobile Adaptive Information Systems on MANET: What We Need as Basic Layer?", *Proceedings of the Fourth International Conference on Web Information Systems Engineering Workshops (WISEW'03)*.
- [15] I. F. Akyildiz, W. Lee, M.C. Vuran, S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey", Elsevier, 2006.
- [16] N. Bharosa, J. Appelman, P. de Bruin, "Integrating technology in crisis response using an information manager: first lessons learned from field exercises in the Port of Rotterdam", *ISCRAM 2007*.
- [17] <http://jade.tilab.com/>
- [18] <http://www.xml.com/>
- [19] OASIS Emergency Management Technical Committee, 2005. Common Alerting Protocol, v. 1.1. OASIS Standard CAP-V1.1, October 2005.
- [20] <http://www.fipa.org/specs/fipa00061/SC00061G.pdf>
- [21] E. Rohn, "A Survey of Schema Standards and Portals for Emergency Management and Collaboration", *ISCRAM 2007*.
- [22] <http://cougaar.org/>
- [23] <http://lime.sourceforge.net/Lime/index.html>
- [24] <http://www.fipa.org>
- [25] <http://jade.tilab.com/community-faq.htm>
- [26] <http://jade.tilab.com/doc/JADEProgramming-Tutorial-for-beginners.pdf>
- [27] F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa, "Jade - a white paper", September 2003.
- [28] <http://jade.tilab.com/doc/administratorsguide.pdf>
- [29] F. van Harmelen, V. Lifschitz, B. Porter, *Handbook of Knowledge Representation, Foundations of artificial intelligence*, Elsevier 2008.
- [30] A. Abdoullaev, *Reality, Universal Ontology, and Knowledge Systems: Toward the Intelligent World*, IGI Global, 2008.
- [31] <http://protege.stanford.edu/>
- [32] http://protegewiki.stanford.edu/index.php/OntologyBeanGenerator_4.0
- [33] <https://www.rdw.nl/Ovi/Paginas/Default.aspx>
- [34] "Toepassingen e-call verkeer en transport", Ministerie van Verkeer en Waterstaat, 2005.
- [35] B. Huang, X. Pan, GIS coupled with traffic simulation and optimization for incident response, Elsevier 2006.
- [36] [http://en.wikipedia.org/wiki/Blackboard_\(computing\)](http://en.wikipedia.org/wiki/Blackboard_(computing))

- [37] http://en.wikipedia.org/wiki/Service-oriented_architecture
- [38] http://en.wikipedia.org/wiki/Event_Driven_Architecture
- [39] <http://en.wikipedia.org/wiki/Peer-to-peer>
- [40] A. Smirnov, M. Pashkin, N. Shilov, T. Levashova , “Intelligent Support of Context-Based Megadisaster Management: Hybrid Technology and Case Study”, *ISCRAM 2007*.
- [41] N. Netten, G. Bruinsma, M. van Someren and R. de Hoog, “Task-Adaptive Information Distribution for Dynamic Collaborative Emergency Response”, *International Journal of Intelligent Control & Systems*, 2006.
- [42] J. Löffler, V. Hernández Ernst, J. Schon, J. Pottebaum, R. Koch, “Intelligent Use of Geospatial Information for Emergency Operation Management”, *ISCRAM 2007*.
- [43] B. Johansson, J. Trnka, R. Granlund, “The Effect of Geographical Information Systems on a Collaborative Command and Control Task”, *ISCRAM 2007*.
- [44] “Incident management in the United States: A state-of-the-practice review”, Texas Transportation Institute, 1997.
- [45] <http://www.telecomwereld.nl/n0001553.htm>
- [46] <http://dekkingskaart.t-mobile.nl/coverage/>
- [47] <http://nl.wikipedia.org/wiki/WiMAX>
- [48] <http://tweakers.net/reviews/662/all/mobiel-breedband-nu-en-in-de-toekomst.html>
- [49] <http://www.c2000.nl>
- [50] http://www.verkeerenwaterstaat.nl/actueel/nieuws/nieuwsarchief/pb-Veiling_UMTS-frequenties_succesvol_verlopen.aspx
- [51] <http://AAF.freeband.nl>
- [52] H. Kosch and M. Döller, “Multimedia Database Systems: Where are we now?”, *IASTED DBA Konferenz*, 2005.
- [53] <http://www.nationaaldatawarehouse.nl/>

Appendix A

Techniques

This section gives an overview of some of the techniques that have been looked into at the start of this project. Most of the text here is from the literature survey that was performed before the system design and did not fit directly into this thesis. It is meant to give a short (and incomplete) overview of the many possibilities information technology offers in the field of incident management. Although some parts of the texts may seem outdated since new insights were gained during the project, some of the options mentioned here have been incorporated in the IMCIS design.

System architectures

The following architectures have been considered during the literature survey. Many of them offer overlapping functionality and are at least partially incorporated in the IMICS design.

Human operated control room

In traffic incident management in the Netherlands, a control room is always involved. One of the possibilities for the IMICS system is to simply collect all information in the control room and have one person, an information manager, check and distribute it to the right persons as in [16]. This approach is quite similar to the current situation. In all probability, this possibility will always be supported as a backup system, but it is not the goal of the IMICS system.

Content Management Systems

A content management system (CMS) is used to manage content in an interactive user group (e.g. a company or a web based community). The content may include for example data files, multimedia files and spreadsheets. These files can be accessed by registered users on company networks or the internet. When a large number of contributors is involved, it is important to keep data up-to-date and functions such as version control become important, both of which are functions supported by a CMS. Other features usually supported are user roles and rights management and user notification when updates are available or action by a specific user is desired.

