Adaptive Workflow Simulation of Emergency Response

Guido Bruinsma

Doctoral Committee:

Chair:	Prof. dr. J.M. Pieters
Promotor:	Prof. dr. R. de Hoog
Members:	Prof. dr. W.R. van Joolingen
	Prof. dr. J. van Hillegersberg
	Prof. dr. M. Neerincx
	Prof. dr. B. Ale
	Prof. dr. I. Helsloot
	Dr. M. Sierhuis

This research was funded by the Dutch Ministry of Economical Affairs (SenterNovum) under contract to the Human-Computer Studies Laboratory of the University of Amsterdam as part of the TAID project (project number MMI04006).

This research was carried out in the context of the Netherlands School of Communications Research.

ISBN: 978-90-365-2999-0
© 2010, Guido Bruinsma

Cover Design: Heidi Ulrich (www.heidiulrich.nl)
Druk: Gildeprint, Enschede

ADAPTIVE WORKFLOW SIMULATION OF EMERGENCY RESPONSE

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
prof. dr. H. Brinksma,
volgens besluit van het College voor Promoties
in het openbaar te verdedigen
op vrijdag 26 maart 2010 om 16:45 uur

door

Guido Wybe Jan Bruinsma
geboren op 15 februari 1978
te Roosendaal

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. R. de Hoog

Dankwoord

Ondanks dat alleen mijn naam de kaft van dit proefschrift siert heeft dit proefschrift niet kunnen ontstaan zonder de hulp en inspiratie van velen. Enkele zou ik in het bijzonder willen danken.

Ten eerste zou ik graag Robert de Hoog, mijn promotor, willen bedanken voor zijn begeleiding en steun gedurende de gehele promotieperiode. Je bent in staat geweest een omgeving te creëren waarin ik me zeer op mijn gemak voelde. Ik heb genoten van onze bijeenkomsten waarin de inhoud van het proefschrift niet per definitie centraal hoefde te staan, maar die altijd nieuwe inzichten opleverden.

Speciale dank gaat verder uit naar de personen die de dagelijkse werkzaamheden bij de Universiteit Twente kleur hebben gegeven. Elian, Jurjen, Marleen, Casper en Wilco, ik heb genoten van onze bijeenkomsten waarin de inhoud van het proefschrift zeker niet centraal stond, maar die wel vaak nieuwe inzichten opleverden. Ik wens jullie veel plezier met het lezen van mijn proefschrift.

Ik wil ook Maarten van Someren en Niels Netten danken voor de fijne samenwerking binnen het TAID project.

Er dient verder opgemerkt te worden dat zonder de onbaatzuchtige hulp van Henk Sligman (veiligheidsregio Twente), Vincent van Vliet (Nederlands Instituut voor Fysieke Veiligheid), Ron van Hoof en Maarten Sierhuis (NASA) dit proefschrift en het TAID project nooit had kunnen worden wat het geworden is. Dank daarvoor.

Mijn zus en nu tevens paranimf Monique Bruinsma die mij tijdens de donkere dagen voor kerst heeft geassisteerd mag ook niet in deze lijst met eervolle vermeldingen ontbreken.

De laatste vermelding in dit dankwoord wil ik schenken aan diegenen die thuis altijd hebben doen aanvoelen als thuis en mij de gelegenheid en steun hebben gegeven mijn proefschrift af te kunnen schrijven. Caroline, Sofie en Wybe, ik zal jullie onvoorwaardelijke steun en opoffering nooit vergeten!

Guido Bruinsma

Table of contents

1. Introduction

For emergency response to effectively deal with the dynamics incorporated in an emergency situation, it has to be as flexible as the emergency itself, demanding innovation and creativity that is unique for that particular situation. During emergency response, a new mitigating organisation emerges consisting of multiple monodisciplinary organisations, such as the fire department, the police department and the medical services that have to complement each others' activities, while working multidisciplinary in a temporarily bound high risk situation. Within this organisation, the sub-organisations are coordinating their activities through mutual adjustment and liaisons to most effectively use the decentralized expertise of the organisations involved in the emergency response. The adhocracy (Mintzberg, 1983), as such an organisation can be classified, is able to dynamically reorganize its own structure and workflow, and, by doing so is able to shift responsibilities and adapt to the changing environment (Van Aart, Wielinga, & Schreiber, 2004). The organisations' immediate purposes during the emergency thus shape the organisation.

The adhocracy (Mintzberg, 1983) is a widely used and empirically well suited structure to coordinate complex and ill-structured team performances, such as emergency response. It provides the multidisciplinary organisation with the means to deal with the extraordinary. The downside of the adhocracy however, resides in the relatively large proportion of time spent on communication in order to coordinate activities (workflow planning), keep track of the current state of the emergency (situational awareness) and to keep track of the current state of the organisation (organisation awareness).

As the complexity of the emergency, and consequently the complexity of the emergency response organisation, increases, information management will play a dominant role during the response. In order to construct a common operational picture of the emergency situation and prevent both information overload (receiving too much information, hampering information processing) and information deprivation (being deprived of elsewhere available, relevant information) well working information management is key.

In addition to increased communication during the response, the second downside of the adhocracy structure that Mintzberg presented concerns the danger of an unbalanced distribution of workload within the organisation. With an increase of the number of emergency responders involved, the complexity of the situation and the high degree of specialized tasks that are performed by the numerous emergency response organisations, some organisations may experience a high workload, while other organisations that could come to the assistance have no workload or are standing by, awaiting to start new activities. The workload can become skewed, not using the full potential of the organisation.

To reduce the quantity of information exchanged during emergency response, semi static information is being inventoried beforehand in the preparation phase. Semi static information includes information about objects, (multidisciplinary) work protocols, inhabitant information, geographical information and organisational (start) capacity (ACIR, 2005). Inventorying and making these information sources accessible for the emergency responders and anchoring their use within the emergency response organisations, decreases the total communication needed during the actual response and builds a common operational starting picture.

However, as stated earlier, the adhocracy-like organisation finds its challenge in adapting to the actual and future situation. Decentralized communication about the current state of the emergency and the measures that are taken by the emergency responders to deal with the situation, withholds precious information that is essential for correct planning of future actions by the response organisation. Given how fast information can become outdated, information management during the actual response thus is critical in order to continuously have the most recent information at one's disposal (for a common operational picture) and effectively plan work. The multidisciplinary and specialist character of the work the emergency responders are involved in however, does not foster interdisciplinary information exchanges. Since information relevance is hard to assess for all specialists in the field, this can lead to situations where all information is exchanged (just in case) or situations where relevant information is held back caused by the inability to assess the relevance of all information for all actors. The first situation results in an increased communication load for the receiver and the communication network, while the latter leads to information deprivation. Both consequently lead to sub optimal decision making, planning and a distorted view of the situation at hand.

In practice, these information exchange challenges are taken on by engaging in intense multidisciplinary training exercises, focusing on both familiarity with possible emergencies that can occur and with the multidisciplinary organisations' members activities and resources (Boin, Kofman-Bos, & Overdijk, 2004). Despite these efforts, recent incidents and training exercises in and outside the Netherlands (Table 1.1) persistently show that not having or not sharing information about the dynamic aspects during emergency response still are the major source of emergency response inefficiency and error, and affect incident outcome through workflow planning that is based on this information (ACIR, 2005; Bruinsma, 2005).

Table 1.1: *Conclusions regarding information exchange and information sharing during emergency response (exercises)*

Incident	Year	Conclusions / quotes
9/11 terrorist attack New York	2001	- "Command and control were affected by the lack of knowledge of what was happening." - "The 911 system remained plagued by the operators' lack of awareness of what was occurring." - "Incident commanders from responding agencies lacked knowledge of what other agencies, and, in some cases, their own responders were doing." *Source: National commission on terrorist attacks upon the United States (2004), The 9/11 commission report*
Fireworks depot explosion Enschede the Netherlands	2000	- "Crisis management is first and foremost information management. If the parties involved are to be able to act effectively to combat a disaster, it is essential that they have the information that they require, when they require it, so that the necessary decisions can be made. Before anything else, this imposes demands on the technical infrastructure." - "A good information position is literally a matter of life and death for the fire brigade." *Source: Commissie onderzoek vuurwerkramp (2001). De vuurwerkramp:*

		eindrapport [The fireworks disaster: final report].
Bonfire, national exercise, the Netherlands	2005	- "The most important thing to note here is that coordination, internal provision of information and crisis communication […] were mainly in the hands of the decision-makers themselves. This put so much pressure on them that they were rarely able to make strategic decisions for the medium or long term." - "Support staff had been expected to ease pressure on the leadership by preparing their meetings and working out the results. This did not go as planned, however. Since they did not always have access to the latest information, they could not provide optimum support. As a result, they were by-passed, so that they had even less access to information. The vicious circle was thus complete." *Source: Institute for Safety Security and Crisis Management (COT) (2005). Bordering on reality: Report of findings on Bonfire crisis management simulation.*
Voyager, national exercise, the Netherlands	2008	- Information exchange and information sharing between the different emergency response organisations and levels of government was inadequate and untimely. […] these organisations and levels can only function satisfactory when they have a minimal amount of information at their disposal. During the voyager exercise not all levels of government or emergency response organisation had this minimal amount of information available to them at the same time. *Source: Capgemini, TNO, Berenschot (2008). Evaluatie oefening Voyager: Een systeemevaluatie van kritische processen bij crisisbeheersing [Evaluation Voyager exercise: A system evaluation of critical processes within crisis management].*

Quoting the Commissie onderzoek vuurwerkramp (2001), crisis management is first and foremost information management. Information determines which activities are started and the adaptation of the disaster response organisation and the organisation's flexibility (Corbacioglu & Kapucu, 2006, Hatum & Pettigrew, 2005). Proper information management furthermore provides the necessary boundary conditions for post disaster evaluation and organisational learning (International Federation of Red Cross and Red Crescent Societies, 2005) being essential for deriving lessons learned from emergency response situations.

1.1 Task Adaptive Information Distribution

The information challenges, and its effect on emergency response workflow, communication load and consequently the workload of the emergency responders involved, are the central themes of the so-called Task Adaptive Information Distribution project of which the research presented in this dissertation is a sub project.

The emergency response organisation experiences many challenges due to the domain it is operating in and the decentralized specialist work it is engaged in. The Task Adaptive Information Distribution (TAID) project approaches the information dissemination issues by developing a tool that selects and distributes information and uses extra information about tasks, the workflow and the cognitive state of the emergency responders to optimize this distribution. It furthermore gives insight into the workflow, workload, communication load, and current cognitive state of the emergency responders. Information distribution is tailored based on the current activity of the emergency responder and his current cognitive state, preventing information overload, information deprivation, speeding up information flow between the multidisciplinary organisations and finally positively influence emergency response duration.

This section will present an overview of the proposed TAID system and its two subsystems.

Within the TAID system, two subsystems reside (Figure 1.1) that are developed in two separate PhD projects. The first project aims at the development of a trainable information distribution tool. Complementary to this project, the second project aims at the development of a tool that is able to simulate emergency response workflow and cognitive aspects of emergency responders that influence their response, providing extra information to tailor information distribution.

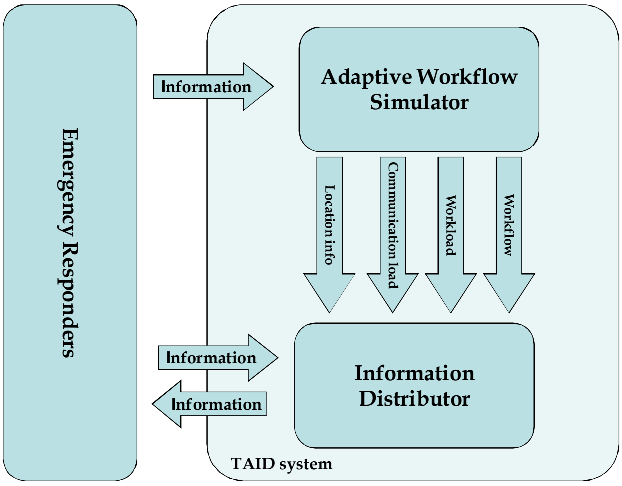


Figure 1.1: *TAID system overview*

Using knowledge derived from artificial intelligence, the first project, the *Information Distributor* (ID), is able to extract meaningful dialog segments during emergency response and assesses the relevance of these dialog segments for specific emergency responders. The ID distributes these relevant segments to the emergency responders that normally would have been deprived from this information, tackling the information deprivation problem within emergency response.

The ID improves information distribution by using additional information about the current task of the responder, location of the responder, and the present cognitive state of the responder as provided by an adaptive simulation of the emergency response. Using this additional information, it can further tailor information distribution, preventing cognitive overload brought on by information dissemination.

The additional information for the ID is provided by the *Adaptive Workflow Simulator* (AWS), which is developed in the second PhD project. The development and testing of the AWS is presented in this dissertation.

Based on the information exchanged by the responders (the same information that is used by the ID), the AWS generates an on-the-fly simulation of the work that is done. The derived workflow (who is doing what), in combination with task attributes (such as task load) and personal attributes (such as work experience) form the ingredients for the AWS to approximate the current and available cognitive capacity of the individual emergency responders. The AWS provides this additional information to the ID to tailor information distribution for the individual responder and tackles the uneven workload distribution problem by providing information about the current available capacity of the emergency responders.

The ID and the AWS are highly interconnected. The AWS provides information to the ID about the workflow, workload, communication load, location information, and the available cognitive state of the emergency responders. This information is used by the ID to tailor information distribution, providing relevant information to the right responder at the right time. The fact that the responder has this information, can trigger new tasks and activities, which in turn adapts the workflow, task parameters and cognitive capacity of the performer. This information is then again used by the ID, completing the circle, thus making the TAID system task and situational adaptive.

Section 1.1.1 and 1.1.2 will briefly discuss the ID and the AWS in more detail.

1.1.1 Information Distributor

Adaptive information distribution is the process of determining relevance of information in order to adaptively distribute this information to the emergency responders for whom it is relevant. Additional domain knowledge about activities is used to improve determination of information relevance (Netten, Bruinsma, van Someren, & de Hoog, 2006). The information used as input by the information distributor is indicated by the arrows with number 1 in Figure 1.2. These refer to the output information of the AWS that contains information about the workflow, workload, communication load and location of the emergency responders. The second input variable contains information that is exchanged by the emergency responders themselves. This contains fuzzy information, since it is exchanged in natural language. Pre-processing this information, indicated by number 2 in Figure 1.2, allows the information distributor to extract features within the text that will improve the selection process.

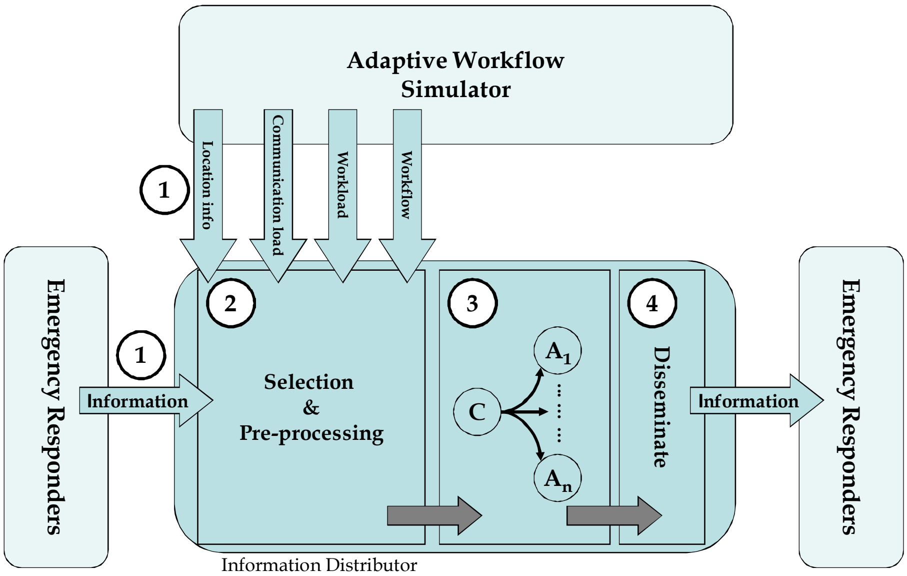


Figure 1.2: *Information Distributor overview*

To increase effectiveness of the predictions made by a classifier, the words (i.e. features) that are irrelevant to the prediction should be removed. A common approach, which was adopted, is removing the stop words. Stop words or also sometimes called function words (for example "the" and "a") have an important role in grammar, but carry little meaning, and therefore do not contribute much to the classification task. The texts we use for classification are different from the standard texts. In the domain of emergency response and management, most of the information is distributed by means of speech. Transcriptions of these speech utterances are different from, for example, news articles. Context features of the actor and the situation are necessary to obtain effective classification results.

In crisis situations there is a plan of attack (i.e., a workflow). Tasks are assigned to actors, for example through the adaptive workflow system. Task descriptions contain information about which tasks are relevant for the actor at a particular moment. When the plan of attack is adapted, we can keep track of changes in the actor's information needs by tracking the actor's tasks. In other words, when an actor changes from one task to another then also his information needs changes, which is represented by means of the new task description.

Assessing the relevance of new information requires some degree of understanding of the meaning of the information. A growing body of research in the Artificial Intelligence (AI) community addresses the problem of learning to classify text documents and of detecting topics of documents (Sabastiani, 2002). A standard machine learning approach to learn which information is relevant for which actor in a particular situation, is to use text classification. Text Classification is the task of automatically assigning semantic categories to natural language text. In our case we assign actor roles as labels to the training example messages.

The communication flows of messages communicated between collaborating emergency actors are often relevant for multiple actors which have different roles. Therefore, the learning task of our system is a multi label classification problem, that is, a training example can have multiple labels (for example, roles) assigned to it. In this case the classifier has to learn multiple (for example, overlapping) target classes. In Figure 1.2 it is indicated by the class labels (A1, A2...AN), which can be coupled to the same message (indicated by number 3 in Figure 1.2). This determines relevance of information to specific emergency responders.

When the information is assigned to one or more emergency responders, it finally is disseminated (number 4 in Figure 1.2) to them.

1.1.2 Adaptive Workflow Simulator

This dissertation will describe and test the Adaptive Workflow Simulator that is used within the TAID system to adaptively provide the Information Distributor with information about the current task, workload, communication load and location of the emergency responders involved in the emergency response and agent specific attributes about the current cognitive state of the agent.

In Figure 1.3, the layout of the Adaptive Workflow System is shown. It is based on an emergency response template (1), that consists of the static elements within emergency response that are constant or have a low dynamic character (ACIR, 2006) between and within emergency situations. This template consists, for example, of organisational assets (human resources and objects), standard activities and roles within the involved organisations.

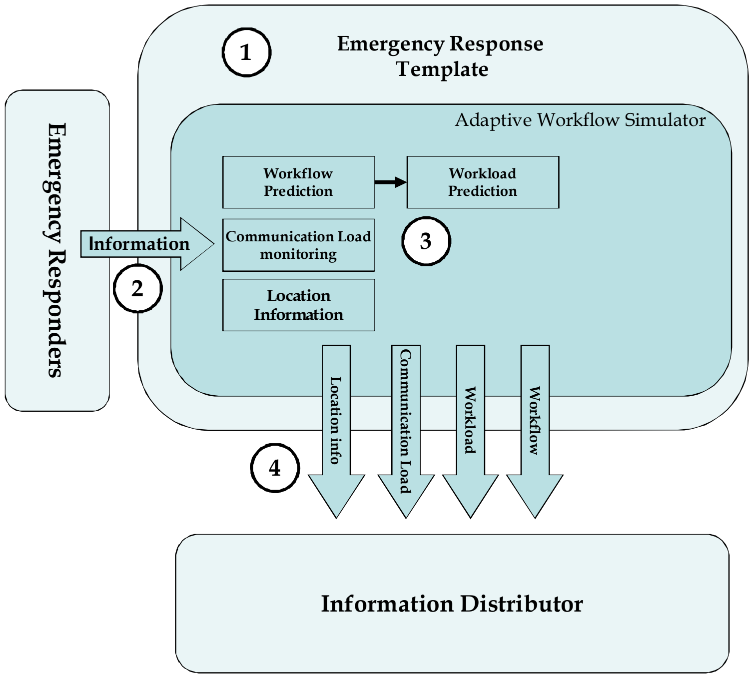


Figure 1.3: *Adaptive Workflow Simulator*

Information extracted from the communication between emergency responders is fed into the AWS (2) and is used as the enabling condition that triggers certain activities of agents within the AWS. From the communication, information about the location and communication load – as experienced by the agent -, can be deduced. The order in which information enters the AWS, furthermore, determines if, and when, enabling conditions for activities are satisfied, building an agent specific estimation of the workflow for all agents within the simulation (3). For an emergency responder to, for example, go to an incident the responder first has to get information that there is an incident combined with a possible location. The moment at which the responder receives this information, determines the timing of the action. When the information is received, a prediction of the current task is made. In the example, the current task is "move to accident location". Based on the role the agent has, further tasks can be specified, and hence predicted, fine tuned by new information input. The communication input and status reports are further used to determine the communication load, location status and information about the workload of the agents at a certain moment in time. This information is then sent to the Information Distributor to improve the text classification process (4). All agent specific workflows combined result in the total emergency response workflow.

1.2 General Research Questions

Given the challenges within the field of emergency response mentioned in the previous sections and the possible solution brought forward by the TAID system, the general research question within the TAID project is if having information available about workflow, workload and communication load of the emergency responders, information can be distributed more efficient, preventing information overload and information deprivation and hence positively influence the workflow and decrease the total experienced workload and communication load of emergency responders during an emergency response.

Specified for the ID this translates to:

> *Can we apply a machine learning method to train a system to determine relevance of dialogue segments for others than the addressee(s) and thereby increase collaborative task-effectiveness?*

In this dissertation, the AWS tool is presented and tested, to investigate if it is able to produce on the fly information about the state of the workflow, workload and communication load of emergency responders as a function of the information exchanged during emergency response. The research question that will be answered in this dissertation translates to the following question.

> *How can we build and ground a generic model to on the fly simulate emergency response and adaptively provide information about the workflow, workload and communication load of emergency responders as a function of the information that is exchanged?*

Based on this general research question, more specific research questions are formulated.

1. What additional developmental and methodological considerations have to be taken into account within the development of the AWS to simulate emergency response?

2. Which modelling and simulation environment is best suited to model the AWS?

3. How can the static elements of emergency response be represented in the AWS?

4. How can we model emergency response workflow?

5. How can we model agent specific workload in the AWS and let it be able to impact the workflow?

6. How can we model the agent specific load associated with handling messages?

An overview of the chapters in this dissertation that separately answer the questions above is shown in Figure 1.4.

Chapter 2 presents the research methods that are used to ground the development of the AWS and will present the standpoints in the development of the AWS regarding developmental and methodological tradeoffs that have to be made when developing a system for the field of emergency response.

Chapter 3 investigates the simulation methods that can be used to develop the Adaptive Workflow Simulator. Based on a set of requirements, the best fitting simulation environment is chosen. Next, this modelling and simulation environment is tested for its capabilities of modelling emergency response practice, answering research question number 2.

Armed with requirements and a modelling and simulation environment that is suited to develop the Adaptive Workflow Simulator in for it to be a generic adaptive simulation of emergency response that operates as a function of information exchanged, Chapter 4 answers the research question if and how the modelling and simulation environment is able to incorporate a reusable generic template of the semi static elements within emergency response (research question 3).

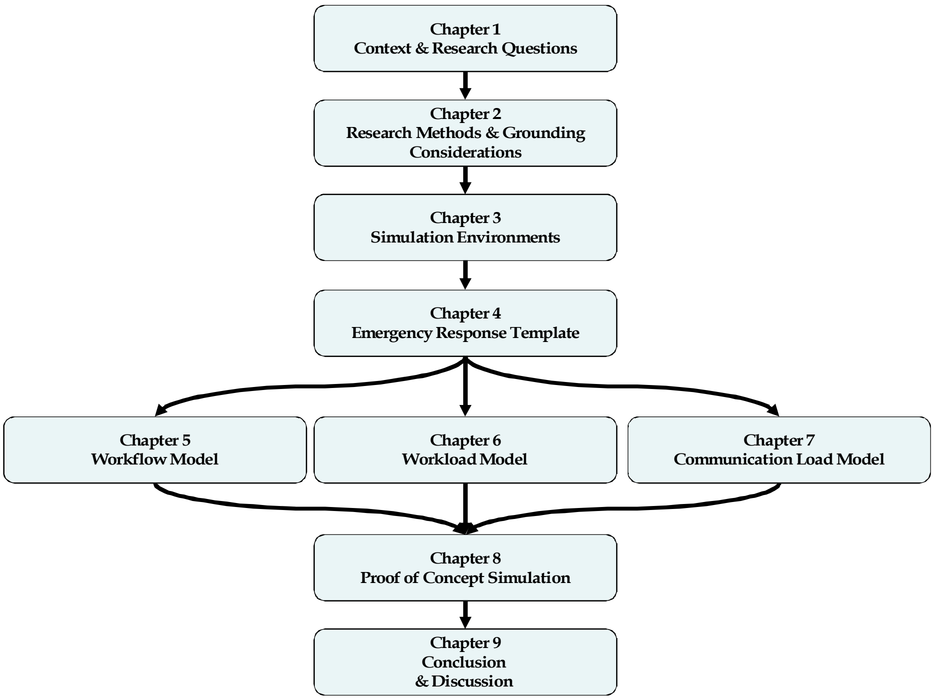


Figure 1.4: *Chapter overview of the dissertation*

Chapters 5, 6 and 7 describe the models that are used within the AWS: the workflow model, the workload characteristics and the communication load characteristics.

Chapter 5 presents and tests the workflow model that is used within the AWS, answering the question how to model emergency response practice workflow within the AWS (research question 4).

In Chapters 6 and 7, the models used within the AWS to assess workload and communication load of the emergency responders are presented and tested, answering research questions 5 and 6.

Chapter 8 integrates Chapters 2 to 7 by answering the general research question if we can use the AWS to on the fly simulate emergency response and adaptively provide information about the workflow, workload and communication load of emergency responders as a function of the information that is exchanged between emergency responders. This is achieved by modelling and simulating part of a real life emergency response exercise.

In Chapter 9, the final chapter in this dissertation we summarize and discuss the conclusions regarding the main research question and sub questions posted.

2. Research Methods

The previous chapter introduced the general functional layout of the adaptive workflow simulator. Furthermore, the main research question and sub questions were formulated that need to be answered in order to equip the AWS with the essential functionality for it to simulate emergency response as a function of information exchange. In this chapter, the research methods that are used to answer the main research question and the sub questions are presented (section 2.3). The research methods' objective is twofold; first they are used to gather the necessary data for the development of the AWS in order to ground its development in emergency response practice. Grounding the development of the AWS simulation is quintessential to minimize the gap between reality and the simulation, will lead to a better founded simulation and increases the soundness of the results (Brenner & Werker, 2007; Auf der Heide, 2006; Turoff, Chummer, Van de Walle, & Yoa, 2004). Grounding here refers to the degree in which the assumptions and relations in a simulation are based on empirical findings instead of being hypothetical. The second objective of the methods is to obtain data to test the AWS and the ID with real time data from practice, providing an indication the possible impact of the TAID system in practice.

Before the actual research methods that are used within this dissertation will be presented, the chapter will first elaborate on the unique developmental considerations that have to be taken into account when developing a system which is intended to be used in the field of emergency response (section 2.1). Grounded development of a system for emergency response entails collecting data from emergency response practice. However, data collection about emergency responses poses substantial challenges. These challenges are sketched in section 2.2. The research methods that are used to gather data from practice for the grounded development of the AWS and for testing the AWS and the TAID system are presented in section 2.3. Finally, the conclusions are presented in section 2.4.

2.1 Positioning the AWS System Development process

The nature and detail of the data that has to be collected largely depends on the system that is developed. General developmental considerations for a system intended for simulating emergency response include if the system is intended to be a generic system which can be applied to all emergency response situations, or a situation specific system, tailored for one specific situation (section 2.1.1). Furthermore, it needs to be clear if the system developed aims at being a person or system focussed system, where the first focuses on competency building for the responder and the latter at relieving the responder from certain tasks (section 2.1.2). Thirdly, more specific simulation related considerations, such as the level of empirical detail and fidelity that is used within the system, need to be considered (section 2.1.3).

2.1.1 Generic versus Specialized Systems

More than systems developed for, and deployed in, "normal" organisational settings, systems for emergency response have to make a trade off between the breadth of their application and the level of detail included within the system. Generic systems can be applied to all situations, but do not possess the level of detail that specialized systems can provide. On the other hand, specialized systems can only be applied to specific situations.

Generic systems deployed within emergency response attempt to include as much functionality as possible to serve a wide variety of responders in dealing with a broad range of emergencies. This is often seen in Commercial of the Shelf (COTS) systems, such as the Mil-EMIS (Emergency Management Information System) by Milsoft ict, or the Incident master developed by the Respond company. The major benefit of these types of systems for emergency response is that they can deal with a large number of possible situations, providing assistance by automating general processes, such as task scheduling, resource planning or general information management. These systems relieve the responders from certain tasks and thus have the potential of reducing workload and communication load of the responders. Unfortunately, adding more detail (such as more specific task schedules, more detailed resource descriptions or detailed scenario information) will add functionality for specific events, but will most of the time significantly slow down the system and its use in general. This, in turn, can slow down or even hamper the actual emergency response and will furthermore increase the likelihood of the system, and its users, getting overloaded (French Minister of the Interior Equipment Transportation and Housing, 1999). Finally, such systems demand exhaustive system knowledge and competencies from its users to use it to its full potential. Generic systems are applicable to a large range of situations and can aid a diverse group of responders, but find their limitations in the (lack of) detail of the aid they can provide.

Specialised systems, on the other hand, used in, for example, forest fires (Keramitsoglou, Kiranoudis, Sarimveis, & Sifakis, 2004), car crashes (e.g., the Crash recovery system by moditech recue solutions) or flooding (de Groeve, Kugler, & Brakenridge, 2007) are tailored for certain situations and/or for certain organisations. The major benefit of these types of systems is that they can aid emergency responders by providing detailed information for specific events for specific users. When unexpected events occur, these systems, however, are in danger of "missing" information, functionality and inter-organisation applicability. With the complexity of an emergency increasing, incorporating several incidents, this would result in the necessity to use multiple different specialist systems simultaneously, increasing workload for the emergency responder.

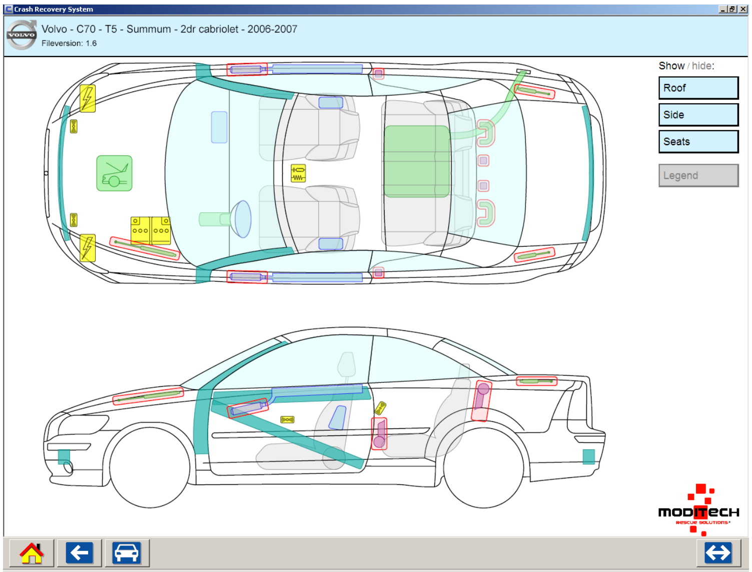


Figure 2.1: *Specialist system example: Crash Recovery System by Moditech recue solutions.*

In Figure 2.1, a screen capture of the specialist car crash software from Moditech is shown. This software is used on a laptop to provide fire-fighters with car specific information about the beams, wiring and no go areas of a car which they can use when extracting a car crash victim from a car. Such a system, while supporting emergency response for car crashes, will not be helpful for extracting victims from, for example, a plane. Changing the setting, changes the system from a life saving to a mostly useless system.

Emergency response practice, interestingly, demands both the general and flexible deployment of the COTS systems to deal with the uncertain aspects of disaster response, and the specificity of the specialized systems for swift and tailored response without the ballast that “other options” bring about. Both system types, however, have their specific pros and cons concerning their level of detail and applicability. When designing a system which has to operate within emergency response, such as the AWS, it has to be determined which level of detail is used within the system and what the breadth of the application will be.

AWS Implications and standpoints
Systems developed for the field of emergency response have to make a trade off between the breadth of their application and the level of detail within the system. Generic systems can be applied in all situations, but do not provide the level of detail specialized systems can provide, while specialist systems can only be applied to specific situations. By addressing the topic of information distribution during emergency response, the TAID system, in essence, is a generic system, finding its application in all emergency response situations. However, by only dealing with the topic of information distribution, it can be seen as a specialized system as well. The TAID system therefore can be considered as a specialized system with a broad application area.

The two subsystems that make up the TAID system share this characteristic. The ID uses general technology from artificial intelligence to classify information relevance that can be applied to all situations. The information however, can vary from being highly detailed to being highly specific. The AWS uses the information to approximate the tasks of the emergency responders. Depending on the level of detail in the communication about the tasks, these are either generic or detailed.

Communication and information exchange during emergency response thus determine the level of detail used in the Information Distributor and the Adaptive Workflow Simulator. The level of detail varies, depending on the communication during the response. The TAID system thus needs to be able to cope with detailed information when this is needed in specific situations, without jeopardizing its general applicability to a wide range of possible emergency response situations.

The next section will discuss the role the system will play in emergency response by addressing the design focus.

2.1.2 Person versus System Design Focus

"Technology is vital in extending our human capabilities to cope with either natural or manmade disasters, but we forget the human role at our peril. This means the human as part of the system, the computer as part of the team, and both the computer and the human working with other people and other computer systems in other agencies, sharing information and working together to manage the emergency, mitigate its effects, and to support the victims after the event." (Carver & Turoff, 2007)

It has to be clear what role the system is going to play within the emergency response and how it is able to aid, and work together with the emergency responders. The first role, that is often seenin systems developed by emergency response practice, tend to utilize a person centred design focus, investing in the responder's competencies by teaching the responder to, for example, deal with certain situations, provide strategies to enhance decision making or information literacy. The second role is often seen in research which has a technological or system focus, which aims at providing "smart" support for the emergency responder during the actual response, relieving and supporting the responder during the response.

Person centred systems emphasize the gain of knowledge of emergency responders through emergency response exercises and training, education and protocol development. Applied in emergency response, these systems are aimed at making the emergency responder information literate so he/she can deal with the vast amount of information and protocols relevant in emergency situations (Lloyd & Somerville, 2006; Boin et al., 2004). These training systems address empirical problems, possess a high level of grounding and have a high level of external validity. Examples of such systems are the Advanced Disaster Management Simulator (ADMS) or the IDM trainer, which are used as simulation training environments by The Netherlands Institute for Physical Safety for emergency responders to practice decision making and planning during the emergency response. The person centred systems are mainly used for training personnel. The system's role thus is to train the responders and provide them with the competencies to better address the issues during the actual response. The person centred system lets the emergency responder work smarter in case of an emergency, by investing in the emergency responders themselves.

Systems centred development has its primary focus on system development itself and applies “smart systems” within the domain of emergency response. The systems developed using this focus in emergency response, mostly have the same goal as is seen in person centred systems – make the emergency responder deal with the vast amount of information that is available – but addresses this with providing technological solutions for the problems during emergency response. These systems, for example, address distributed systems architecture, human machine interaction or decision support in emergency response, but possess a low level of empirical grounding and have a low level of external validity to emergency response practice. The system centred development lets the emergency responder work smarter in case of an emergency by investing in smart support for the emergency responders.

The person centred focus in emergency response thus is inclined to emphasize and invest in how the responder can deal with information, neglecting ways to deal with the information in other ways, while systems centred development emphasizes the “system” to deal with certain aspects within the emergency. When developing a system for emergency response, it has to be considered what kind of focus will be taken to approach the problem for which the system is a solution, and, by doing so, determine the role that the system is going to play in emergency response.

AWS Implications and standpoints

To achieve the goal of the TAID project, it uses a combination of the system centred approach with grounded development. The system centred approach lays the focus on the system for aiding the emergency responder by developing a support tool. In the TAID project, this is achieved by the development of the information distributing system that addresses the information dissemination issues that are seen in emergency response practice. It furthermore uses extra information about tasks, the workflow and the cognitive state of the emergency responders to optimize this distribution.