Web-based information systems

Web-based information systems are systems which can be accessed with a standard web browser. They can be used to store or retrieve data, fill out forms or to search a database. Web-based information systems can be used as a CMS. The advantage of web-based information systems is that any person with access rights can access the system from any personal computer with an internet connection.

Wiki systems

Wiki systems are a kind of collaborative software which can be accessed through standard web browsers. They allow users to update and complete other users' documents. Information can be collected collaboratively and can be linked to other relevant documents. In this way, wikis can be used as a form of content management system. Although wikis are very flexible, the fact that any information can be edited by any user with the necessary access rights makes it hard to check consistency and requires regular reviews. Generally wiki systems provide a function to see recent changes. The organization of information and links can be challenging and may require editing by hand. The lack of structure in the information makes this architecture less suitable for the incident management domain.

Blackboard systems

A blackboard system is modelled after the physical concept of a blackboard. As data is entered into the system by either a user or an automated process, other components or users can use this information to update their knowledge and come up with new solutions. These new solutions can once more be used by others, comparable to students using a professor's notes and collaboratively working out solutions on the same blackboard. Blackboard systems are composed of three components. The first are software specialist modules, called knowledge sources, which provide specific expertise needed by the application. The second is the blackboard itself, a shared repository of knowledge, problems and (partial) solutions provided by the knowledge sources. The final part, the control shell, controls the flow of problem-solving activity in the system, organizing the use of knowledge sources in the most effective and coherent way [36].

Service oriented Architectures

Service Oriented Architecture (SOA) is an architectural style in which functionality is decomposed into distinct units called services. These services can be distributed over a network and exchange data independent of the operating systems and programming languages underlying those applications. Combining these services specialized business applications can be constructed adapted to the users' need [37].

Event-driven systems

An event-driven architecture (EDA) is a software architecture pattern based on triggering or responding to events. This architectural pattern may be applied by the design and implementation of applications and systems which transmit events among loosely coupled software components and services. An event-driven system typically consists of event consumers and event producers. Event consumers subscribe to an intermediary event manager, and event producers publish to this manager. When the event manager

receives an event from a producer, the manager forwards the event to the consumer. If the consumer is unavailable, the manager can store the event and try to forward it later.

Building applications and systems around an event-driven architecture allows these applications and systems to be constructed in a manner that facilitates more responsiveness, because event-driven systems are, by design, more normalized to unpredictable and asynchronous environments. Event-driven architectures complement service-oriented architectures (SOA) because services can be started by triggers such as events [38].

Peer-to-peer systems

In a peer-to-peer system, instead of clients communicating through a centralized server, each computer in the network is treated as an equal node, functioning as client and server simultaneously. Every computer can be connected to any other in the same network, possibly sharing content or resources. Usually the connections are managed in an ad-hoc fashion, meaning that connections are made and broken when needed.

The distributed nature of peer-to-peer networks also increases robustness in case of failures by replicating data over multiple peers, and -- in pure P2P systems -- by enabling peers to find the data without relying on a centralized index server. In the latter case, there is no single point of failure in the system [39].

Services and functions

At the moment, many techniques are being developed to support incident and crisis management all over the world. Although some of these techniques have been fully developed and used in practice, many are still in development or being tested. The functions and techniques mentioned here could support incident management. Once more, this is not meant to be an exhaustive list, but merely a list of options that were considered for IMICS.

Integrated communication

Since one of the main goals of the system is to improve the communication between emergency services, it makes sense to combine all communication lines in one system. With the system, it should be possible to reach all stakeholders and obtain all relevant information.

Phonebook and address book

It might not be possible to equip all stakeholders with the system, therefore it should be possible to communicate with them in other ways. To support this, it should be possible to look up addresses or phone numbers.

Human operator

Although IMICS could be a standalone software application, it could be beneficial to have a human operator or supervisor in the system. This person could monitor communication and progress and advise or assist the emergency respondents at the incident scene. In the role of supervisor, an operator could verify the consistency of the information in the system [16].

Decision support systems

In chaotic situations, people can get overwhelmed by large amounts of information. To combat this, decision support tools can be added to the system. Using artificial intelligence techniques like a knowledge base and inference rules it is possible to infer the best next action. An example of a decision support system is described in [40].

Relevance inference

Another way to prevent people from being overwhelmed by large amounts of data is to filter the incoming information. By using artificial intelligence techniques, statistics, or simple priority rating systems, irrelevant or less important data can be ignored in stressful situations. These same techniques can be used to automatically send new information to the persons (or roles) to whom it is relevant. A good example can be found in [41].

GIS (Geographic Information Systems)

Geographic Information Systems integrate data on a geographic representation (e.g. a map). In this way, users get a clear overview of the situation. Adding a 3D representation might help emergency respondents to get an even better situational understanding [42]. Some possible features of GIS include superimposing high risk zones over maps and locating nearest emergency services. Research suggests that GIS can significantly improve incident response [43].

Tracking & tracing

Using the Global Positioning System (GPS) it is possible to keep track of all dispatched emergency vehicles. This allows finding the nearest emergency services and suggesting alternative routes when congestion has occurred.

Logging

Since evaluation is considered a very important aspect of improving the incident response, logging of all data is very important. Every change to the information in the system should be recorded, along with time and responsible persons. This information can later be used to find bottlenecks in the incident response and to suggest improvements. If mistakes have been made, the people responsible can be traced, which might result in suggesting training courses for example.

Statistics

To support the evaluation of the incident response even better, the system can be augmented with extensive statistical tools. Using standard templates for queries and graphs to find irregularities like excessive clearing time could help trace procedures which could be improved.