The use of the adaptive workflow simulator of emergency response, which approximates and provides information about the workflow and workload of the responders to the information distributor, only has added value when it is able to provide valid information about these elements. For the AWS it therefore is crucial that it is grounded thoroughly in practice and achieves a sufficient level of functional fidelity.

A well grounded simulation can, furthermore, be a valuable addition for emergency response (Boin et al., 2004) for testing and evaluation of technology and work practice. “[Simulation] enables the study of various issues, such as feasibility, behaviour and performance without building the actual system, thus saving precious time, cost and effort. A simulation can be adjusted to run at any speed relative to the real world and according to possible scenarios. The results gathered from the simulation indicate how the real system behaves, thus enabling researchers to understand and improve on their design without the actual implementation.” (Sulistio, Yeo, & Buyya, 2004)

The TAID project, as a consequence, is developed using the system focus, but also addresses the validity issues often seen within this approach by grounding the simulation in practice.

2.1.3 Level of Simulation Detail

Modern computers, virtual reality and simulation techniques are able to minimize differences between the simulation of emergency response situations and the actual operational situation that is seen during the actual emergency response. The degree of similarity between a simulation and the operational situation which is simulated is referred to as the simulation fidelity (Hays & Singer, 1988). This similarity can be divided into the physical and functional fidelity of a simulation.

The physical fidelity refers to the degree in which the simulation represents the look and feel of the operational situation: does it look like the actual situation? The physical situation of the simulation closely represents the operational situation. Examples of simulations with a high physical fidelity are simulators that are used for training pilots, simulations used for teaching the operation of cars and air traffic control tower simulators. In the area of emergency response, high physical fidelity simulators, for example, are the Advanced Disaster Management System (ADMS) (see Figure 2.2) or the IDM trainer, which are used as simulation training environments by The Netherlands Institute for Physical Safety, the virtual clinic (Losh, 2007) or the Trusim triage trainer (www.trusim.com). These simulation environments try to match the operational situation as closely as possible, so the responder can interact with it.

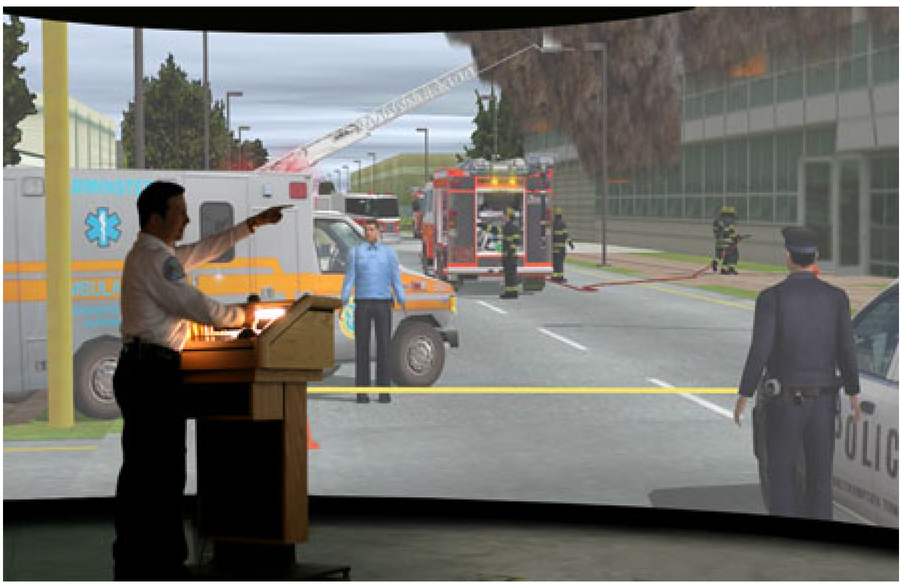


Figure 2.2: *ADMS Command system. (source: www.admstraining.com)*

In Figure 2.2, the ADMS command system is shown. Within the simulation the commander is able to navigate through the virtual emergency and practice with such an emergency in real time. The consequences of the commander's decisions are dynamically reflected in the behaviour of the simulation.

In high physical fidelity systems, not all elements from the operational situation are represented with the same level of detail. This level of detail is determined by to what extent an element is relevant for attaining the goal of the simulation. A simulation about general fire fighting procedures, for example, does not have to contain a detailed model of the combustion engines used within the fire trucks. This, however, can be useful in simulations about tackling engine or car fires. In a general fire fighting simulation, this information is irrelevant for the simulation's goal and will result in simulation overhead, decreasing the performance of the simulation overall. Furthermore, adding detail and manoeuvrability will significantly add cost to the development of the simulation, while not adding much functionality. One therefore can tune the simulation detail to the functional level that should be addressed within the simulation.

The degree in which the simulation represents the functional characteristics of the operational situation and refers to the informational and stimulus response options within the simulation environment, is defined as the level of functional fidelity (Hays & Singer, 1988). Does the simulation environment evoke similar stimulus response reactions as is seen in the operational situation, or, does the simulation behave as the real situation? The degree of functional fidelity of a simulation represents the external validity of the simulation.

For a system to represent certain phenomena, it has to have a fairly high functional fidelity, however, the physical fidelity can vary. Low physical fidelity simulations make an abstraction of the situation by providing only the elements of interest in a simplified manner, without the burden of having to implement all elements within a high physical fidelity system. When designing simulation systems, it must first be determined to what degree which elements should be implemented and with which detail, to achieve the simulation goals. This strongly determines what type of data and what level of detail the data used to build the system should have.

AWS Implications and standpoints

Since the focus of the TAID project and the AWS is on information exchange and workflow, aspects within the simulation should be reduced to the level on which is communicated during emergency responses. This limitation reduces the number of attributes and states of a concept to the number of attributes used in communication.

By having this limitation, we intend to create a flexible, adaptive simulation of emergency response that focuses on the aspects that significantly differentiate emergency response from day to day incident responses: workflow complexity, communication and information. By describing these reoccurring aspects in detail, and simplifying others, the simulation's applicability to other emergency types can significantly increase. A general emergency response simulation template emerges.

2.2 Methodological Challenges and Implications

Minimizing the gap between reality and the simulation by using empirical data instead of hypothetical assumptions, entails overcoming methodological issues associated with conducting empirical research in the complex setting of emergency response (Killian, 2002). The nature of the emergency and the tailored response by the emergency responders rule out the application of commonly used methods that are applied to ground systems developed for "normal" organisational settings. In addition, systems built to operate in emergency response, require additional development effort to meet the requirements raised by the context in which they will operate (Turoff et al., 2004).

Emergencies are characterised by their single occurrence, their sudden and fast changing character and by that they never exactly follow the same route. The emergency response organisation changes its occupation, and consists of multiple sub organisations that are organized monodisciplinary but have to act multidisciplinary, have their own sub culture, and use different information systems in their day to day routine. Within each sub organisation, also regional differences between information systems and organisational structures exist. The tasks that the members of the emergency response organisation execute, furthermore, change according to the demands of the situation, are time pressured, ad hoc, and consist of snippets from multiple protocols. These adaptive characteristics of the emergency and the response, pose difficulties for emergency response research to assess accurate and current information about the setting, organisation, tasks and information exchanged during the actual emergency response. The limited frequency of occurrence, the adaptive and complex character of the emergency and the response to it, hamper real time data gathering and make it hard to determine when, where and what to look for during emergency responses.

Real time data gathering in emergency response is furthermore challenged by difficulties that originate from the multi modal way in which data or information is exchanged, documented and used. During emergency response, information is exchanged and used which is documented (such as protocols and procedures, plans of attack, geographical information), verbal (radio communication, face to face communication) or non verbal (signs, gestures). Emergency response research is faced with the difficulty of coming to grips with these different modes, and the volatile character of the information (Manoj & Baker, 2007). Real time data gathering thus is hampered by the scattered and complex nature of the information, making it hard to determine where to look for to extract meaningful data.

The combination of the limited occurrence of emergencies, their highly unpredictable timing and progression, and the volatile and multi modality of information and a complex information exchange process, makes that "There is no area of social research in which the scientist must operate with less freedom than in the field of disaster study" (Killian, 2002). Research into emergency response phenomena is faced with the impossibility to conduct experimental or quasi experimental research in actual emergency response settings, which excludes causal inference about the impact of these phenomena during actual emergencies. Furthermore, comparison between similar emergencies is hampered by differences caused by local response nuances (regulations, protocols, resources), differences between emergency timing (changed regulations, implemented changes based on lessons learned), or symmetry breaking events (Corbacioglu & Kapucu, 2006).

To overcome the difficulties associated with gathering real time information of an emergency and the subsequent emergency response, post emergency reconstructions can be held to fill missing data points or collect additional information. Although these reconstructions are able to provide lessons learned on a general level, unfortunately the data used can be prone to recall biases, caused, for example, by persons' or stakeholders' interests. This can, consequently, lead to the release of incomplete, methodologically non transparent or incorrect data (Auf der Heide, 2006). When fully basing system development on reconstruction data, it can lead to systematic errors, jeopardising the validity of the system.

A second way to overcome the difficulties to obtain real time information, is to collect data from more frequently held incidents or emergency response exercises that are based on re-enactments of snippets from past emergencies or enactments of possible future emergencies. Common emergency responses such as, for example, responses to small fires, can be monitored more easily given the limited number of actors and low complexity of the emergency and the response. Furthermore, their commonality enables comparison between emergencies, opening the door for quasi experiments. However, as mentioned by Maclean (1983) "Not much about fighting big fires can be learned by fighting small ones", since the difficulties of large scale emergencies differ and surpass the issues present in smaller incidents (Auf der Heide, 1989). As a consequence, the findings are limited, being applicable to only the response to smaller incidents.

The complexity and size of emergency response exercises, however, can be manipulated by the researcher. Emergency response exercises can be held in a controlled environment, where both the incident scenario and actors are known, and allow some experimental manipulation. The price paid for experimental control, however, lies in generalizability since, for example, stress levels and the consequences of actions differ significantly from actual emergencies. Furthermore, given the costs, the time investment of the emergency personnel associated and the inconvenience caused for citizens, large scale emergency exercises cannot be held too frequently.

To summarize, when developing a system for emergency response, grounding is essential to minimize the gap between reality and the system developed. Valid data that can be used for grounding is difficult to acquire. This is due to the adaptive character of the response itself; the volatile and multimodal content of communication and information; lack of experimental control within and between emergencies; post emergency recall biases; the limited prescriptive value of small emergency responses for large emergency responses; and the limited generalizability of emergency response exercises to actual emergencies.

AWS Implications and standpoints

Firstly, the adaptive workflow simulator, which is developed and tested in this dissertation, provides the information distributor with information about the tasks, workload and communication load of emergency responders. The AWS approximates these attributes of the responders by using the information that is exchanged during the response. As mentioned earlier, information is exchanged using different modalities and is hard to retrieve, given the complexity of the emergency response. Key for the development of the AWS is to focus on information that: has a sufficient level of detail to derive the responder attributes for most people involved in the emergency and the emergency response; has a generic source that is present in all emergencies so it can be retrieved in different types and sizes of emergencies; can be retrieved and monitored in "real time" thus excluding recall biases. Following from this, it has to be determined at which level of the emergency response organisation these requirements can be met.

Secondly, in order to overcome the generalizability problem associated with common emergencies and exercises, the AWS has to focus on the generic elements in the emergency response that are similar over the different types of emergencies. These generic elements can be derived from both exercises and past emergencies, not fully relying on one of the two, but using their overlap to develop a generally applicable, reusable model, reducing the likelihood of incorporating systematic error.

2.3 Methods Used for Grounded AWS Development

Grounding the development of a system in emergency response poses practical and developmental challenges for the developer. Practical challenges include methodological challenges to acquire the necessary data to build and ground the system in practice, where the developmental challenges lie in the requirements associated with developing a system for the emergency response domain. Based on section 2.1, two main development principles, shaping the development of the AWS, are adopted:

- AWS uses a generic template based system, applicable to many situations with the possibility to increase detail. The level of detail is determined by the level of detail present in the communication during emergency responses.
- AWS uses a system development focus, using data from practice to ground the elements incorporated in the system.

This section will describe the data acquisition methods that are used during the development of the AWS, taking these principles into account.

The first part of the development of the AWS leads to the development of a generic (static) emergency response template that acts as the main modelling architecture for the AWS. Section 2.3.1 will describe the data acquisition activities to develop this template, using two observational studies which results are used to, firstly, determine and ground the level of detail incorporated in the generic emergency response template (that will be presented in Chapter 4), and, secondly, determine the organisational focus for the proof of concept simulation that will be presented in Chapter 8. Since the results from the first goal directly influences empirical data acquisition for the proof of concept simulation model, results leading to the determination of the organisational focus of the proof of concept will also be presented in section 2.3.1.

Expanding the generic AWS template, the second part of the development work leads to the incorporation of generic models in the AWS that approximate the workload and communication load of the individual emergency responders included in the simulation. Section 2.3.2 will describe the expert questionnaire used to acquire data to ground the models that will be presented in Chapter 6 (regarding workload) and Chapter 7 (regarding communication load). Although general results of the questionnaire will be presented in the chapters describing the individual models, initial results such as the experts' demographics will be presented in section 2.3.2.

Finally, based on the organisational focus determined by the two observational studies used to develop the AWS template, the final part of the development a proof of concept simulation is built to determine the applicability of the AWS (incorporating the static and adaptive models) to model emergency response using the information that is exchanged during the response. Section 2.3.3 will describe the data acquisition to develop this proof of concept simulation by using another observational study of emergency response.

2.3.1 Grounding the Emergency Response Template Model

The aim of the AWS template that will be presented in Chapter 4, is for it to be used as a generic model of emergency response which can act as the main modelling architecture for the AWS emergency response simulation; a reusable template consisting of the elements that are shared between emergency responses. Grounding the template model is key to successful future applicability and reusability and, furthermore, prevents over- and under-specifying the elements and their level of detail which are incorporated in the model.

Under-specifying the model can lead to an incomplete template, shifting additional effort to the actual modelling phase. Furthermore, under-specifying can lead to unforeseen shortcomings in the general structure of the model that are hard to undo. Over-specifying the model, on the other hand, will shift additional effort to the development phase with the danger of incorporating redundant detailed (hard to model) elements, cluttering the model.

This section presents the data acquisition methods that are used to ground the AWS template in emergency response practice in order to prevent over- and under-specifying of the template model. Given the focus of the AWS on modelling workflow, workload and communication load as a function of the information that is exchanged, the AWS template, consequently, must be able to represent the typical information that is exchanged during emergency response practice. The main research question addressed by the data acquisition methods therefore is:

- Which topics typically emerge in emergency response practice situations?

Furthermore, since new information typically emerges from the operational level (units in the field) and is condensed for the first time at the COPI (coordination at the incident location) level, the likelihood to encounter rich information about the workflow of the emergency response organisation is high at these levels. Therefore, it needs to be determined which of these levels of the emergency response organisation will be the focus of the proof of concept model presented in Chapter 8. The second research question that will be addressed therefore is:

- What level of the operational emergency response organisation is best suited to be the focus of the proof of concept simulation?

As was indicated in section 2.2 regarding the methodological challenges faced when collecting empirical data, emergency response exercises provide a controllable environment that can be used to overcome the difficulties to gather real time information exchanged in real emergencies. From an information perspective it, furthermore, can be assumed that no large differences exist between the type of information that is exchanged during emergency response exercises and actual emergency responses.

In order to ground the development of the AWS template and determine the organisational level best suited for the proof of concept simulation, two different emergency response exercises were attended and transcribed. One emergency response exercise concerned an exercise at the operational level (section 2.3.1.1) and one exercise concerned an exercise on the COPI (commando incident location) level (section 2.3.1.2). In these sections an overview of the most prominent topics that emerged during the emergency response exercises are presented. Combined with information about the flow of information, a comparison between the two emergency response exercises can be made (section 2.3.1.3), leading to the determination of the best suited organisational focus for the proof of concept simulation.

2.3.1.1 The Operational Emergency Response Exercise

The emergency response exercise that was observed at the operational level of the emergency response organisation, concerned the emergency response to a major traffic accident. The mock traffic accident was situated in an indoor location and simulated an accident on a provincial road at night during a dense fog. In total 9 cars, 2 vans, 1 truck, 1 trailer and 1 motorbike were involved, leading to 22 victims and a fire. Using limited resources, the fire fighters, commanding officers and the officer on duty of the fire department had to rescue the trapped victims, manage the walking wounded and put out the fire. Figure 2.3 shows a picture taken during the mock emergency response, showing the working conditions under which the response had to be performed.

Figure 2.3: *Extracting a trapped victim from a car during the operational emergency response exercise.*

In total 27 emergency responders actively participated, consisting of four basic fire department trucks (pump-water-tenders), each carrying 1 fire fighting unit (consisting of 5 firemen and one unit commander); one rescue vehicle carrying 2 firemen; and one officer on duty. The duration of the mock emergency response was approximately one and an half hour.

Information about the topics that were communicated about during the emergency response exercise were extracted by monitoring the walkie-talkie communication between the dispatcher and the units in the field (unit commanders and officer on duty), between the officer on duty and the commanding officers and between the unit commanders. Communication within the fire department units was not recorded since this typically possesses a very high level of detail and is very fragmented, given that a unit most of the times works in close proximity of each other and often mutually adjust its activities using face to face communication. Finally, given the complexity of recording, face to face communication during emergency response situations in the field (noise, number of emergency responders involved), was also not recorded. It, however, can be assumed that the general topics (such as victims, parties involved in the accident) exchanged through the walkie-talkie channels are similar to the general topics about which information is exchanged in the field.

Topic Extraction

The recorded information exchange during the mock emergency response consisted of unconstrained natural spoken language. In order to be able to derive topic information from natural language, the recordings were transcribed and coded. The transcriptions revealed a total of 374 utterances (single speech acts). However, since a single utterance is able to contain many topics, utterances were subdivided in order to reveal the unit of analysis: a single topic on which information is exchanged (referred to as an information element).

Similar to the “act” that is used as the unit of analysis in the Bales group communication analysis (Bales, 1970), an information element contains a subject (topic of which is spoken) and a predicate (what is said about the topic) and is sufficiently complete in order for the receiver to interpret the information element and act on it. Within the AWS, subjects refer to the elements that have to be incorporated in the template model (see Chapter 4) and predicates refer to the attributes and attribute values that can be assigned to these topics. For example, the information element “the car is red” has a topic (the car), an attribute (colour) and an attribute value (red). The attribute value also can be “unknown”, representing the communication of absence of knowledge (question) concerning topic and attribute combination. For topic extraction, communication thus is divided into utterances, that in turn consist of one or multiple information elements which incorporates a single topic, a single attribute and a single attribute value.