Automatic data gathering

There are many tools to automatically register an incident. Examples are the eCall system under development in the European Union [34] and electronic surveillance. More examples can be found in [44]. Although at the moment incidents are still reported to the control rooms, the possibility to automatically detect incidents should be taken into

account. Another example of automatic data gathering is to automatically request vehicle registration information, saving time in the process.

Synchronization

If the system is to be used on a large scale, synchronization becomes an issue. What happens if two people try to update the same data for example? Another important issue is clock synchronization, if someone tries to update information while his clock lags behind, the system might discard his update since it appears to be older than the current information. For evaluation purposes this is also important.

Many advanced techniques have been developed for these issues (some of which are available in of the shelf solutions). For this project we acknowledge the problem, but at the moment we will not discuss it too deeply. To cope with clock synchronization a centralized database seems to be a good choice, since the system could then simply use the clock on the server.

Flexibility

Though a well structured approach is used in incident management, flexibility is still important. It should be possible to reassign roles for example. It may also be desirable to add the possibility to freely enter additional information (e.g. explaining why a certain task took longer than usual).

Supporting network

Some of the existing network infrastructure solutions are listed below.

GSM network

The existing Dutch GSM network provides a generally stable infrastructure, facilitating reliable voice communication and fast data communication through standards such as GPRS, its successor UMTS or even faster standards like HSDPA which theoretically supports speeds up to 14Mbps. Network coverage is generally good, for example HSDPA coverage of the KPN Network was about 90% in spring 2007 according to [45]. Figure 1 shows coverage of the T-Mobile Network according to the T-Mobile website [46].

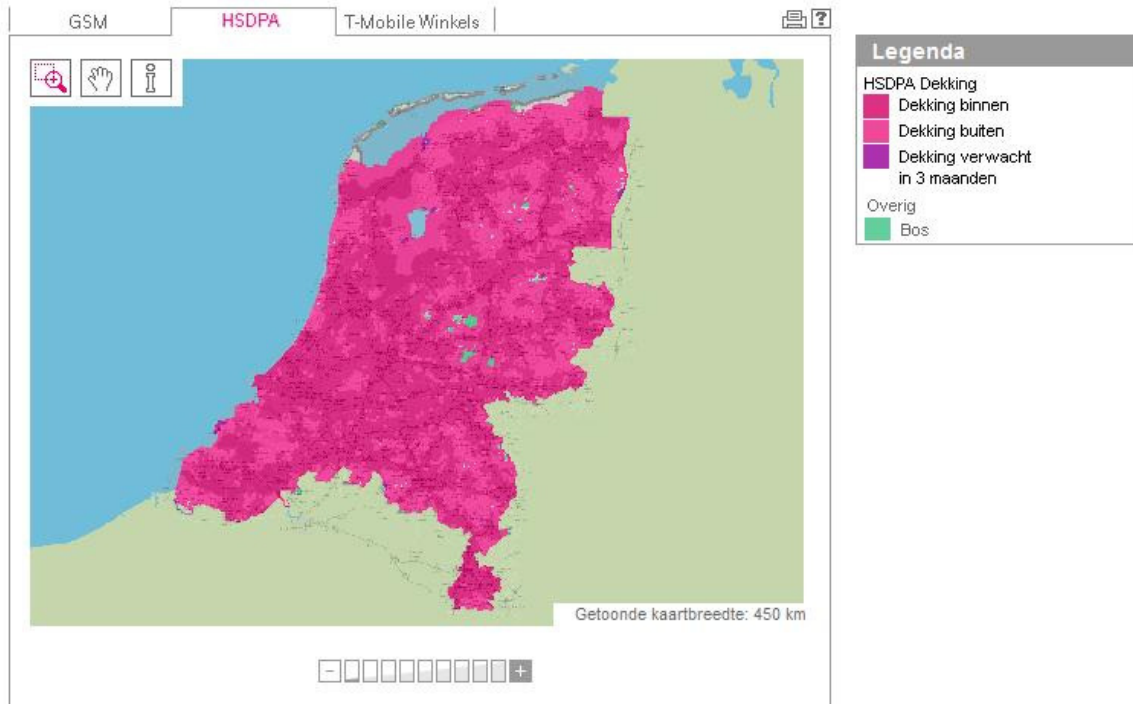


Figure 34 - GSM coverage by a Dutch telecom provider (T-Mobile).

Wi-Fi

Wi-Fi is a wireless data communication technique supporting high speed data communication (300 Mbps for modern routers), often used to implement wireless home or office networks (Ethernet). It is available in so-called hotspots, but has very limited range though (usually less than 30 meters) and therefore is not very applicable in incident management, unless it is available on site. In combination with a mobile base station with access to a different network (e.g. satellite internet or the GSM network) it could be used, but its use would still be limited. Wi-Fi has a relatively high energy usage and may therefore be inappropriate for mobile communication devices. Interference by other signals can lead to disruption of the network, making it less reliable.

WiMAX

WiMAX (Worldwide Interoperability for Microwave Access) is a standard based on the IEEE 802.16 standard for wireless broadband networks with medium range. Theoretical range is about 50 kilometres and maximum data transfer speed is 70 Mbps. These numbers are for fixed stations though. For mobile stations in practice, transfer speeds are below 10 Mbps, and the range is generally limited to about 2 kilometres. The available bandwidth has to be shared with all users in both directions. Since WiMAX has a more master-slave oriented structure, in which the base station has full control over transmissions, WiMAX guarantees a better quality of service than Wi-Fi. WiMAX is sensitive for interfering signals, just as Wi-Fi. Another disadvantage of the WiMAX standard is that it is not yet widely available in the Netherlands and it might still be a while as suggested in [47].

A more detailed overview of existing network standards (in Dutch) can be found at [48].

C2000 network

The C2000 network has replaced the nearly 100 outdated analog networks used by the Dutch emergency services previously. It allows fire department, ambulance service, police and military police (Koninklijke Marechaussee in Dutch) and associated organisations to communicate with each other directly using specialised phones. Since C2000 is one nationwide network with at least 95% coverage, emergency services can communicate reliably [49]. The network could be used as the supporting network for data communication for the IMICS system. An advantage of the network is that security issues have already been taken care of by means of built in encryption. The network has also been designed to function reliably under extreme circumstances. Many crucial parts of the network have backup facilities. Since most technical details of the C2000 system are classified though, little information is available about data transfer speeds and accessibility of the network. In field tests some problems were shown to exist, these are expected to be solved in the coming years though [13].

Satellite internet

To use satellite communications for internet, a satellite antenna dish is generally required. There are mobile phones available that can communicate through a satellite network (e.g. Iridium). These phones are usually very expensive though and calls are very expensive as well. Data transmission over the satellite network is possible, but offers very limited bandwidth. Altogether, for local communications with generally good GSM coverage available in the Netherlands, satellite phones do not seem to offer a very large advantage.

Short discussion

As mentioned in the main body of this report, the prototype is designed for the GSM network. The GSM network provides fairly fast communication and is widely available, making it a very good choice. Security issues will have to be solved though, but we do not focus on this subject in this project. Network availability could be improved by using ad-hoc networking.

WiMAX could provide a good alternative to the GSM network. Since it is not yet available in the Netherlands we discard this option for the moment though. Since modern PDAs support both GSM and WiMAX access, switching from the GSM network to WiMAX would not require much effort.

For a fully operational system a different choice could have been made. Since the C2000 system already provides security and guarantees a high availability it seems like a very good choice, however at the moment little information on for example data transfer speed could be found due to its confidential nature.

Adaptive techniques

Although some existing networks already could provide the required infrastructure, there are two techniques that are worth mentioning since they could improve reliability and availability of the system. Some existing networks already implement these techniques, others could support one or both of them.

Ad-hoc networks

Ad-hoc networks are usually based on wireless connections. Devices discover other nodes within range and establish connections to create a network. To find a target node out of range, a request can be sent over the network which is forwarded by other nodes. Routing protocols then provide stable connections automatically, even if nodes are moving around.

Cognitive radio

In today's wireless networks, fixed frequencies are assigned to companies or organizations under a license. For the commercially available frequencies in the Netherlands, these licenses have been auctioned for large sums in 2000 [50]. The frequencies can only be used by the proprietor, but a large portion of the assigned spectrum is only used sporadically. Geographical variations in the utilization of assigned spectrum ranges from 15% to 85% with a high variance in time. To improve on this inefficient frequency usage a new communication paradigm has been developed, known as cognitive radio [15]. This technique exploits unused frequencies which are officially reserved under license. It allows users to determine which part of the spectrum is available and detect the presence of licensed users when operating in a licensed band. Cognitive radio allows users to select the best available channel, coordinate access to this channel with other users and clear the channel when a licensed user is detected.

Although emergency use might have a high priority, taking advantage of other organizations' frequencies brings all kinds of legal issues. It is important that before use, this technique should be discussed with license owners and clear agreements should be made.

The AAF project

A nice example where these techniques are combined is the AAF project by the Dutch national research program Freeband Communication [51]. The AAF project is about data communications in emergency situations through cognitive radio. Although this project is aimed at large scale disasters, it is no surprise that it has a lot in common with incident management research. The AAF project intends to use an ad-hoc communication network based upon cognitive radio.

Data storage

There are a few possibilities that will now be described.

Local storage

Data can be stored locally at the incident site, for example on a handheld device like a PDA. Since this is where most information is produced storing locally simplifies the input and storage of data. However if control rooms require information, they would have to retrieve data from the device. Since mobile devices are generally limited in processing speed and storage capacity, this is probably not an option.

Centralized file servers

One of the most common means of storage in a network is on centralized file servers. By storing all data in one specialized system synchronization becomes a lot simpler. Since it only involves one dedicated system, storage capacity and processing speed become less of an issue. In terms of security and reliability it is more vulnerable, but this can be countered by means of firewalls and backup systems. Maintenance becomes easier as well because of the single system involved.

Distributed storage

In stead of storing information in one place, it is also possible to store it in multiple locations. In this way, storage resources are shared, reducing the need for large storage systems. The distribution of files also increases robustness of the system since if one storage device fails, only part of the data is lost. If this is combined with redundant storage robustness is increased even more. This has some disadvantages however, first of all synchronization becomes an even larger issue. If a file is updated, it has to be known to the rest of the network and a means to ensure clock synchronization becomes unavoidable well. It is also necessary to keep track of where specific data is stored throughout the network and keeping this data up-to-date. The distributed nature of the system could make maintenance cumbersome. While distributed storage has its advantages, the required overhead makes this option slightly less practical.

Whether data is stored locally, on centralized servers or in a distributed fashion, it should be decided how the required data is stored. This is not a trivial question, since it involves matters of security, consistency and synchronization, robustness and availability and it even involves maintainability and support (by the supplier). Some of the most relevant options for data storage are described below.

Database systems

Most data could be stored in a simple database system. Database systems allow simple access to data through queries. Database systems are less suitable to store image and video data, however solutions do exist in the form of multimedia databases [52]. It is also possible to store links to multimedia data in an ordinary database. A browser could simply request such a link by means of a query and let the operating system handle displaying the multimedia file. Databases allow for simple data storage and retrieval and easy selection of data based on search criteria. Some well known database standards are Oracle and MySQL.