Unconstrained natural language, however, often is fragmentary and uses references to, for example, previous topics, the sender or the receiver. This makes it difficult to extract information elements by just using the information provided in the utterance. When an emergency responder, for example, communicates “I will do that” it is not clear who will be doing what; the topic and attribute remain unclear. However, when the information element is seen in its context as it was the dispatcher who communicated this in reaction to a request from the officer on duty to arrange communication with the head officer on duty, both the topic and the attribute can be filled in using the context in which they occur.

In this dissertation, the topics and their attributes will be incorporated in the general template model presented in Chapter 4. The information element, furthermore, will play a crucial role in estimating the communication load (Chapter 7) of the individual emergency responders.

General results
We have already seen that the total number of utterances that were communicated during the mock emergency response exercise was 374. In addition, 491 information elements were exchanged (average of 5 per minute) on 66 topics that flow through the emergency response organisation. Table 2.1 shows the top 10 topics that appeared during communication, accounting for 57% (282 information elements) of all the information elements communicated.

Table 2.1: *Total number of information elements exchanged of the top 10 topics that appeared in communication during the mock emergency response exercise.*

Topic	Topic examples (translated from Dutch)	Number of information elements exchanged	% from total number of information elements
Accident	"It is a very large accident" "The accident is situated near hectometer marker 69.2"	61	12.4%
Last Message	"The message was loud and clear"	38	7.7%
AC	"I will arrange transportation" "I have no knowledge of that"	34	6.9%
OvD-B	"I am heading towards the OvD car" "Please wait, I am busy" "I am standing next to the yellow van"	28	5.7%
110	"We are about to investigate the truck" "Can the 110 be contacted?"	28	5.7%
6231	"6231 moving to the accident location" "6231 provide mutual aid" "Our current location is at the bridge"	25	5.1%
120	"You can go and pack your vehicle" "You are assigned the head of the accident" "We are located at the tail of the accident" "Our vehicle is stuck"	21	4.3%
6481	"We are awaiting instructions from the OvD" "Provide mutual aid" "We are at the accident location"	20	4.1%
Victim	"Victim is freed" "The victim is situated in the car"	16	3.2%
H-OvD	"He is moving towards the incident location" "The H-OvD takes care of the press" "Is the H-OvD at the incident location"	11	2.2%

The top ten topics refer to characteristics of accident itself, the last message, the dispatcher (AC), the officer on duty (OvD-B), fire department unit 110 (110), fire department unit 6231 (6231), fire department unit 120 (120), fire department unit 6481 (6481), victims and the head officer on duty (H-OvD).

The topics refer also to abstract (non tangible) concepts such as "the accident" or the "last message" and to concrete things, like, for example, a single person, such as the officer on duty. Information elements that were exchanged that had the emergency responders as their topic referred to the activities of the responders (such as moving/arranging/waiting), attributes of the responders (such as busy/location assignment) and location determination (such as located at). Furthermore, some topics that are referred to have multiple meanings. The topic "120", for example, can refer to a single person (the unit commander of unit 120), a group (all persons that are a member of unit 120) or an object (the vehicle that is used by unit 120). For the extraction of topics this, however, has no consequences, since topics are used to cluster information elements that also posses predicates. The topic thus is disambiguated within the information element.

Generic topics that are shared between emergencies and emergency responses, not included in the top 10 topics are, for example, (with the number of information elements exchanged about the topic between brackets): communication network (8), 130 (8), fog (7), HV2 rescue vehicle (6), medical services (5), fire department materials (3), fire (3), source area (1).

When looking at the senders and receivers of information elements during the mock emergency response, two central nodes emerge (Figure 2.4): the officer on duty and the dispatcher.

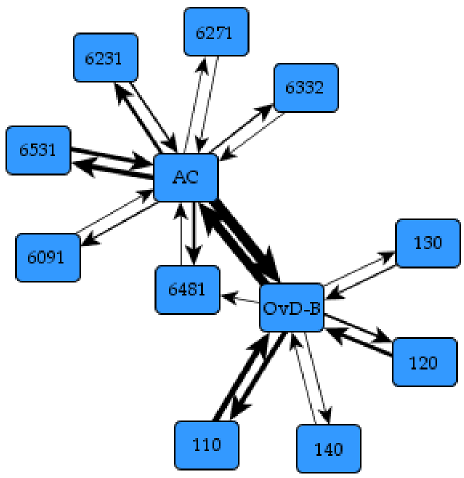


Figure 2.4: *Flow of information elements throughout the emergency response organisation*

In Figure 2.4, the boxes represent the units and responders that were involved in the emergency response that either transmitted or received information elements. The arrows indicate the direction of the information flow. Furthermore, the width of the arrows indicates the number of information elements that are exchanged between the emergency responders. How wider the arrow, the more information is exchanged. Figure 2.4 visualizes the typical information element flow during the total fire department turn out. Interestingly, no information elements were exchanged between the units in the field with the use of the walkie-talkie.

In the beginning of the emergency response, units communicate with the dispatcher (AC) about the accident they are turning out to. In this phase of the emergency response, each unit is addressed by using the number of the vehicle they use to go to the emergency (6091, 6481, 6531, 6231, 6271, and 6332). As can be seen in Figure 2.5, the vehicle numbers are mainly used in the beginning of the emergency response, or when new units are being dispatched. Figure 2.5 shows the cumulative number of information elements that are exchanged during the exercise using 5 minute intervals. The numbers 0 and 1 on the time axis represents the hour in which the five minute interval is located.

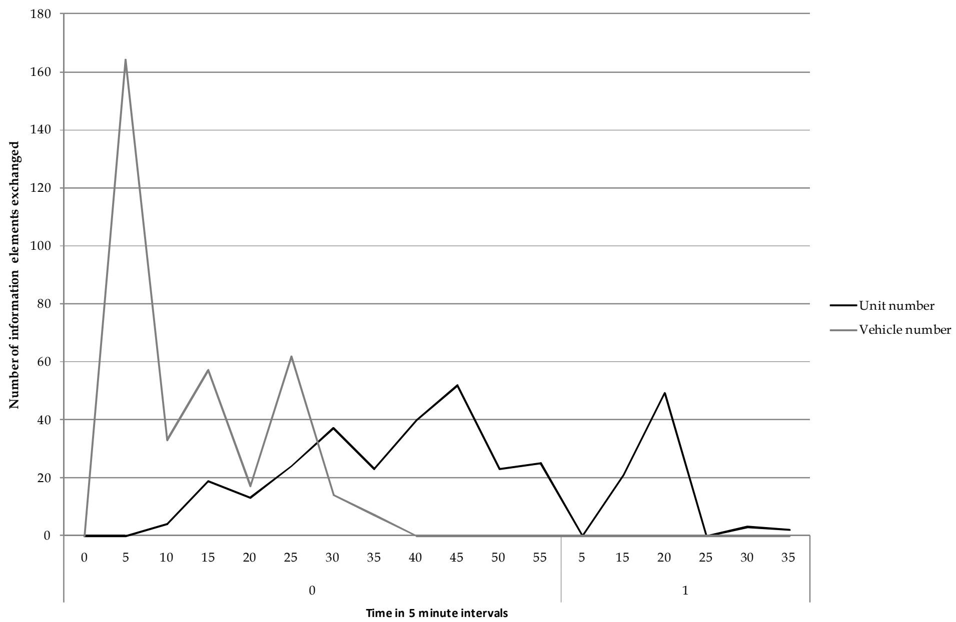


Figure 2.5: *Information elements being sent or received from units addressed by their vehicle number or unit arrival number.*

When the units arrive at the accident location, their activities are coordinated by the officer on duty (OvD-B), which, in turn, is in continuous contact with the dispatcher to coordinate the external consequences of the response. As can be seen in Figure 2.4, the officer on duty sends and receives information elements to and from the commanding officer of the units 110, 120, 130 and 140. These are the same units that initially communicated with the dispatcher during the initial turnout phase, but are now allocated a unit number based on their arrival sequence (the 110 being the first unit and the 140 being the last). In the mock traffic accident, the unit 110 maps to unit 6531, unit 120 maps to unit 6332, unit 130 maps unit 6231 and unit 140 maps to unit 6481. Other vehicle numbers that were used were the 6271 that referred to a rescue vehicle and the car of the officer on duty that is referred to as the 6091. As can be seen in Figure 2.5, during the mock emergency response all vehicles arrived at the scene within 30 minutes, and no additional vehicles were turned out.

Attributes that apply to a unit addressed by their vehicle number thus also apply to the mapped unit when addressed by the unit number based on the arrival sequence.

From these general results, it can be concluded that during the mock emergency response most recorded information elements were exchanged between the officer on duty and the dispatcher and that these are central nodes in the mock emergency response regarding the flow of information at the operational level. Furthermore, it can be concluded that the topics and information elements exchanged by the unit commander regarding the activities of the unit possess the highest level of detail needed in the AWS, the duty and task level (See section 4.1.2 for an elaboration of the desired level of detail in the task model in the AWS). The information elements that surface in the recordings during the mock emergency response, furthermore, consist of a mixture of abstract and concrete objects that are likely to turn up during comparable (traffic accident) fire department turn outs. The mock emergency response, therefore, can be used to feed the AWS template with topics, attributes and attribute values since the "traffic accident" is identified as one of the 18 distinct emergencies (see section 4.1.1.1) with which the emergency response organisation can be faced.

2.3.1.2 The COPI Emergency Response Exercise

In contrast to the mock emergency response to the major traffic accident presented in the previous section, the actual activities of the units in the field in the COPI emergency response exercise were simulated. The focus of the COPI exercise was on coordinating the multidisciplinary response instead of physically performing a mainly monodisciplinary response. However, although the COPI exercise was a table top exercise, it was situated in the mobile commando unit normally used to situate COPI meetings in the field and made use of walkie-talkies to communicate with the (simulated) dispatcher and the units in the field. The emergency responders that actively participated in the emergency response exercise were the officers on duty from the police department, fire department and medical service, the head officer on duty (chairman of the COPI) and the COPI information manager.

The scenario of the mock emergency started with a closed fire at a garden furniture store. Complicating the matter, the furniture store was located in the basement of a school (containing 300 students) where the first unit on the scene discovered an illegal power tap. In the initial phase, furthermore, one person is missing. During the initial reconnaissance, the second unit on the scene discovers a depot holding 3000 kilos of firework at the rear of the building. At this moment, smoke entered the school, which makes the evacuation of the school that was already on the way, more urgent. Several buildings and events are affected by the smoke and the fire. One of these buildings is the local discotheque, where at that moment the annual meeting of the national association of physically challenged youth is held. The emergency response, furthermore, is made difficult due to a limited water supply and due to the crowd that is turning up, being curious about what is happening. Within this hectic situation, a car runs in to a tanker truck that was supplying the nearby LPG filling station with liquefied natural gas. The car immediately sets on fire, and this sets the scene for a possible BLEVE (boiling liquid expanding vapour explosion). The goal of the COPI exercise was to multidisciplinary handle this situation and share discipline exclusive information of multidisciplinary importance.

Similar to the traffic accident exercise, communication between the active participants was monitored at the level that could lead to the discovery of topics that had to be included in the AWS template. The goal of the data acquisition was not aimed at getting a complete overview of all information elements that were exchanged, but was aimed at topic discovery. The communication that was recorded during the COPI exercise, therefore, was limited to the COPI meetings themselves and communication that took place in the COPI meeting room. During these meetings, information is exchanged between the active participants that is of importance for the emergency response as a whole. Walkie-talkie communication between the officers on duty and their commanding officers thus was not recorded. Topic extraction of the monitored meetings was done by using the same topic extraction method that was presented in the previous section.

General results
During the COPI emergency response exercise a total 4 COPI meetings were held. The total duration of the exercise was 49 minutes. During these 49 minutes, a total of 1122 information elements (an average of 23 information elements per minute) could be identified in 573 utterances that touched upon a total of 126 topics. Table 2.2 shows the top 10 topics that appeared, accounting for 36% of the total number of information elements exchanged.

Due to the complexity of the COPI emergency response exercise scenario, a large number of topics surfaced. The key elements describing the emergency and the response, however, can be found in the topics in the top 10 topics list shown in Table 2.2. The commando disaster area group had to deal with a fire (fire topic) in a cellar (cellar topic). The fire department had to deal with the fire (fire department) and the AGS, that is the specialist on dangerous substances (AGS topic) had to do measurements to determine the consequences of the fire for the surrounding area. The two major focus shifts in the scenario also are present, the discovery of fireworks (fireworks topic) and the situation at the filling station. As a consequence of the location of the emergency (school, disco), a large number of civilians were involved that had to be looked after (casualty collection point topic).

However, beside the scenario specific topics that emerged, a large number of topics are shared with other emergency responses, giving them a generic character. Examples of these reoccurring topics from the COPI exercise are (with the number of information elements exchanged about the topic between brackets): GRIP (27), OVI (31), police department (26), smoke (23), cordon (21), OvD-P (21), OvD-B (19), scenario development (17), situation report (17), effect area (14), source area (11), water transport (11), wind direction (10).

Table 2.2: *Total number of information elements exchanged of the top 10 topics that appeared in communication during the COPI emergency response exercise.*

Topic	Topic examples (translated from Dutch)	Number of information elements exchanged	% from total number of information elements
COPI	"What will be the location of the COH" "The next COPI is at 10 minutes past 2" "First we will discuss the current state of the emergency and the emergency response"	103	9.18%
Casualty Collecting Point	"The casualty collection point must be located windward" "The casualty collection point is located in *sporthal de strugel*"	38	3.39%
Filling Station	"Are there still people in the filling station" "Is the filling station stabilized?"	35	3.12%
Fire	"What is the intensity of the fire?" "The fire is under control" "PVC (Polyvinyl chloride) is involved in the fire"	33	2.94%
Fire Department	"The fire department is arranging water transportation" "The fire department capacity is sufficient"	33	2.94%
Cellar	"The cellar is a no go area" "The cellar is located under a school"	33	2.94%
H-OvD	"I will contact you later" "No, I have no knowledge of that"	33	2.94%
Fireworks	"Fireworks in the proximity of the fire" "3000 kilograms of fireworks" "Consumer fireworks"	31	2.76%
AGS	"The next step is that I will be checking for the presence of chloride" "We have already checked for the presence of sulphur dioxide"	31	2.76%
Incident	"It is a very large fire" "There are a lot of bystanders"	31	2.76%

When turning to the flow of information elements between the senders and receivers (see Figure 2.5), one central node can be identified: the head officer on duty (H-OvD). The boxes in Figure 2.5 represent the senders and receivers of information elements during the COPI exercise. Apart from the H-OvD, the COPI meeting was attended by the officer on duty from the fire department (OvD-B); the officer on duty from the police department (OvD-P); the officer on duty from the medical services (OvD-G); the specialist on dangerous substances (AGS) and finally the information manager of the COPI meeting (IM). Since the head officer on duty is the linking pin between the operational level and the tactical level, he furthermore communicates with the officer of communication and information (OVI) which is the information manager at the tactical level and with the regional commander (R-CvD) who is the operational leader (see section 4.1.1 for an overview of the emergency response organisation).

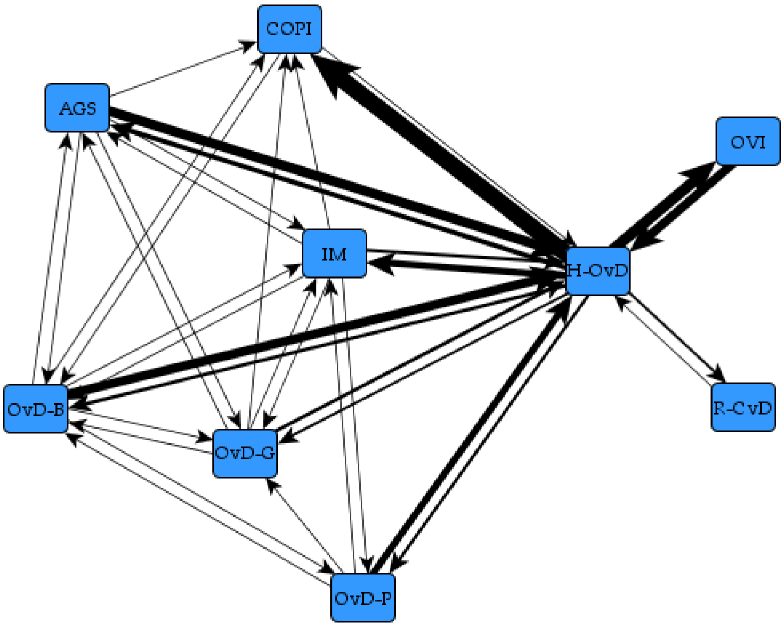


Figure 2.5: *Flow of information elements during the observed COPI meeting.*

Besides addressing the individual responders with directed questions or providing information which is directed to a specific emergency responder, COPI members also addressed the group as a whole (COPI). As can be seen in Figure 2.5, this is mainly done by the H-OvD. To differentiate between general remarks directed to the group and remarks addressed to persons, it should be noted that when the H-OvD asks a COPI member for information, the response from the COPI member was scored as being directed to the H-OvD; the H-OvD thus becomes the sole receiver of the reply, even though the information element is noticed by all members of the COPI. Information elements directed to COPI thus explicitly were directed to the group as a whole.

It is likely that the general communication pattern shown in Figure 2.5 will emerge in most COPI meetings, since the meeting structure is predefined, its key members (OvD-B, OvD-G, OvD-P, IM, H-OvD) are always the same and the H-OvD is the designated person to communicate with the tactical levels. However, depending on the type of emergency, the constitution of the COPI group can change by adding specialists such as the AGS to the COPI group. The width of the connections in Figure 2.5 may, furthermore, change due to the emergency response focus. The main focus of emergency response scenario, that unfolded in the COPI emergency response exercise, was on dealing with the fires (that concerned the activities of the OvD-B and the AGS) and evacuation (that concerned the OvD-P).

From the COPI emergency response exercise that was recorded, it can be concluded that the topics that surfaced, represent key topics of a relatively complex emergency response dealing with fire, rescue and evacuation. The flow of information elements showed that information elements are exchanged between the multidisciplinary COPI members and that this process is directed by the H-OvD. The H-OvD concurrently communicates information elements to the tactical level of the emergency response.

2.3.1.3 Concluding Remarks

The first goal served with the data acquisition activities presented in the previous two sections is to ground the development of the AWS template. This template, in which the reoccurring elements of emergency response are included, provides the modeller with a main modelling architecture in which situation specific elements can be modelled. To build a strong link between emergency response practice and the simulation, and prevent over- or under-specifying the template, the template is grounded by using the general non emergency specific topics that appeared in the two emergency response exercises investigated.