Information repositories

An information repository is a form of data storage where data can be stored safely and in a centralized manner. Information repositories feature centralized management for all deployed data storage resources. They are self-contained, support diverse storage resources, and support resource management to add, maintain, recycle, terminate and keep track of media. To reduce required maintenance information repositories are automated and operate autonomously. They are generally flexible and easy to extend and offer built in redundancy.

Data warehouses

A data warehouse is a kind of repository used to store historical data. While operational systems are optimized for simplicity and speed of modification, the data warehouse is optimized for reporting and analysis. A data warehouse is not the right choice for active use and modification of data during incident management, however storing data in a data warehouse after clearing the incident site might support the evaluation of the incident response. The Dutch government is currently supporting work on the Nationaal Datawarehouse project at the moment, which might be perfectly suited for this purpose as well [53].

Appendix B

Interviews

In order to gain a better insight in the practical side of incident management, a few experts from the field were interviewed. This section provides a summarized overview of these interviews.

Interview Marianne Kuijpers-Linde and Ben Immers

My first interview was with Marianne Kuijpers-Linde and Ben Immers. Marianne is CEO of GeoDan (a Dutch company that develops, among others, Geographical Information Systems). At the time of the interview, she was working on a project on Incident Management in the Dutch province of Utrecht. Ben is my supervisor from TNO Bouw en Ondergrond and is Senior Research fellow at the division Mobiliteit en Logistiek. He is also part time professor at the Catholic University of Leuven, Belgium. His expertise is in the field of traffic and infrastructure.

The goal of the meeting was two sided, the main goal was for Marianne to interview Ben for his expertise. Ben asked me to join the interview since it could prove valuable to my thesis as well. The following is a summary of the points discussed in the interview.

According to the Ministry of Transport, Public Works and Water Management, about 20% of congestion is incident related. In Germany and the United States, estimates are as high as 50%. Whether this difference is a result of the way incidents are handled in the Netherlands is not clear at the moment, but improved incident response could obviously have a positive impact on congestion. Although obtaining data can be hard at times, in general Incident Management on the Dutch motorways works fairly well. When compared to the old procedures, the incident response is generally a lot faster. On the secondary road network, the incident management procedures have not been implemented yet, and here too a substantial difference in response time can be seen.

In order to improve Incident Management even further, many initiatives have been started to provide supporting IT systems. The system GeoDan is developing and the IMICS system proposed in my thesis are two of them. As another example, the salvage companies have tracking and tracing software that keeps track of their equipment. Deploying a new IT system for incident management in the entire Netherlands at once could be complicated. It may therefore be preferable to deploy a new system in phases, one region at a time. Ben notes however, that ideally all safety regions in the Netherlands should follow the same approach. Using the same procedures across the entire Netherlands prevents conflicts and other problems.

According to Ben, one of the most important stakeholders in Incident Management is the traffic control centre. The traffic control centres' main tasks are traffic- and incident management. Actually they are the only stakeholders for whom incident management is core business. Yet, they are not part of the coordinated regional incident response (Gecoördineerde Regionale Incidentbestrijdings Procedure, or shortly GRIP in Dutch) and are generally not seen as one of the emergency services. They also do not have access to the C2000 system.

Although things have improved the last years, the training of traffic control centre personnel could be improved (as compared to air traffic control, which is even more critical). Information overload is an important issue at the traffic control centres and often when an incident happens, personnel is staring at the screen for the first minutes, while

especially at these moments, they should respond quickly. One of the future goals of incident management is to integrate incident management and traffic management. Updates of an incident should be directly sent to the traffic control centres so they can respond adequately.

Incident management constitutes both intra- and interagency procedures. These can both lead to problems as the following examples illustrate.

One of the problems with incident management is the bureaucratic nature of some of its stakeholders. For example, the Dutch police force has many departments, each with different responsibilities. Although police are very motivated to improve incident management, it can be hard to deal with conflicting interests between departments.

Between agencies, conflicts can arise about authority and responsibilities. For example, securing the incident scene could be taken care of by the inspector from the Department of Public Works, so the police can start gathering evidence if required. The authority of the road inspector, however, is not always recognized by the police. The road users sometimes do not know what authority the road inspector has.

Another example is the fact that multiple reports can be received of the same incident. In practice it happens that these are treated as different incidents either within an organization, or between organizations.

The interview concluded with a discussion of the strengths and weaknesses of the current incident management procedures in the Netherlands.

Strengths:

- Incident management is well organised in the Netherlands on a strategic, tactical and operational level.
- Technologically, there is a strong position (there are many projects and initiatives)
- Cooperation between stakeholders is very good (there is one directive for all emergency services)
- The incident management procedures have been adapted to each other
- Certified emergency respondents are cross trained so they have adequate knowledge of other organisations' tasks.
- Most procedures have been adapted nationwide

Weaknesses:

- Better communication to the public is required
- Although much energy is put into improving incident management, feedback and evaluation currently lag behind
- The information chain is incomplete. (With the new shared control room system, this is greatly improved)
- Not everyone is fully aware of the current agreements (especially on a regional level)
- Funding for improvements is limited

- Although all stakeholders are willing to improve incident management, they are less dedicated in the Netherlands than for example in the United States.
- Currently, all effort for improvement is aimed at the organisations themselves, not the road users.

Interview Eelco Kaper

My second interview was with Eelco Kaper. Eelco is a consultant at Imtech currently working for the Department of Public Works. His expertise is mainly in the field of information and communication.

In the interview, Eelco gave a short overview of some the solutions currently used and projects currently being developed at the Department of Public Works. A large part of the interview focused on a strategic level, what are the bottlenecks with a system like IMICS and how can they be dealt with. According to Eelco, the most important issue with a project covering multiple agencies is system deployment.

The first step for this project would be to analyse the current situation and to create a system design. One of the most important issues with incident management is the fact that people are in contact with each other all the time, but little direct data is available. Most stakeholders agree that availability of real-time data is currently the most important issue in incident management. Therefore the main focus of the design must be to provide up-to-date and correct information. When dealing with a large group of primary stakeholders however, it is important to realise what information can and cannot be shared. It is critical to analyse the responsibilities and information needs of the different stakeholders and to determine who has access to what information.

Once the design has been completed, a deployment strategy must be developed. This strategy should keep in mind the development of related systems. Should the design rely on other developments, they should be ready when the system is deployed. The envisioned system aims to support chain integration for decentralised governmental organisations. This is closely related to the field of change management. As the project is a proof of concept, deployment is beyond the scope of the project and will not be analysed in detail, but it is important to keep this in mind during the development of the system.

Eelco notes that it is also very important to keep an eye on the added value of the system for the partners, and also for the incident victims. IMICS must at least provide an advantage compared to the existing situation. The system could for example provide better security at the incident scene and improved information availability.

The interview concluded with a discussion of possible points that could be improved by IMICS:

- Direct availability of correct information
- Keep track of tasks and assignment of personnel
- Provide accurate estimated time of arrival, keeping traffic conditions in mind
- Clarify if an investigation for evidence is required

Interview Eeltje Hoekstra

While the former two interviews took place in the early phases of the project, the interview with Eeltje Hoekstra took place to discuss the basic design after it had been worked out. Eeltje is Program Manager Incident Management at the Dutch national traffic control centre (Verkeerscentrale Nederland or VCNL in Dutch) for the Ministry of Transport, Public Works and Water Management. While the previous interviews were more of an explorative discussion, for this interview a list of specific questions was prepared. For this reason, the way this interview is discussed is structured differently. The questions were divided into 4 subjects, which the discussion of the interview will follow.

1: Stakeholders and tasks. First I had some questions concerning the tasks of a few of the stakeholders.

a) Should the regional officer hazardous substances (ROGS) and consultant hazardous substances (AGS) be at the incident location or can they assist by phone? Should respondents wait for their arrival?

The assistance of ROGS and AGS is not always required. They are only required when hazardous materials are involved. In this case, they are often at the incident scene, but this is not mandatory. At the moment the emergency services are looking to define separate procedures for incidents involving hazardous materials.

b) What is the task of the lorry salvage consultant and when is he required? Should respondents wait for his advice?

The lorry salvage consultant is an insurance expert who can estimate the costs and value of equipment and gives advice on the best way to recover lorries and cargo. The IM lorry recovery dispatch centre (CMV) estimates whether the lorry salvage consultant is required. This is discussed by phone.

c) What is the role and composition of the Copi team?

In small scale incidents, the different emergency services discuss and coordinate their tasks at the scene in an informal way, called the 'motorkapoverleg' in Dutch. If the incident is complicated, it can be decided to scale up the incident response. The coordination at the scene is then structured in a more formal group called the coordination team incident location (Coördinatie team Plaats Incident or shortly Copi in Dutch). Copi is composed of members of the police, fire brigade and ambulance service. The inspector from the Department of Public Works has an advising role and is not an official member of Copi.

2 Incident management in general. In this part of the interview, some general questions about incident management in the Netherlands were asked.

a) What projects are currently under development in the field of incident management?

There are many projects in different stages of development in incident management at the moment. An example fairly similar to the proposed IMICS system is the Man In the Middle system, in which the emergency services could post e-mail messages to a central system to which all stakeholders had access. Although its functionality was limited, it was received very positively.

At the national traffic control centre, a system was tested that used PDAs with a camera and gps to take pictures with gps coordinates attached. The location could immediately be viewed through google maps and the pictures could be accessed through a special site. These pictures usually improve situational awareness. It must be noted however, that finding enough gps satellites to get a reliable location often took too long. Taking pictures was often forgotten because it was not part of the normal job and other tasks had a higher priority.

b) What are the bottlenecks in current incident management according to Eeltje

Eeltje pointed out two major concerns. First of all, communication and registration of information takes time. Currently, there is too little registration and there is a high chance for errors. In a busy situation, quick notes are made on a notepad, which easily leads to mistakes. A possible solution could be to provide an IT system in which clear options can be selected, but ambiguity is still an issue here. Do the options a user selects really represent the information he tries to convey and do other users interpret this in the same way? Secondly, organizations are reluctant to couple information systems or share information because of privacy issues. It must be clear what information can be shared and what information can not be shared.

3 IMICS. I explained the proposed global design of the system and had some specific questions. Eeltje also had some ideas and remarks that have been processed in the following part.

a) What do you think of the project in general?

The general idea is very good. The task part is a good idea, but will be troublesome in practice since people tend to focus on their primary tasks. Entering data would be a task typical for the control room, since this coincides with their current tasks. At the incident site, the involved people currently only call to the control room incidentally. The director role would also be best for the control room.

In practice, people are very occupied with their direct tasks and the system could help to make them more aware of their surroundings. A better understanding of

the complete situation would improve safety, but would also have a positive impact on incident management overall. Many people in the field have suggested that a simple photo of an incident could greatly improve understanding of the incident situation.

b) What are the system's strengths and weaknesses?

The biggest risk is privacy. The different organizations are reluctant to couple their networks and security is a critical component. The biggest strength is the fact that the intended users from all organisations are very enthusiastic about these kind of developments as they can clearly see their advantage (as for example in the Man In the Middle system). There is a large incentive to do things better and more efficient.

c) Is it advisable to include unverified information into the system? Should the system indicate whether something has been verified?