The general results from the two exercises showed that they provide a fruitful source of general and more specific topics that can be used for this purpose. Since the emergency response exercises concern general emergency response activities (rescue, evacuation, tackling fire, dealing with effect area consequences), they provide a sufficient starting point in providing the AWS template with elements that typically surface during emergency response. The combination of the topics from the two exercises (169 topics) therefore is used to guide the building of the grounded template of emergency response.

The second goal served with the data acquisition activities is to determine the organisational focus of the proof of concept simulation. As was indicated at the start of section 2.3.1, new information typically emerges from the field and is condensed for the first time at the COPI level. These operational and COPI organizational levels possess the highest level of detail concerning the workflow of the emergency responders in the field.

The patterns representing the flow of information at the “hands on in the field” operational level and the COPI level, reveal the typical pattern of how information flows through the emergency response organisation. Commanding officers report to the officer on duty, who reports to the head officer on duty who then reports to the officer of communication and information and the regional commander. When information reaches the tactical level, it has been compressed and transformed numerous times. Detailed information elements that emerge at the “hands on level” so are lost and are replaced with information elements that have a higher abstraction level.

It should, however, be noted that the loss of detail of information does not have to affect the overall quality of decision making, since the abstraction of information needed for decisions co-vary with the level at which decisions are made. At the governmental decision making level, for example, it is not crucial to know where the hydraulic spreader is situated and if it is in use. A general picture of the situation will meet the information requirements at this decision making level of the response.

Although decision making levels themselves also generate new information regarding, for example, actions that have to be taken within the emergency response, these decisions are based on information that finds its genesis “in the field”. As indicated in the introductory chapter, missing out on this information or using badly compressed information can lead to ill informed decision making. Faulty information at the "hands on level" so snowballs through the emergency response organisation.

Furthermore, when looking at the information element input at the officer on duty level, they receive information that find its genesis in the field, from the officers on duty from the other disciplines and from the tactical level of the response organisation through the H-OvD. The first information source contains information about, for example, what unit is performing which task in the field; the second information source contains information about, for example, status of a fire, victim status, or police cordons; the third information source, for example, contains information about the measures taken to reduce the indirect consequences of the emergency response.

As can be seen in Figures 2.4 and 2.5, the OvD-B participates in both the emergency response exercises at the operational and the COPI level. To some extent Figure 2.4 thus extends Figure 2.5. As can be seen in Table 2.3, which provides an overview of the actors that are present or simulated by other emergency responders, the OvD-B is the linking pin between both exercises.

Table 2.3: Overview of the present and simulated actors during the observer emergency response exercises.

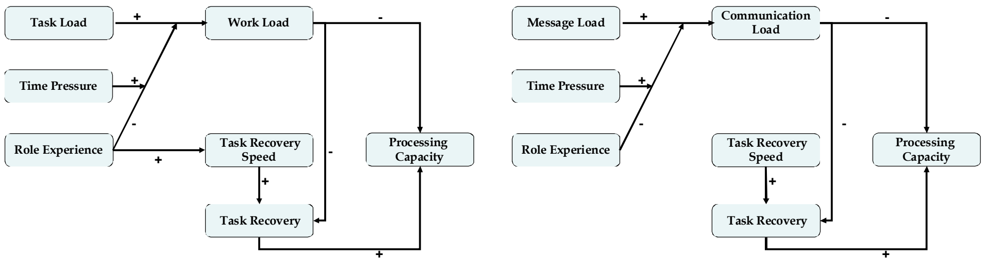
	Operational	COPI
Units	Present	Simulated
Unit commander	Present	Simulated
OvD-B	Present	Present
OvD-P		Present
OvD-G		Present
IM		Present
H-OvD		Present
OVI		Simulated
Dispatcher fire department	Simulated	Simulated
Dispatcher medical services		Simulated
Dispatcher police department		Simulated

In contrast to the exercise at the operational level, the COPI exercise, furthermore, is multidisciplinary and uses a complex (multidisciplinary) scenario during which information plays a crucial role. The course of the scenario can change, due to changes made by the COPI members.

To summarize, given the centrality of the OvD in the emergency response organisation concerning the genesis of information and the coordination of the workflow of the units in the field in combination with the multidisciplinary complex open ended character of the COPI in which the OvD participates, it can be concluded that the OvDs activities at the COPI level are the best suited level for the proof of concept simulation presented in Chapter 8 and for the TAID system as a whole.

2.3.2 Grounding the Emergency Response Workload and Communication Load Models

The workload and communication load models that are presented in Chapters 6 and 7 approximate the impact on the processing capacity of an individual emergency responder due to respectively executing a task and processing information that is communicated.



(a) Workload model (b) Communication load model

Figure 2.6: *AWS workload and communication load model.*

The workload model shown in Figure 2.6 (a), describes a situation where task load is the main predictor of workload (subjective impact related to task execution). Task load refers to the stable attributes of a task which defines its mental and physical complexity. When no moderating factors of the relationship between task load and workload would be present, it could be concluded that task load = workload. However, as can be seen in the workload model, the relationship between task load and workload is negatively moderated by the number of years of role experience of the emergency responder[1]. Furthermore, time pressure amplifies the damping effect of experience on the relationship between task load and workload. Finally, the number of years of role experience has a positive influence on the task recovery speed from task execution. Depending on the present workload, the performer is able to recover and “refill” the available processing capacity which is depleted by the workload associated with task execution. A detailed description of the workflow model can be found in Chapter 6.

The communication load model describes a similar situation as for task load, where the message load is the main predictor for communication load (subjective impact related to processing capacity). Message load refers to a stable attribute of a message which defines its information complexity. The subjective experience of the message load is expressed by the communication load. Similar to the workload model, the relationship between message load and communication load is negatively moderated by the number of years of role experience of the emergency responder; time pressure amplifies the damping effect of

[1] The negatively moderating effect of the number of years of role experience refers to a decrease of the workload associated with execution of a tasks caused by a higher level of experience (expressed by the number of years of role experience). Task load is not affected by the number of years of role experience.

experience on the relationship between message load and communication load. The actual communication load, furthermore, has a negative influence on recovery. Task recovery (recovery from the task of handling messages) also is able to "refill" the available processing capacity that is depleted by the communication load. A detailed description of communication load can be found in Chapter 7.

2.3.2.1 Expert Questionnaire

The theoretical models thus describe how the processing capacity of an individual emergency responder is depleted by workload and communication load, and is "refilled" due to recovery. However, to ground these models, nine questions need to be answered regarding workload, communication load, recovery speed, role experience and model integration. In the form of a questionnaire, these questions were presented to a group of experts in the field of emergency response. Experts possess the domain knowledge needed to ground the elements in the model and possess the knowledge needed for making the abstraction to "translate" the theoretical concepts into practice. The questionnaire administered, consisted of 27 items covering the 9 questions. Using these questions, the general layout of the questionnaire will be presented below. In most cases the questionnaire uses Likert scales to assess the (dis)agreement of the experts with a statement. The five point Likert scale ranges from "strongly disagree" to "strongly agree" with a neutral "Neither agree nor disagree" category. The questionnaire can be found in Annex 1.

Workload

1. Do the number of years of role experience result in a lower workload for experienced emergency responders compared with less experienced emergency responders when executing the same task?

In the questionnaire, this question was addressed with one item (item 11a in the questionnaire) on which the experts had to indicate, on the Likert scale, to what extent they agreed or disagreed with the statement that when a person possesses more experience in the role he/she has during emergency response, he/she experiences less workload than less experienced emergency responders.

2. Do the number of years of role experience result in a lower workload for experienced emergency responders compared with less experienced emergency responders when executing tasks that possess different proportions of mental and physical activities?

In the questionnaire, this question was addressed by five items (items 11b, 11d, 11f, 11g, 11h in the questionnaire), each using the Likert scale. In each of the five items the proportion of which the task consisted of mental and physical activities was varied, ranging from a task that for 100% consisted of mental elements (such as thinking, decision making, remembering) to a tasks that for 100% consisted of physical activities (such as moving, pushing pulling). Table 2.4 shows which combinations of task proportions were included.

Table 2.4: Task combinations assessed to determine the influence of experience on tasks that consist of different proportions of mental and physical activities.

Task combination	Proportion of mental activities in the task	Proportion of physical activities in the task
1	100%	0%
2	75%	25%
3	50%	50%
4	25%	75%
5	0%	100%

To aid the respondent, three of the five combinations (2-3-4) included a pie chart, representing the proportion in which the task consisted of mental and physical activities (Figure 2.7).

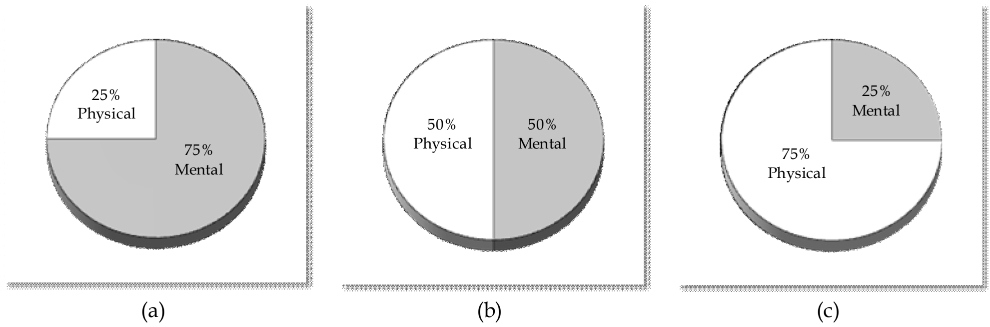


Figure 2.7: *Graphical representation of the proportion in which a task consists of physical and mental activities.*

The items addressing combination 1 and 5 in Table 2.4 were administered using a Likert scale, without the graphical representation shown in Figure 2.7.

3. How can we represent the relationship between the number of years of role experience and workload as a consequence from executing mental and physical tasks?

In the questionnaire, this question was addressed by two items (item 11c and 11e in the questionnaire), asking the experts to indicate which of the nine graphs shown in Figure 2.8 to their judgement best represented the relationship between role experience on the horizontal axis, and, firstly, workload caused by the execution of mental tasks, and, secondly, workload caused by the execution of physical tasks on the vertical axis. How can this relationship between the number of years experience in performing a task and the workload associated with that task be represented? Will the workload decrease over the years; will the workload increase over the years; or will the workload be stable over the years. The graphs shown in Figure 2.8 represent these plausible “simple” relationships that can exist between experience and workload.

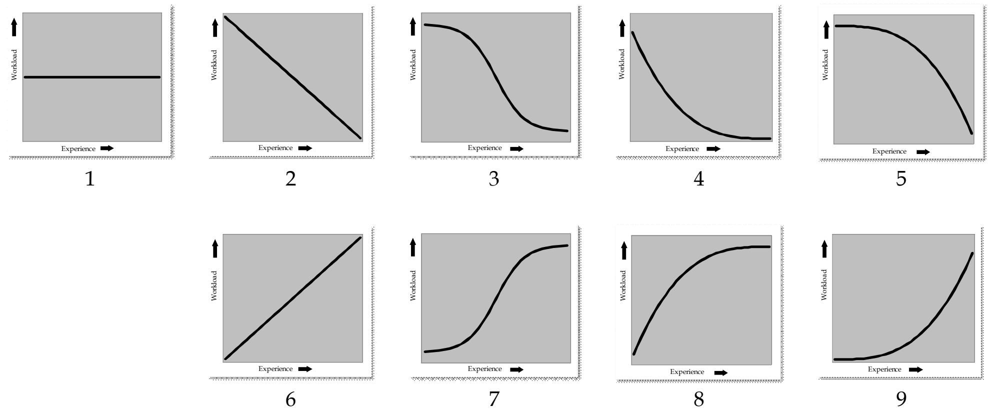


Figure 2.8: *Nine relationship graphs used in the expert questionnaire to indicate the relationship between role experience (horizontal axis) and workload(stemming from mental or physical tasks).*

Graph (1) indicates that there is no relationship between experience and workload. No differences exist between the workload of experienced emergency responders versus less experienced emergency responders, given the same tasks. Graph (2) indicates a negative linear relationship between experience and workload. The workload of the emergency responder when performing the same task decreases with a fixed amount with each extra year of experience gained by the emergency responder. Graph (6) is the mirror image of graph (2). The workload of the emergency responder when performing the same task increases with a fixed amount with each extra year of experience gained by the emergency responder.

Graph (4) indicates a z-shaped relation between experience and the workload of the emergency responder when performing the same task. During the early years of the career of the emergency responder the workload when executing the same task gradually decreases. After some years, the workload associated with that task decreases quickly and stabilizes in the last phase. An extra year of experience in this last phase thus leads to a relative small decrease of the workload related to execution of a task compared with the decrease in speed in the middle phase. Graph (7) is the mirror image of graph (4), an s-shaped relation between the number of years experience in performing a task and the workload associated with that task. Workload increases slightly in the first phase, increases fast during the second phase and stabilises in the third phase.

Graph type (8) is the mirror image of graph (4). It shows a quick increase of workload related to the execution of a task in the first years that stabilises towards the end.

Graph (5) shows a slow decrease of workload associated with the execution of a task in the first phase of a career. With every year of experience added, the workload will decrease with larger steps. Graph (9) shows the mirrored graph of (5). At the start of the career the workload associated with the execution of a task increases slowly with each added year. However, with every year of experience added, the workload will increase with larger steps.

Communication Load

4. Do the number of years of role experience result in a lower communication load for experienced emergency responders compared with a less experienced emergency responders when handling the same amount of information.

In the questionnaire, this question was addressed with one item (item 8c in the questionnaire) on which the experts had to indicate on a Likert scale to what extent they agreed or disagreed with the statement that that when a person possesses more experience in the role he/she has during emergency response, he or she experiences less communication load compared with less experienced emergency responders.

5. How can we represent the relationship between the number of years of role experience and communication load?

In the questionnaire, this question was addressed by one item (item 9 in the questionnaire) in which the experts were asked to indicate which of the nine graphs shown in Figure 2.8 to their judgment best represented the relationship between role experience on the horizontal axis and communication load on the vertical axes. What is the communication load of the same amount of information for emergency responder with varying levels of experience?

6. Does an increase of the number of years of role experience result in a better and quicker assessment of the relevance of information?

In the questionnaire, this question was addressed with two items (items 8a and 8b in the questionnaire) on which the experts had to indicate on the 5 point Likert scale to what extent they agreed or disagreed with the statements that:

- When a person possesses more experience in the role he/she has during emergency response, he or she is able to better assess the relevancy of information when compared with less experienced emergency responders.
- When a person possesses more experience in the role he/she has during emergency response, he or she is able to quicker assess the relevancy of information when compared with less experienced emergency responders.

7. Does communication load at the tactical and strategic level of the emergency response organisation increases when the emergency has a higher GRIP status, and secondly to what extent does the communication load increase when more emergency responders are involved in the response?

In the questionnaire, this question was addressed with four items (item 10a, 10b, 10c, 10d in the questionnaire) on which the experts had to indicate on a 5 point Likert scale to what extent they agreed or disagreed with the statements that:

- The communication load at the tactical level increases when the incident has a higher GRIP level.

- The communication load at the strategic level increases when the incident has a higher GRIP level.
- The communication load at the tactical level increases when more emergency responders are involved in the emergency response.
- The communication load at the strategic level increases when more emergency responders are involved in the emergency response.

Recovery Speed

8. Do the number of years of role experience result in a quicker recovery speed from mental and physical tasks for experienced emergency responders when compared with less experienced emergency responders?

In the questionnaire, this question was addressed with two items (item 12a and 12b in the questionnaire) on which the experts had to indicate on the 5 point Likert scale to what extent they agreed or disagreed with the statements that:

- When a person has more experience in the role he/she has during emergency response, he or she is able to recover more quickly from the workload resulting from physical activities (moving, pushing, pulling) when compared with less experienced emergency responders.
- When a person possesses more experience in the role he/she portraits during emergency response, he or she is able to recover more quickly from the workload resulting from mental activities (thinking, decision making, remembering, searching) when compared with less experienced emergency responders.

Load Contributing Factors

9. To what degree is the experienced load during emergency response (at the operational, tactic, strategic and governmental level of the emergency response organisation) determined by workload, communication load, time pressure, (in)ability to recover during tasks and multitasking?

In the questionnaire, this question was addressed with four items (item 4, 5, 6, 7 in the questionnaire), covering the different levels of the emergency response organisation. In each item, the experts were asked to distribute 100% over the five pre-specified load contributing factors. In addition, they could specify two other factors that they thought determined the load on the emergency responder during an emergency response. The pre specified factors were:

- The mental and physical aspects of the tasks executed
- The quantity of information that is exchanged
- Time pressure
- The inability to recover from tasks
- Multitasking

2.3.2.2 Procedure and Respondents

In total 90 experts were approached to fill out the questionnaire described above. The questionnaire was distributed via e-mail to emergency responders that were involved in the tactical, strategic, local government and regional governmental layers of the emergency that were active in the security region Twente the Netherlands. In addition, the questionnaire was distributed via e-mail to the members of the special interest group of the lectureship crisis management of the *The Netherlands Institute for Physical Safety* and the *Police Academy*. The special interest group also was provided with a paper and pencil version of the questionnaire during a meeting of the group. Table 2.5 shows the number of invited people and the response to the questionnaire for the subgroups.

Table 2.5: *Response to the expert questionnaire*

	Invites	Response
SIG crisis management	32	7 (22%)
SR Twente	58	19 (33%)
Total	90	26 (29%)

From the 90 experts that were invited to fill out the questionnaire, 26 (29%) responded (see Table 2.5). Given that participation was voluntary, no incentives were provided and none of the responders knew the researcher nor knew about the research, this is a satisfactory response rate when compared with response rates on undirected questionnaires in general. The respondents were affiliated with the traditional parties that are involved in emergency responses (fire department, police department and medical services) and with an organisation that provided education and did research in the field of emergency response and crisis management (see Figure 2.8).

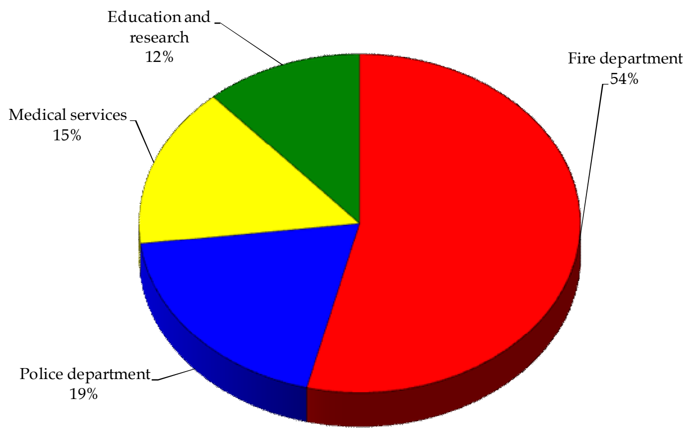


Figure 2.8: *Organisational affiliation of the respondents to the questionnaire.*

From the 40 roles that the 26 experts had in emergency response, the roles that were most prominent in the responder group were officer on duty (25%), chief officer (10%) and staff member of the regional operational team (7.5%). Thus most of the roles that are present in the emergency response organisation also are represented in the group that responded to the questionnaire.

The responders possessed a broad range of experience on the multiple levels of the emergency response organisation. The respondents had an average of 12,67 years (N=21, SD=9.86) of working experience on the operational level, with a minimum of 1 year and a maximum of 31 years of experience; an average of 11 years (N=21, SD=8.64) of working experience on the tactical level, with a minimum of 1 year and a maximum of 34 years of experience; an average of 10,88 years (N=16, SD=9.15) of working experience on the strategic level, with a minimum of 3 years and a maximum of 35 years of experience; and finally an average of 10.18 years (N=11, SD=9.62) of working experience on the governmental level of emergency response organisation, with a minimum of 1 year and a maximum of 31 years of experience. It should be noted that, since not all respondents had working experience on all levels of the emergency response organisation, the number of respondents that answered the questions (N) that relate to their experience on the levels is different. In this respect, 21 of the 26 respondents have experience on the operational level; also 21 of the 26 respondents have experience on the tactical level; 16 out of the 26 respondents reported to have working experience on the strategic level; and 11 out of 26 respondents reported to have working experience on the governmental level.

From all respondents, 81% had a direct working experience on the operational level, 81% had a direct working experience on the tactical level, 62% had a direct working experience on the strategic level; and 42% had a direct working experience on the governmental level. Since all respondents perform or had performed an active coordinating role in the emergency response organisation, all respondents were familiar with the activities performed at all levels of the emergency response organisation. Thus, all levels of the emergency response organisations are sufficiently represented by the respondents of the questionnaire.

Although the response to the questionnaire tends over represent the input of the fire department when compared with the other traditional parties, it should be noted that the questions about the workload and communication load model focus on the entire emergency response organisation and do not address the separate organisations. This makes it less likely that organisational bias played a role in the answers.

Taken together, these response characteristics provide us with a sufficient basis to use results from the questionnaire for grounding the development of the AWS workload and communication load model.

2.3.3 Grounding the Proof of Concept Simulation

The emergency response exercises presented in sections 2.3.1.1 and 2.1.3.2, helped to ground development of the AWS template. The questionnaire presented in the previous section, in addition, ground the workload and communication load models that are incorporated in the AWS. To test the applicability of the combined AWS and information distributor, a proof of concept simulation will be build, based on information exchange during a real time simulation of a part of an emergency response exercise.

The previous emergency response observations that served the purpose of topic discovery and determining the best suited level of the proof of concept simulation, monitored only a part of the information that was exchanged during the emergency response exercise. For the purpose of topic discovery, monitoring a part of the information exchange was sufficient. However, for the proof of concept simulation the information flow needs to be monitored as completely as possible, since this provides the simulation with the most realistic flow of information that is used to approximate the workflow at the operational level of the emergency response.

As was concluded in section 2.3.1.3, the officer on duty level is the best suited level to be the focus of the proof of concept simulation. The OvD is a central node between the units in the field, receives multidisciplinary information from the other OvD's in the COPI and receives information from other coordinating layers via the H-OvD. Secondly, the complexity of the scenario, the multidisciplinary response, the focus on information exchange and the flexible open ended character of a COPI exercise in which the OvD participates makes it the best suited exercise to monitor for developing the proof of concept simulation.

The data acquisition activities performed to acquire detailed (complete) data concerning the information exchanged at the OvD level that is used for the proof of concept simulation, entailed monitoring all information exchanged to and from the OvDs (fire department, police department and medical services) during a COPI exercise. This was achieved by monitoring the walkie-talkie channels that were used by the officers on duty and by equipping the officers on duty with microphones that recorded all face to face communication during a COPI exercise.

The mock emergency response used, concerned an accident between a passenger train and a flatbed owned by the military carrying a Leopard II tank. The collision took place in the proximity of the central station of Almelo, the Netherlands. As a consequence of the collision, the passenger train derailed and partially fell into a not yet finished railway tunnel next to the temporary track the train was travelling on. The first two compartments were hanging in the construction pit. Complicating the accident was the fact that the train was overly crowded with people and that recently a sound wall alongside the track was placed over a length of 325 meters. The combination of the temporary (narrowed) track with the construction pit on one side and the sound wall on the other side, complicated the self help of the train passengers. Furthermore, given that at the same time of the accident the courthouse in the city of Almelo was the scene of a high profile terrorist court case, the emergency responders were suspicious of possible foul play or a possible terrorist act. This suspicion is fed when a fire fighter finds a suspicious backpack. Similar to the COPI exercise described earlier, the goal of this COPI exercise was to multidisciplinary deal with this situation and share discipline exclusive information of multidisciplinary importance.

Similar to the COPI exercise described in section 2.3.1.2 and the mock emergency response at the operational level described in section 2.3.1.1, the recordings from this COPI exercise were transcribed. The utterances were further divided into information elements that will be used to reveal the flow of information for the proof of concept simulation. During the exercise, lasting 133 minutes, 1837 utterances and 4709 information elements (35 information elements per minute) were identified.

When looking at the flow of information elements throughout the emergency response organisation, this COPI exercise reveals additional information compared with the COPI exercise described in section 2.3.1.2, since also communication was monitored outside the COPI meetings. Corrected for the information elements exchanged during the COPI meetings, this revealed the communication pattern shown in Figure 2.9.

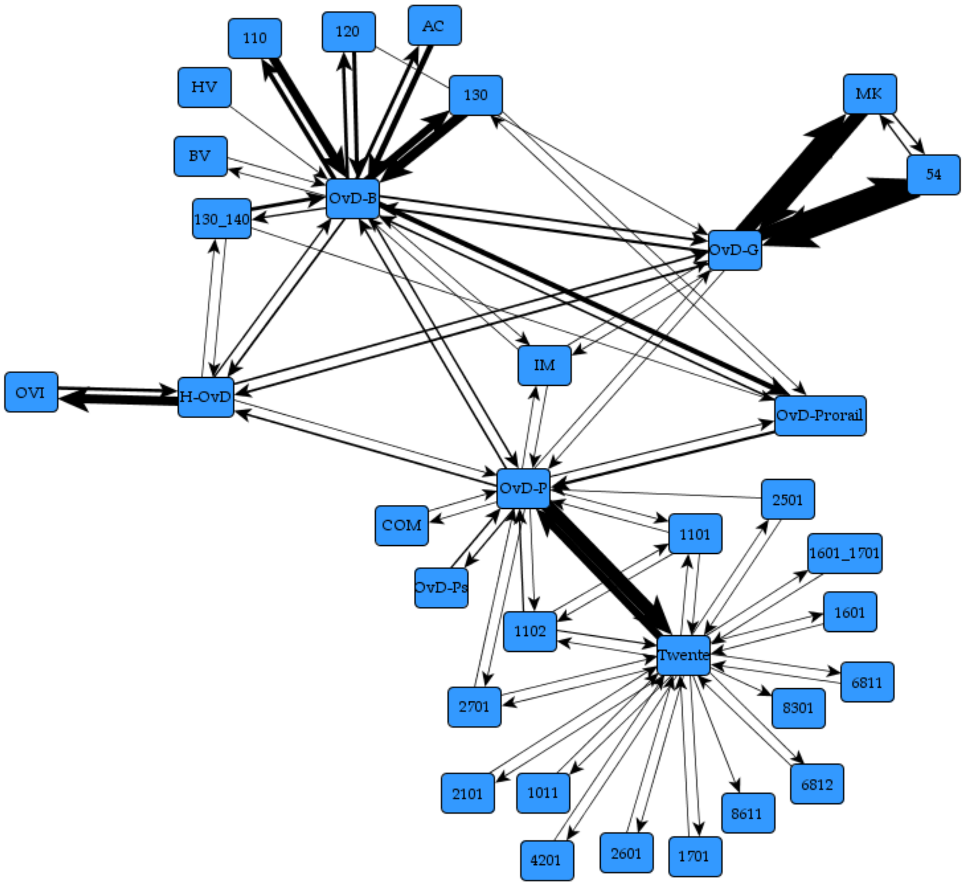


Figure 2.9: *Graphical representation of the flow of information elements outside the COPI meeting room.*

As can be seen in Figure 2.9, three clusters emerge. The first cluster surrounds the officer on duty from the fire department (OvD-B), who within the cluster communicates with the fire department units (110, 120, 130, 140), the dispatcher (AC), the unit commander on the rescue vehicle and finally to groups such as all unit commanders (BV) or a subset (units 130 and140). This is similar to what we saw in Figure 2.4 regarding the flow of information elements during the emergency response to the mock traffic accident. The second cluster surrounds the officer on duty from the fire department (OvD-G), who within the cluster communicates with the first ambulance on the scene (54) and with the dispatcher (MK). The third cluster surrounds the officer on duty from the police department (OvD-P) and the dispatcher from the police department (Twente). The dispatcher communicates with the police units in the field (indicated with numbers surrounding “Twente”). Besides occasional communication with a unit in the field, the OvD-P mainly communicates with the dispatcher.

Furthermore, it can be seen that, since it is a train accident, the officer on duty from Prorail (company responsible for construction, maintenance and safety of the railway infrastructure in the Netherlands) is present. Similar to what we saw in the previous COPI exercise with the AGS, the OvD-Prorail participates in the COPI as a emergency type dependent stakeholder.

To summarize, given that the focus of the COPI meetings and the OvD's is on coordinating the operational emergency response as well as being a liaison between the operational and tactical level (and consequently the higher levels) of the emergency response organisation, the COPI information flow contains sufficiently rich information to approximate the workflow of the emergency responders at the operational level as well as the tactical level. As a consequence, the data from this COPI exercise can provide a good starting point to assess the effects of having, or not having information on the workflow of the emergency response organisation (AWS goal). Furthermore, as can be seen in Figure 2.9, the information flow outside the COPI meeting mainly is directed inward, within the monodisciplinary organisations, clustered around the OvD's. Since the goal of the information distributor (ID) is on assessing the relevancy of information in order to enable relevant information disseminating between these monodisciplinary organisations, the COPI exercise is a well suited test bed. The TAID system as a whole can assess, using these data, the impact of a change in information dissemination on the workflow.

2.4 Conclusion

This chapter described the research methods that are used to acquire data to ground the development of the AWS in emergency response practice and, secondly, acquire empirical data that can be used to test the AWS and the ID with real time data from practice, providing an indication of the possible impact of the TAID system.

By elaborating on the positioning of the AWS system development, it was concluded that the level of detail in the TAID system should be determined by the level of detail that emerges in communication during an emergency response situation. Despite that the TAID systems in essence is a specialized system with the focus on information exchange and workflow, its broad application forces the system to deal with highly detailed information, as well as with information that has less detail, that emerges during emergency response situations. Within the TAID system, therefore, a generic approach is taken by building a generic emergency response template of the reoccurring elements in emergency response.

Given the difficulties in acquiring data, and the limitations of the data that can be acquired from emergency response practice, in order to ground the development of the AWS and provide a test bed for the TAID system, the data acquisition methods used for the development consist of the observation and analysis of emergency response exercises and administering an expert questionnaire.

The template that is developed in the AWS, is grounded by analysing the flow of information elements during two mock emergency response exercises. The first exercise concerned an operational emergency response exercise that dealt with a monodisciplinary emergency in which the fire department had to handle a severe traffic accident. The second emergency response exercise concerned a mock emergency response to a complex multidisciplinary emergency at the COPI level of the emergency response organisation. Besides that these mock emergencies provided data that could be used to help the development of the AWS by providing information about which topics generally emerge in emergency response situations, they also provided information about the level of the emergency response organisation that should be the focus of the final proof of concept simulation that will be presented in Chapter 8. It was concluded that the best focus, given the goals of the TAID system and the AWS, was the officer on duty level.

The data acquired for the proof of concept simulation also consisted of an observation of a mock emergency response exercise at the COPI/OvD level. The detail with which the communication was monitored was higher than during the other two mock emergency response observations. A complete information flow resulting from communication could be created for the officers on duty that were involved in this mock emergency response, which provides a good test bed for the TAID system and the AWS.

Finally, in order to ground the theoretical models that approximate the workload and communication load of the individual emergency responders that are involved in the simulation, a questionnaire was constructed and administered to 26 experts in the field of emergency response, touching upon topics such as workload, communication load, recovery speed, the role of experience and general load contributing factors in emergency response.

This chapter resulted in a decision about the focus of the TAID system and the AWS and detailed how data were acquired to ground the development and testing of the TAID system and the AWS. In the next chapter it will be investigated and decided in which kind of simulation development environment the TAID system has to be build.

3. Emergency Response Modelling and Simulation

In Chapter 1 the general functional layout of the TAID system, the ID and the AWS were presented. In order to develop such a system for emergency response, Chapter 2 elaborated the unique developmental and research considerations that are interlaced with developing and grounding the TAID system and the AWS in practice. Chapter 2 also presented the data acquisition methods that were used. It was observed which problems exist in practice; a general solution with the TAID system was sketched and the methods to acquire data to develop, ground and test the AWS were described. Together these data set the scene for the actual development of the AWS.

However, before we can commence with building the AWS, we first need to determine what type of simulation the AWS is going to be and which simulation environment is going to be used. In this chapter, first an overview of simulation types used in social science research and their application to emergency management is provided (section 3.1). In section 3.2, successively a specification of the requirements for the AWS will be presented and it is determined which simulation environment best fits the goals of the AWS and the TAID system. Section 3.3 will describe and test the selected simulation environment by modelling and simulating a test scenario. Finally, section 3.4 presents the conclusion of the chapter and its consequences for the next chapters.

3.1 Simulation Environments

Simulation environments have been categorized based on: the degree in which they rely on empirical data instead of hypothetical assumptions and the generalizability of their results (Brenner & Werker, 2007); their theoretical basis (Crystal & Ellington, 2004); their application area, usage taxonomy, simulation taxonomy and design taxonomy (Sulistio et al., 2004). When applied to the use of simulation in the social sciences, Gilbert & Troitzsch (2005) point out that simulation types can be categorized using four characteristics: the number of levels that can be modelled in the simulation, the number of agents in a simulation, the complexity of the agents, and if communication between agents is possible or not.

A simulation environment can be capable of representing a number of abstraction levels within the simulation and the interaction between these levels. A level, for example, refers to an agent, organisation, objects or concepts that exist in the environment that is modelled. Interactions between these levels can only exist when two or more levels can be modelled in the simulation environment. Furthermore, in order to simulate emergent behaviour, more than two levels have to be included in the simulation, describing the individual behaviour of an agent or system and collective behaviour of multiple agent or systems. In Table 3.1 the simulation types used in social science are summarized. The number of agents in this table refers to the number of instances that can be modelled within each level, where system dynamics is seen as one agent representing a single system. Furthermore, due to the complexity of the agents that can be modelled in multi agent models, the number of agents is limited. Interaction between and within the levels (Communication between agents) refers to the possibility of the instances within the levels to exchange information with each other. This also is a prerequisite for between and within level interaction of the instances and emergent behaviour. Without communication interaction is not possible. The fourth characteristic refers to the complexity of the behaviour that can be modelled in the instances. When using these distinct characteristics seven distinct simulation types are identified by Gilbert & Troitzsch (2005).

Table 3.1: *Simulation types used in social science (Gilbert & Troitzsch, 2005)*

	Number of levels	Communication between agents	Complexity of agents	Number of agents
System dynamics	1	No	Low	1
Micro simulation	2	No	High	Many
Queuing models	1	No	Low	Many
Multilevel simulation	2+	Maybe	Low	Many
Cellular automata	2	Yes	Low	Many
Multi agent models	2+	Yes	High	Few
Learning models	2+	Maybe	High	Many

System dynamics, which focuses at one level, contains one agent (the system), without modelling communication and has a relative low complexity, is applied in the field of emergency response to, for example, simulate personal information processing resources at the individual responder level (Otto & Belardo, 2006), information system performance simulation (Van den Eede, Muhren, Smals, & van de Walle, 2006) or risk dynamics (Rich, 2006). The use of system dynamics in emergency response is well suited to explore and simulate single isolated phenomena, since it generally describes the behaviour of a single agent or system at a single abstraction level.

Using rules describing individual agent's behaviour, micro simulation simulates the behaviour of the total group of agents and the single agent over time, without communication between agents. Complexities of the rules incorporated in the agents can vary. Within emergency response, micro simulation is used, for example, to simulate evacuations for industrial sites (Georgiadou, Papazoglou, Kiranoudis, & Markatos, 2007), evacuations from rail stations (Kang, 2007), hurricane evacuation procedures (Chen, Meaker, & Zhan, 2006), crowds in general (Henein & White, 2005) or traffic (Shah, Kim, Baek, Chang, & Ahn, 2008). Using rules, one can use micro simulation to predict crowd, firm and traffic behaviour over time in case of an emergency.

Queuing models used in the field of emergency response attempt to, for example, optimize dispatch of ambulances (Iannoni & Morabito, 2006) by simulating space, time and state of the agents and the locations in the model. Queuing models address only one level (i.e., overall dispatch performance), using low complexity rules, not enabling communication between agents, but using many agents. Queuing models thus mostly are used to address complex planning issues using simple rules.

Multilevel simulation allows the modeller to simulate group interactions based on a limited number of attributes which values differ within a population. The population in multilevel modelling are large groups (such as the inhabitants of a country). Multi level simulation environments are applied in the field of sociology and socio dynamics, simulating the emergence and suppression of elements or attributes of large groups. Within emergency response, multilevel simulation can be applied as a macro version of micro simulation, addressing issues as mass panic or other collective phenomena.