In practice, verification hardly takes place as it takes too much time. Because many people are usually working in a group effort, mistakes are generally corrected quickly. Therefore, information does not have to be verified before it is entered into the system. This way of working does require trust in each other though.

d) If the proposed task part of the system is used, the fact that the order of tasks is not always clear in advance could prove to be problematic. How do the emergency services respond to unexpected situations and how could the system cope with them?

In practice many unexpected situations occur (it may for example turn out after an hour that gathering evidence is required because the injuries of a victim turn out to be more serious). Although providing as much information as possible could reduce the occurrence of unexpected situations, this can never be completely prevented, so the procedures and the IMICS system must allow for certain flexibility.

e) In practice it can happen that one incident is known as multiple distinct incidents by the emergency services. Do you have suggestions how to prevent this using IMICS?

Currently the amount of reports received on an incident range from 0 to 60. Sharing information is the first way to prevent these reports being interpreted as different incidents. Incidents should be identified by their time of occurrence and location, but the decision whether it is one incident or multiple incidents at the same location should be made by a human. This could probably best be done at the control room. Since not every piece of the Dutch motorways is monitored by cameras, this could be tricky.

Eeltje also suggested I should talk to Peter Grinwis at the traffic control centre in Rhoon, as he has hands on experience with many aspects of incident management. He would probably have a good idea who should take the director role and who should enter data into the system.

Interview Peter Grinwis

Following Eeltje Hoekstra's suggestion, I had an interview with Peter Grinwis from the regional traffic control centre in the Dutch town of Rhoon. He is a coordinating road traffic controller for the Ministry of Transport, Public Works and Water Management, and as such is experienced in the incident management practice.

Peter showed me the traffic control centre and the systems they use. The traffic control centre is responsible for managing and monitoring the critical infrastructure of a region. They receive video images from the most important motorways in the region and automated systems highlight video streams with unusual traffic behaviour. If necessary, the traffic controllers are able to adapt the digital traffic signs of the entire region from this centre and the system can calculate the probable consequences of traffic measures for the rest of the road network. Because of the critical nature of the infrastructure, the entire centre runs on a highly secured, self contained military network.

After this interesting introduction, I interviewed Peter on a few subjects. He openly answered my questions and we discussed points where the current system could be improved. The results of this interview are described below.

1 Communication at the control rooms. My first questions were about the practice of the shared control rooms, as a good understanding of their tasks is critical for the deployment of IMICS.

a) Have the control rooms been completely integrated or do the emergency services still use their separate control rooms?

Currently all 25 safety regions have been integrated and use the shared control room system (Gemeenschappelijk Meldkamer Systeem in Dutch, or shortly GMS). They communicate using the C2000 system and data is stored using GMS.

b) Is it possible that multiple control rooms are involved in the same incident?

When a call is made to the 112 emergency number, the emergency control centre asks which of the emergency services is required and forwards the call to the regional shared control room. Here all required units are sent to the incident location. If an incident is at the border of two safety regions, it is possible that two control rooms are involved, but one (usually the one of the first unit to arrive at the scene) is in charge. And the other will only assist if requested by the other control room.

c) Would the control room be a good location to enter data into the IMICS system? Do they have time for this and is this a logical location for inputting data?

In the current situation, the control room writes down the information gained from the questioning procedure. Inputting data into the IMICS system would also best take place here. Ideally, the system should follow this procedure.

d) The new shared control room system (GMS) contains a large part of the functionality. Has it been completely deployed? Does the traffic control centre have access to this system?

The GMS system has been completely deployed. The traffic control centres do not have access to this system.

e) What kind of systems do the control centres have in use?

They have systems that monitor and log the incident location, incident type and any specifics. They use the infoweb system. The most important issues with the system in use at the traffic control centres are the fact that the system is too large and complex, contains too much information and users cannot share their view of the system. They would rather have a system like GMS that is concise information that is easy to communicate to others.

f) Do these systems contain a standard model of the incident situation (an ontology)?

Yes.

g) Is it currently possible to access these systems at the incident scene?

Not really, but the people at the scene (at least from the Department of Public Works) would not have time for this anyway. The police can read incident reports in their cars. The inspectors from the Department of Public Works would like to be able to see these reports as well before they are dispatched and accept. In this way they can refuse if they happen to have encountered an incident themselves. They would also like to be able to notify that they have arrived by a simple system.

h) How many people are available at a control room?

At least 3

i) How many people are usually working on one incident?

This varies greatly, depending on the type of incident

j) What is the procedure for incidents on the border of two safety regions? Do these incidents lead to conflicts of authority?

It is possible that assisting vehicles come from two sides. The control room who is at the scene first is often the one in charge of handling the incident. At the scene they usually consult with each other. There are no arguments at the scene. It does happen that in retrospect it would have been easier if the other control room would have been in control.

k) How are units dispatched to the incident scene?

At the control room someone is notified or is already looking along with the procedure. He then calls the required persons (through C2000 for the police, ambulance service and fire department). If an incident is very large, every organization sends an officer on duty to the scene. Their task is mainly to coordinate the relief effort in the Copi team.

l) Could this be replaced by a simple IT system that also contains the incident information? (For example by clicking a button to request a police unit)

It could be, but the down side is you never know if the available information is enough. Experience helps, but in a phone call one can discuss the situation and immediately hand on new information. So this could be done digitally, but phone calls can definitely help at times.

m) How do the emergency services keep in touch with each other?

Through phone calls, C2000 and the control room.

n) Who is responsible for forwarding the information of the incidents and coordinating the response, the emergency response centre, or the local control room?

The local shared control room.

o) Would the shared control room or the traffic control room be suitable for the director role as proposed for the IMICS system? Who would be most suitable to take the director role?