Direct interaction between neighbouring actors in a model is used within cellular automata simulations to provide an indication about the spread of information within a single population. Each actor is represented by a single cell in a grid, having only few attributes (alive-dead, on-off, having information or not having information). Depending on the state of its neighbours, the cell's value can change. Inferences can only be made about the group of cells and the values of single cells. Within emergency response, cellular automata is used to, for example, model the spread of fire (Yamamoto, Kokubo, Yamashita, & Nishinari, 2008) or the evacuation of people (Li, Tang & Simpson, 2004).

Similar to micro simulation, multi agent models are able to include fairly complex rules of behaviour between the agents. Multi agent models let the agents react, change and reason about the environment, other agents or objects through communication. Other agents, in turn, react to the changed state. The rules embedded in these agents can be very complex in order for the agents to interact with all elements in their environment. The agents within the model, furthermore, can make wrong inferences about the environment or have beliefs that are outdated. These outdated beliefs can be the cause of (faulty) actions. Multi agent simulations are a situated simulation type, that incorporates the environment and uses distributed intelligence and agent subjective beliefs to model social processes and emergent behaviour. In emergency response, it is used, for example, for modelling human behaviour when evacuating aircraft (Sharma, Singh, & Prakash, 2008), human naturalistic decision making during evacuations (Alavizadeh, Moshiri, & Lucas, 2008) and coordination of disaster management and response (Abramson, Chao, Macker, & Mittu, 2008). Multi agent simulations are able to address multiple levels such as the state of the agent, the environment or objects, but also the state of groups or the organisation.

When a multi agent model is ran twice with the same starting values, its progression in the second run remains the same (given that no random elements are included in the model). Stemming from, and using artificial intelligence, learning models are able to adjust the underlying model to adapt to the data encountered in earlier simulations and are able to progress and evolve to better fit the initial goal. Within emergency response, learning models can be used for anomaly detection (Dominguez, 2003),or information distribution (Netten et al., 2006).

The next section will present the requirements for the AWS and determine the simulation type best fitted to simulate emergency response using the information that is exchanged.

3.2 Simulation Requirements

In the TAID project, the ID uses learning models to derive a model from empirical data which is used to assess the relevance of information and enable the distribution of relevant information to the actors that need that information. In a training phase, the model "learns" how to discriminate between relevant and irrelevant information, and keeps learning when it is confronted with new information or information that either contradicts or confirms the underlying model.

The AWS, on the other hand, simulates emergency response as a function of the information that is exchanged. It situates the emergency responders in the environment. Based on the subjective knowledge that is stored by the actors through communication with other actors, reasoning or interaction with objects, actions or work is initiated. This in turn can change the attributes of the actor or objects.

For the AWS to simulate emergency response, it should therefore be able to model and simulate the dynamics of the emergency setting, the actors, communication, and tasks while abstracting the key properties of emergency response from the complex detail of the emergency itself (i.e., detailed models of fire dynamics). A simulation environment simulating emergency response as a function of information exchange must include the elements on which information is exchanged through communication (see Table 3.2).

The elements that should be incorporated in the AWS are divided into five categories, namely geography, objects, organisation, actors and tasks. These categories were identified by the Dutch advisory commission ICT and emergency management (ACIR, 2006) as the main categories on which information is exchanged during emergency management. As noted, emergency response describes the interplay between the setting and the reaction of the emergency responders. The setting is determined by the geography of the emergency location and the objects and actors (such as victims) that are directly involved in the emergency. The reaction by the emergency responder is described by the objects that are used in the response (such as fire trucks), the actors that are involved with the mitigation (emergency responders), the tasks that the responders perform, the emergency response organisation (number and affiliation of the responders). Communication is the engine in the model, linking all levels.

The elements within these categories can be either semi-static or adaptive, or a combination of both. Semi-static means that information about a category is accessible before the emergency and is independent of the actual emergency, or is not likely to change. Adaptive information is information that can only be accessed during the actual emergency response. These include the variables that are monitored by the emergency responders to give direction to their response.

Table 3.2: *Overview of the semi static and dynamic information categories determining the emergency setting and the response.*

	Geography	Objects	Organisation	Actors	Tasks
Setting	X	X		X	
Response		X	X	X	X
Semi-Static	X	X	X		X
Adaptive	(X)	X	X	X	X

The interplay between these elements shapes the adaptive character of emergency response, deriving elements such as workflow and workload on the fly. The semi static elements are able to form a template-like structure of emergency response. The simulation environment thus should be able to model this abstract ontology of categories in emergency response and their behaviour. The next paragraphs will discuss which elements within these categories have to be incorporated in the AWS to simulate emergency response.

The geography entails the elements within the emergency response environment that refer to the geographical layout and attributes of the environment in which the emergency response is situated. The geographical layout of the emergency response situation contains the locations and distances between the locations which are used for navigation and geo positioning. The geographical attributes consist of location attributes (such as road conditions) and weather related elements, such as temperature, precipitation or wind direction, which determine emergency response conditions, and therefore determine emergency response decision making and workflow. These elements are subjects of conversation during the dispatch phase of the emergency response (Nibra, 2005).

Within the geographical layout, concrete objects reside that play a role in the emergency response. Within the simulation, the term concrete object refers to the non living elements that existing emergency response situations, and reside at a specific location, that can elicit a response by the emergency responder. The attributes of the objects can be affected by actions of the emergency responders or can undergo state changes themselves. An object, for example, can have an attribute “on fire”, with a decrease in severity when an emergency responder is extinguishing it, or an increase when no cooling is provided. Both used (such as phones, cars, fire-hoses etc), and used up objects (such as water, foam, oxygen) can in this way be addressed in the simulation model. Objects’ availability also functions as an enabling or restricting condition for task execution. Driving to an incident location when, for example, no car is available is not possible. The objects that are included in the model and the level of detail in which the objects are modelled, is determined by the communication during the emergency response about these objects. Standard objects that are always available during emergency response - such as standard fire fighting material (Joorse, Slofstra & van Dijke, 1995) - should be available in the template at the class level (with class specific attributes); specific instances can be created when needed.

The geography and the objects jointly build the structure for the locations in which the emergency response is situated, providing locations, objects within these locations and characteristics of and distances between the locations and objects that can adapt to actions by emergency responders. A kind of “ghost town” is created in which multiple simulations can be situated. Due to the generic character, which is based on information exchange, instances of, for example, buildings can be created easily, changing the simulation without having labour intensive 3d modelling consequences, providing high functional fidelity without the need of a high physical fidelity (Hays & Singer, 1988). The focus is thus not on describing how elements within the simulation “look and feel”, but on their properties at the functional (information exchange) modelling level.

Before being able to fill the “ghost town” with instances of emergency responders and other actors, a generic organisational model is needed to structure the flow of organisation specific knowledge that the organisations’ members embody and the roles that these members play during an emergency response. In addition to organisation membership, emergency responders are part of monodisciplinary teams and form a multidisciplinary group of responders for a specific emergency. The organisations’ members, furthermore, are able to make use of organisation specific assets and materials. A specific unit commander, for example, is a member of, and can have knowledge about his unit, a specific fire station, the fire department of a town, a fire department region and furthermore can have multiple roles within these groups. The unit commander knows that specific materials and tools are at his/her disposal; that are owned by the organisations and groups the unit commander is a member of. The roles, and the positions a person holds, furthermore, determines the activities the responder is engaged in during a response. The simulation thus has to be able to handle multi role and multi group membership to organize the flow of knowledge, task obligations and assets of the emergency responders involved in the response.

Individual differences between actors' knowledge and task obligations during emergency response, for a large part, are due to group memberships. These memberships and task obligations do most likely not change during the actual response. The tasks that are performed, and the setting in which these tasks are performed, however, do add differentiating elements between emergency responders during the response, affecting their workflow. Unlike differences caused by group membership, these elements in general are attributes shared by all actors involved in the emergency response that adaptively change for a specific agent and depend on the workflow and the situation the responder is faced with. Examples of these actor specific differentiating elements are the physiological state (task restricting injuries), psychological state (workload) and knowledge restrictions (not having situational knowledge). The simulation model therefore has to be able to handle specific on the fly emerging actor states.

Instances (actors) of emergency response groups that have basic knowledge about their organisation(s) and their role(s) receive information and act on the information by engaging in tasks. Task execution affects their physiological, psychological and knowledge state, which in turn affects their workflow. Beside the elements within the emergency response environment and the actors' knowledge, tasks are the final essential element, describing the actual interplay between the setting and the emergency responders. Task execution is the consequence of the situation, and influences the situation through it. To be able to achieve such an adaptive level in a simulation, the simulation has to incorporate the tasks themselves, task enabling conditions, task consequences and an on the fly adaptive workflow. Task execution is knowledge and resource dependent, confining execution to those actors who possess the correct combination of knowledge and roles, have access to the right material and are in the correct situation to exhibit the task behaviour. Within the simulation environment it isn't necessary to create new tasks for the actors. As long as tasks are defined at the proper level of granularity that fits the needed detail, tasks can be seen as generic and reoccurring in most emergencies (de Muralt, 2007; SAVE & Adviesbureau van Dijke, 2000). As a consequence, the sequence of task execution is more likely to change than the generic tasks themselves.

With the addition of tasks, the simulation now is able to construct the setting and the response of the emergency responders, but remains silent, since a special type of task, communication, is not present. Communication between actors; between objects and actors and the use of communication devices must be incorporated. In actor – actor communication, both verbal and non-verbal communication play an important role in emergency communication. Actor - object communication involves, for example, entering a report sheet into a computer. Object – actor communication can be seen as a warning signal from a monitoring device that can be interpreted by the actor. Finally, object – object communication can be seen as the linkage of two computers in a network. Communication provides the simulation with the final piece of the puzzle, enabling it to simulate the interplay between the emergency response situation and the emergency responders with the use of communication.

The description of the categories that make up the AWS, requires it to simulate the environment in which the emergency response takes place, the objects and actors and organisations that are involved in the response, the tasks of the responders and the communication between the responders. The AWS thus has to incorporate multiple levels to facilitate emerging behaviour; has to facilitate communication between agents, objects and environment; has to incorporate fairly complex rules to describe work practice and the consequences of actions; and, finally, has to incorporate a generalized number of actors and objects that are involved in emergency response situations.

The requirements for the AWS, based on the description of the categories that have to be incorporated in the AWS to simulate emergency response as a function of the information that is exchanged, is presented in Table 3.3.

Table 3.3: *AWS requirements*

Model requirements	
- Geography	- Tasks
- Locations	- General
- Distances	- Linked to Roles
- Attributes	- Enabling conditions
- Climatologic	- Task consequences
	- Communication
- Objects	- Actor – Actor
- Location bound	- Actor – Object
- Autonomous behaviour	- Object – Object
- Attributes	- Verbal
- Used (up)	- Non verbal / observation
- Availability	
	Simulation Requirements
- Organisation	
- Organisation specific knowledge	- Low fidelity
- (Multiple) Roles	- Multiple levels (2+)
- (Multiple) Membership	- Communication between agents
- (Group) Assets	- High agent complexity
	- Relatively few agents
- Actors	
- Physical characteristics	
- Injury	
- Psychological characteristics	
- Workload	
- Communication load	
- Subjective knowledge	

Based on Tables 3.1 and 3.3, the simulation type that best fits these requirements is a multi agent simulation, which allows using multiple levels to facilitate emerging behaviour, allows communication between agents (objects, environment, and actors), allows the modeller to generate complex agent models, and is able to handle a sufficient number of agents. A multi agent system thus is able to simulate emergency response as a function of information, to provide information about the tasks, workload and communication load of the responders.

A multi agent environment focusing on modelling and simulating work practice is the Brahms multi agent modelling and simulating environment (Clancey, Sachs, Sierhuis, van Hoof, 1998). The Brahms environment has been and is used to simulate work practice of astronauts on the moon (Sierhuis, 2000), the activities of astronauts onboard the international space station (ISS) (Acquisti, Sierhuis, Clancey, & Bradshaw, 2002), and space shuttle mission operations (Sierhuis, Diegelman, Seah, Shalin, Clancey, & Selvin, 2007).

However, the Brahms modelling and simulation environment has not been applied to the domain of emergency response. The following sections, therefore, will explore and test if the Brahms multi agent modelling and simulation environment is really suited to simulate emergency response as a function of the information that is exchanged and derive information about tasks, workload and communication load.

3.3 The Brahms Simulation Environment

The Brahms modelling and simulation environment was developed to support the design of work by describing not the formal elements of how work should be done, but by focusing on how work actually is done. The tool incorporates multiple views, such as people, information, systems and geography. It has its roots in object oriented and other agent based languages, which shows in the tool's use of an ontology-like, hierarchical structure that can both describe the common characteristics and differences between objects, agents and locations in an efficient way. For example, tasks that are performed by the emergency services are hierarchically structured in such a way that people in subgroups inherit the properties from the parent group (see Figure 3.1). As a result, the tasks of the main disaster organisation apply to all services it branches into. The specific tasks for the fire departments supplement these basic tasks. The following branch, in turn, inherits the fire departments tasks and the general tasks, supplementing them with their own group defining tasks. The further the branch reaches, the more specific the task becomes.

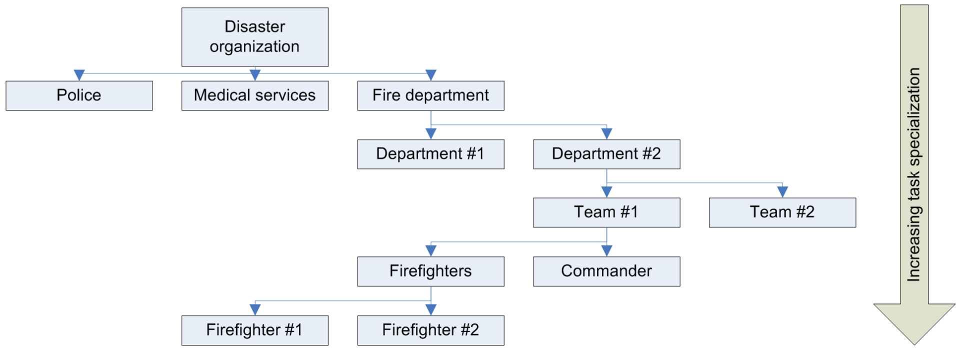


Figure 3.1: *Snippet from the hierarchical structure of a disaster organisation.*

This hierarchical approach enables the modeller to simply and quickly apply changes to the model, affecting all subgroups. It further resolves the problem of having to define all characteristics for a new group that enters the organisation. In disaster response this is especially valuable since this enables the model to adjust to fast changing organisation characteristics or resources. Furthermore, it provides the opportunity to make a disaster response ”template”, where general disaster response is pre specified high in the hierarchy, and minor adjustments at a lower level, making the model easily applicable to completely different disaster situations. Using the taxonomy shown in Figure 3.2, a world is created as a hierarchically structured modelling formalism, consisting of agents, objects, beliefs and facts, activities, and geography. In the simulation environment these concepts interact on the basis of beliefs agents hold (representation of information).

Groups contain
- agents who are located and have
- beliefs that lead them to engage in
- activities that are specified by
  - workframes that consist of
  - preconditions of beliefs that lead to
  - actions, consisting of
  - communication actions
  - movement actions
  - primitive actions
  - other composite activities
  - consequences of new beliefs and world facts
  - thoughtframes that consist of
    - preconditions and
    - consequences

Figure 3.2: *Brahms Taxonomy*

Agents, and groups of agents represent the people and the organisations involved in the mitigation process. Agents are structured in groups based on similar tasks, similar organisational type or competencies. Figure 3.1 shows that fire-fighter #1 and #2 belong to the same group of fire-fighters, and belong to team #1. This implies that, in absence of differentiating characteristics, they normally work on the same type of tasks defined in their organisations' protocol.

Objects and groups of objects (classes) represent all objects that can be used by the emergency personnel during the mitigation process (such as fire trucks and stretchers), but also all objects that are located in the disaster environment that may hinder the mitigation (such as obstacles and fires). It consists of objects that can be used once and objects that can be used repeatedly. Objects are grouped hierarchically.

Beliefs and facts are representations of respectively subjective and objective representations of the world by agents and objects. This includes the agents and objects themselves. Facts are pieces of information that are not private to the owner of the information and can be "seen" by everyone who is in the same location as the agent or the object. An example of a fact is organisational membership of a particular agent. This can be communicated by an agent wearing a particular uniform, without actually actively communicating it. Beliefs form the "inner thoughts" of agents and objects, which can only be extracted by interacting with them. Agents and objects can have beliefs about anything in the world, and they can be inconsistent with the facts in the world. An agent, for example, can possess the belief that the colour of the car that caused a particular accident is red (belief of the agent), while in fact its colour is grey (fact of the object). Beliefs thus are subjective representations of the world, and can, under certain circumstances, be biased. Beliefs are the trigger for the activities performed by an agent.

Activities and work frames, respectively, represent tasks and series of tasks performed by an agent. Just as in normal life, people engage in activities based on beliefs that they have about the world. When someone feels hungry (belief), this person will probably start eating (activity) shortly afterwards. In emergency situations, hunger is replaced by information about the emergency or disaster, triggering tasks or a series of tasks and protocols. Individual differences caused by the absence of information or presence of faulty information, can thus have an effect on the tasks that are performed by the agent. Information processing biases -caused by information overload- can also be modelled this way, by filtering the information that becomes a belief and triggers a task or a series of tasks. Activities are linked to agents and objects and are organized hierarchically.

Geography represents the spatial relation between agents and objects. It provides locations where agents and objects can be situated and move around in (such as a country, an office or a car). Geographical relations between locations are expressed using the time it takes to move from one location to another. The relative distance can be defined by, for example, the mode of transport or obstacles on the road and can differ for all the agents or objects.

The concepts mentioned above, are entered in the Brahms modelling environment. Figure 3.3 shows its main window in which the sub models, defining geography, agents and object are shown. The hierarchical way of structuring these concepts can clearly be seen. The Netherlands contains the town of Enschede, which, in turn, contains several fire departments, and a disaster area. After all concepts are entered in the model, the simulation can be activated.