The control room should have the director role, but especially when the operator is inexperienced, the officers on duty (usually the most experienced officers at the incident scene) should keep an eye on the situation.

p) Is there a shared control room for each safety region?

Yes

q) Is there only one emergency response centre for the entire Netherlands?

Yes, it is located in the town of Driebergen. The emergency response centre forwards the received calls to the local shared control rooms.

r) Who is responsible for entering information into the IT systems?

This is mainly at the shared control room, in practice they are the ones that enter the most data into the system. It is possible that people enter data into the IMICS system at the incident scene, but this would probably mainly be requests for backup or traffic measures, since they are often occupied with their primary tasks.

s) What is the added value of the shared control room system in your opinion?

The fact that information is shared is a great advantage, especially at the start of the incident response. This allows the people involved to build a concrete image of the situation for themselves.

2 Traffic control centre: Next I asked a few questions specific for the traffic control centre.

a) Is it possible that the traffic control centre is the first to notice an incident?

Yes

b) What are the procedures in such a situation?

They call the control room. They don't do the questioning procedure, but follow a fixed protocol to hand on information. They also put traffic measures in place if required. (In areas without cameras they receive requests for traffic measures, but for other areas they determine if traffic measures are required themselves)

c) How is most information received by the traffic control centre?

Information is received by camera, phone (by the emergency services) or email (by the VHD if a towing vehicle is required). Sometimes they are called by an insurance company that a vehicle has broken down at a certain location.

d) How is this information stored and processed?

Their IT system stores everything in logs.

e) How do the traffic control centres communicate with other organisations?

Mainly by phone.

f) Is storing information in the system of high priority?

It can wait when it is very busy, normal work has higher priority. The current system at the traffic control centre is not of great help. The GMS system is much more suitable.

3 Procedures: The following questions are about the incident management procedures in general.

a) How are the emergency services that are initially required determined?

This is based mainly on experience. And there are standard procedures for certain situations. (For example, in case of a trapped driver, the fire brigade is required with special equipment).

b) Who decides if extra assistance is required?

When an agent in the field feels more assistance is required, the control room is notified without further discussion. It is better to have too many persons at the scene than too few.

c) Who is responsible for contacting the extra assistance?

This is usually someone from the organization from which more assistance is required. This is also done by the first to arrive at a scene when the incident turns out to be more complicated than expected.

d) Who is in charge of the incident response?

Usually the police are in charge, but if the fire brigade is required, they are in charge of the incident scene. However, they always consult with each other.

e) Who do the people in the field have contact with except with the people at the incident scene?

Everyone calls their own control room.

f) Do field officers only call to their own control room?

Yes, this is a standard procedure, since otherwise the control room would lose oversight.

g) Do the organizations exchange data between each other?

The organisations only share the necessary information. Sensitive information is not shared.

h) Who determines for example if a car has to be towed away?

This depends on the situation. If for example evidence has to be gathered, the police have to release the scene before a vehicle can be towed away. Otherwise there is not one specific organization responsible for this.

4 Bottlenecks in incident management: The next questions are meant to give me a better understanding of the kind of problems that occur in incident management.

a) Is information sometimes communicated incorrectly?

Yes

b) Is it possible that information is not communicated to the people that need it?

Yes, for example at the border of 2 safety regions it could happen that information is communicated to the wrong safety region, but this does not happen often.

c) Does it happen that information is not communicated fast enough?

It does sometimes happen.

d) Is it possible that people can not or will not communicate? For example because they are too busy or because of conflicts of interest?

When people are busy they can be snappy, but everyone knows the others can be very busy and this does not lead to problems in practice.

e) Is the communication equipment always functioning correctly?

Sometimes the network is overloaded, for this reason they have the C2000 system and generators available.

f) Does the current communication equipment have disadvantages?

At the traffic control centre, it is mainly 1 direction traffic, it is not possible to share information to all at once. C2000 does provide this functionality, but the traffic control centre is not allowed to use it. The system at the traffic control centre is also very slow.

g) Are there any clear points of improvement for the communication equipment?

The speed of the system could be greatly improved. Also the system should follow the questioning procedure and should be developed in cooperation with the end users in stead of being enforced by the nationwide organisation.

h) Is everyone sufficiently familiar with the procedures?

When changes are made to the procedures it is possible that not everyone is aware of it. This does not lead to problems in practice.

i) Do the procedures lead to conflicts in practice? (For example when people have to wait because of hazardous materials while they are trying to save a life)

Personal safety has first priority, so this does not really lead to problems. Gathering evidence can sometimes be in conflict with other tasks, but this can usually be done quickly so it does not lead to real problems.

j) Do the procedures guarantee the safety of the emergency services and road users sufficiently?

Yes. This is discussed on a regular basis in the incident management board and suggested improvements are adopted if required.

k) Are the emergency services always at the incident scene within the time limit laid out in the procedures?

They rarely arrive too late. If this does happen, it is usually because of a flawed report.

4 Man In the Middle system: Concluding the interview I asked Peter a few questions about a test project slightly similar to the IMICS system, called the Man In the Middle system.

a) Could you describe the global design of the Man In the Middle system?

The Man In the Middle system consisted of a central server to which the emergency services could send basic email messages about facts from incidents. Other stakeholders involved in the incident could access these messages and respond to them. Although its functionality was very limited, all involved users were very enthusiastic about it.

b) What were the results of this test?

The main advantage was the fact that users could check up on each other's work. For example the police force made mistake when entering the incident location.

The traffic control centre discovered with the help of their cameras that the actual location was 200 meter further. These kinds of corrections can be important, say for example when the location is near an exit road, it could mean a detour is required.