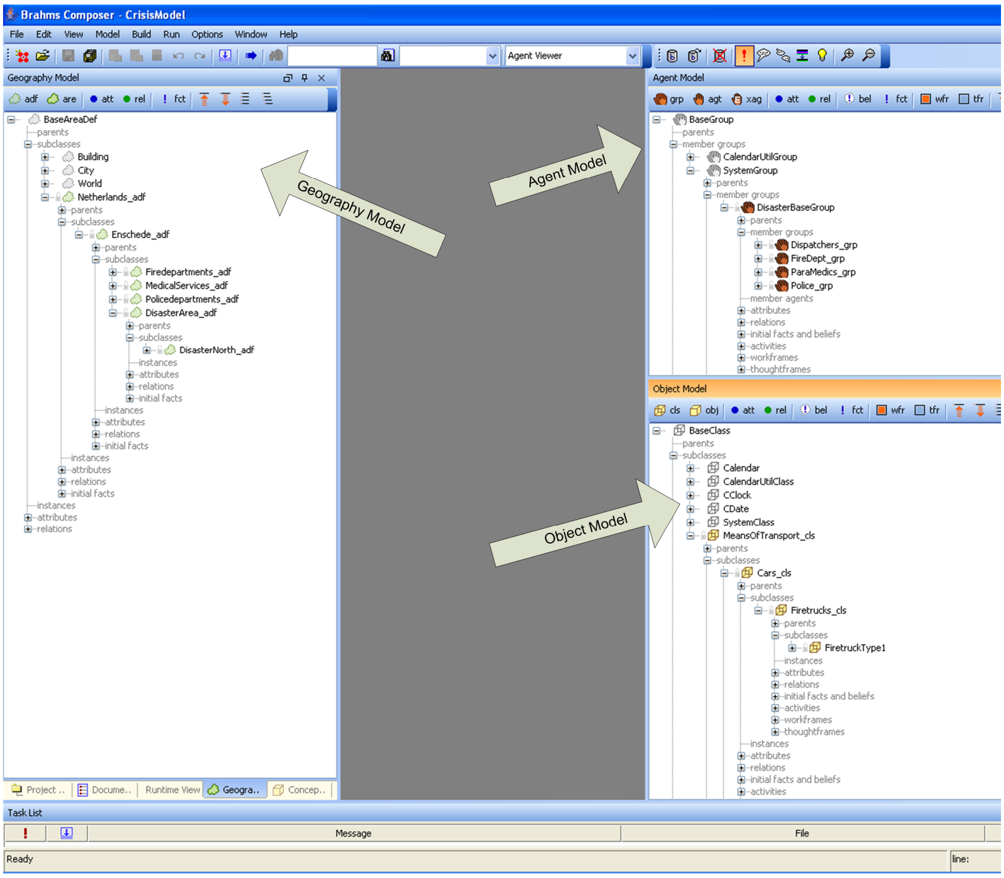


Figure 3.3: *Brahms modelling environment*

Returning to the requirements which minimally must be met to effectively model and simulate disaster response for work and protocol optimization, it can be concluded that the Brahms environment is able to this. Locations can be entered in the geography model. Distances between these locations can be modelled by either constructing direct paths using time as a measure for distance, or by variable travelling time indicators that influence travelling time. The latter enables modelling of relative distance between two locations. The locations are able to adapt to changes in the environment by, for example, becoming off limits to everyone, and sending out information that, for example, a building has collapsed.

The object model makes it possible to model both durable and non durable, objects (by assigning an adaptive belief about its quantity to the object). Furthermore, objects can be modelled in such a way that only agents with a belief containing “knowledge” about the use of the object are able to use or move it.

With the use of beliefs, modelling of basic life characteristics (such as breathing, walking), is made possible, and are able to change according to circumstances.

Concerning the communication aspects of the model, all types of communication are possible using all types of communication equipment. For example, a restriction on the number of active communication devices at one time makes it possible to limit the communication infrastructure's capabilities, leading to loss of information or wrong messages. Communication errors during the transmission and acceptance of the message can be incorporated by equipping the receiver with certain information processing biases, or by not incorporating certain beliefs (with the consequence of not triggering a particular task).

The hierarchical modelling structure enables protocols to be divided in subtasks and series of tasks. These parts can be addressed by the emergency personnel without having to use the entire protocol, resulting in efficient restructuring of the workflow, depending on the needs of that particular moment. This way of organizing tasks, enables normal tasks (basic life tasks) such as breathing to be easily modelled at a high level in the hierarchy, being applicable to all people in the simulation.

The remaining requirements (the use of events and adaptability), are also met by the modelling environment. Events can be initiated by an "event agent" that at a certain moment in the simulation fires a belief, thus changing the environment, which in turn fires certain tasks with agents. Events can occur randomly or at specified times, thus testing the rescheduling abilities of the agents and the protocols in certain situations.

3.4 Brahms Test Model of Emergency Response

As illustrated in the previous paragraphs, adaptability of the setting, actors, communication, tasks and events in the Brahms modelling environment theoretically can be achieved.. To test this claim, this section will describe the results from a trial implementation of a modelling disaster response during the Hercules disaster. Based on this trial implementation, a final decision can be taken about Brahms as the to be used modelling environment.

During its landing, a Belgian Hercules military aircraft crashed at an army airbase in Eindhoven, the Netherlands, while carrying 37 members of the Royal Dutch Army brass band and a crew of four. The initial crash itself caused none of the passengers or crew serious harm. However, due to the kerosene fire that followed, in combination with errors made during the disaster response, 34 persons died and the remaining 7 were seriously wounded. The 27 investigation reports that were published later, showed that the disaster organisation failed on several points:

- lack of clarity about protocols and task delegation;
- inefficient collaboration between the airbase fire department and the Eindhoven fire department;
- insufficient resources (human and material) to cope with the rescue;
- far from optimal use of available information.

Furthermore, the mitigation during the Hercules disaster was faced with difficulties due to:

- discrepancies between beliefs and facts;
- activation of multiple protocols;
- collaboration between multiple organisations;
- delay caused by misinterpretation of non verbal signals;
- inefficient collaboration;
- communication errors.

Based on the transcriptions from the actual disaster, the Hercules disaster was reconstructed using Brahms, entering agents, objects, locations, activities, communication and beliefs to the model. Figure 3.4 shows a sample of the initial beliefs and facts of the airbase fire department. Initial, refers to the start state of the simulation. These facts and beliefs refer to:

- knowledge about who uses certain radio frequencies ([…]GebruiktPortofoonKanaal[…])
- knowledge about own membership (current.ismemberof[…])
- knowledge about others and their membership of organisations ([…]isMemberOf[…])
- knowledge about the organisational structure ([…]hasBevelvoerder[…], […]hasOSC[…])

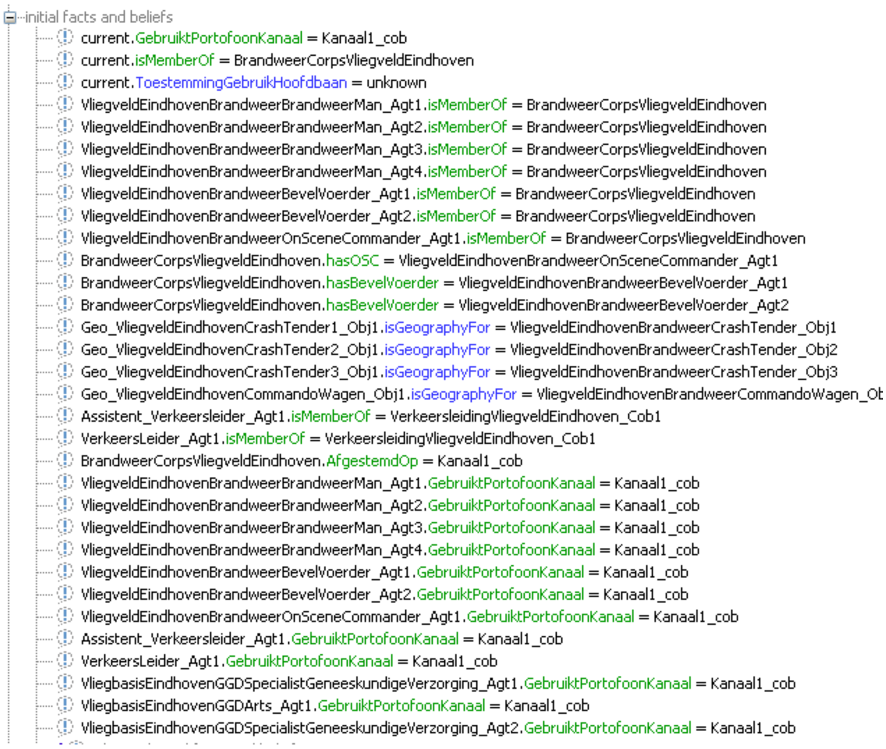


Figure 3.4: *Sample from the airbase fire-fighters' initial beliefs and facts*

These facts and beliefs can be replaced during the simulation by more up-to-date beliefs. When, for example, a switch is made to another radio frequency, and one of the fire-fighters hears this, his belief is changed by the new information. However, it is also possible that this information does not reach one of the fire-fighters. Then, the belief is not changed immediately, but will be if he asks someone else, or someone else tells him. The impact of delay of information can be operationalized this way.

Tasks carried out during the Initial response by the fire-fighters are shown in Figure 3.5. Since these tasks are shared by all airbase fire-fighters, they are quite general. For example, the task ‘GaNaarAccidentLocation’ (move to accident location) consists of: 1.)‘Instappen’ (get in the car), assuming that the car and the fire-fighters are in the same building; 2.)’WegRijden’ (drive); 3.)’Uitstappen’ (get out of the car); 4.)’WerkenTijdensRijden’(working while driving). The latter includes multitasking activities such as ‘getting in gear’ or ‘calling’ The preconditions for initiating the task of going to an accident location are:

- the On-scene Commander must have clearance from the tower to use the runway;
- the fire-fighters must be aware of an accident;
- the accident must occur in their district;
- the accident location must be known;
- the fire-fighters must know which car to take, and what their role is in the organisation.

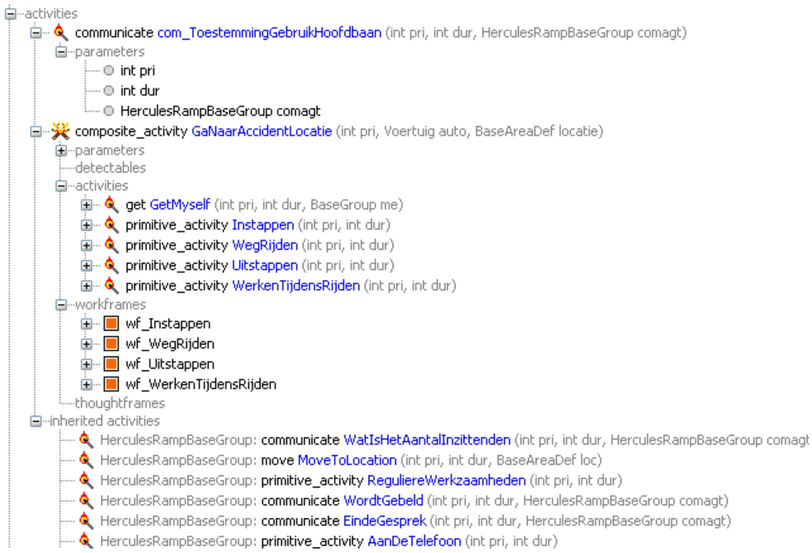


Figure 3.5: *Tasks concerning the initial response by the airbase fire department*

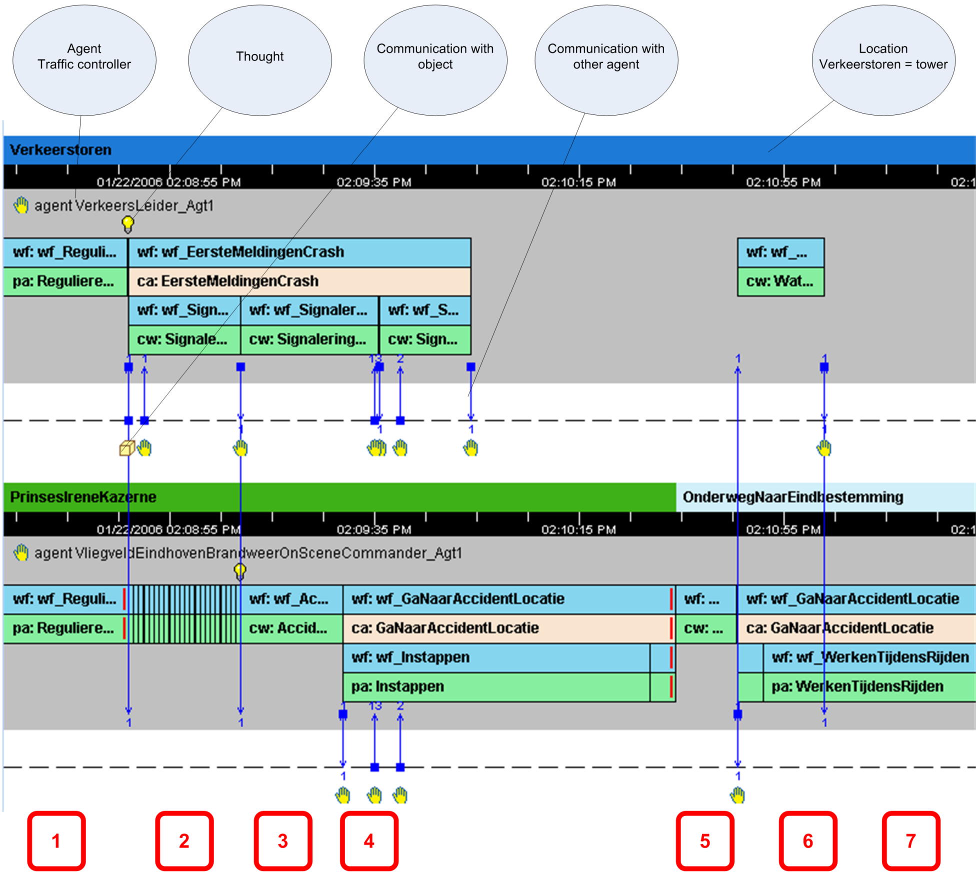


Figure 3.6. *Interaction and workflow of the traffic controller and the on-scene commander*

After having entered the characteristics of the Hercules disaster into the Brahms modelling environment, it is entered in the Brahms simulation environment. This runs the simulation and returns a bar like workflow diagram for all agents and objects in the model. As is shown in Figure 3.6, the simulation results include information about the name of the agent or object, the location and agents' thoughts. On the basis of the interaction and workflow between the on-scene commander and the traffic controller shown in Figure 3.6, the output in Figure 3.6 will be explained using the numbers 1 to 7 at the bottom of that figure. In Figure 3.6 agents are shown as , objects are shown as , thoughts are shown as and communication is shown as . To illustrate its use, for only the first two numbers these symbols are incorporated in the text

1.) At the start of the simulation the on-scene commander and the traffic controller are having a normal working day, doing normal work (Reguliere werkzaamheden).

2.) The traffic controller sees that the Hercules plane () is in trouble, and gets confirmation from the pilot (). At that moment he is aware of the accident (). He initiates the crash protocol, calling first the on-scene commander ( ). At the end of the call the on-scene commander is aware of the accident ( ).

3.) The traffic controller then calls the airbase medical services. Parallel to this, the on-scene commander informs the air base fire-fighters, who in turn are aware of the accident.

4.) The traffic controller calls the airbase dispatcher and orders to call the emergency services with a request for ambulances. The on-scene commander gets into his car and awaits an answer from the tower to a question from another fire-fighter for permission to drive on the runway.

5.) After permission is granted, the fire trucks leave the fire station. The on-scene commander asks the traffic controller if he knows if the Hercules was carrying passengers, and if so, how many.

6.) The traffic controller says, using the radio, that the number of passengers is not known to him.

7.) The on-scene commander continues working in the car until he reaches the disaster area.

The power of the simulation lies in the fact that tasks are initiated by information that reaches the agent or object. The on-scene commander will not initiate the protocol if he isn't aware of the accident. Furthermore, task duration can be random (between boundaries of a minimum and a maximum specified duration). In practice, a task never takes the same time to complete every time. The simulation of the Hercules disaster thus let us conclude that modelling disaster response using the Brahms modelling and simulating environment can work., Furthermore, it illustrates that low fidelity tools, such as Brahms, can effectively simulate work practice in disaster response, without the need to develop costly visualizations that are not essential for coming to grips with the basic processes that constitute an emergency.

Summarizing, using a set of requirements that should be met for simulation of emergency response (Table 3.3), it can be concluded that the Brahms simulation language can effectively simulate work practice in dynamic environments.

Workflows can be restructured to adapt to the changing circumstances of the disaster, which makes it possible to answer "what if" questions that can arise. The low fidelity of the way disaster response was simulated using Brahms, furthermore, doesn't cause the simulation of work practice to be less accurate. This way, workflow in disaster response can be structured, manipulated and investigated in an efficient, cost effective way.

The results from the simulation of the Hercules disaster response show that Brahms can be used for protocol and workflow modelling. The hierarchical method of structuring the elements in Brahms (such as agents, objects, tasks) greatly simplifies the applicability to other disaster situations without the necessity to build a completely new model from scratch, increasing the modelling efficiency.

3.5 Conclusion

This chapter addressed the question which modelling and simulation method is best suited to model the Adaptive Workflow Simulator, in order to let it be a generic adaptive simulation of emergency response that operates as a function of information exchanged.

To answer this question, first an overview was provided of simulation types and their characteristics that are used in the social sciences and emergency response research. Based on the requirements that should be met by the AWS, it turned out that a multi agent based simulation would best fit its purposes, since it allows using multiple levels to facilitate emerging behaviour, allows communication between agents (objects, environment, and actors), allows the modeller to generate complex agent models, and is able to handle a sufficient number of agents. A multi agent system thus is able to simulate emergency response as a function of information to provide information about the tasks, workload and communication load of the responders.

The Brahms modelling and simulation environment was chosen to model a test simulation of an emergency response to a plane crash that occurred in the past. Based on this simulation, it was concluded that the Brahms modelling and simulation environment made an efficient and valid representation of the emergency response possible.

As a consequence, the Brahms modelling and simulation environment will be used to develop the AWS and will serve as the testing environment. In the next chapter, the hierarchal approach from the Brahms modelling and simulation environment will be used to its full potential to provide a generic reusable template of emergency response that can be used and reused for modelling emergency response situations.

4. Emergency Response Template

The previous chapter presented the model- and simulation requirements that have to be met by a simulation environment for it to be suitable to simulate emergency response as a function of the information that is exchanged. By modelling and simulating the Hercules plane crash, it was demonstrated that the Brahms modelling and simulation environment meets these requirements. With respect to the model requirements presented in Table 3.3, the object oriented approach that is incorporated in Brahms provides us with the opportunity to model emergency response at a generic level, as well as at a more detailed level, depending on the level of detail derived from the focus of the simulation.

Following the object oriented approach, the generic model (at the class level) of emergency response consists of the reoccurring elements, where, as detail increases (at the lower classes and the object and agent level) the model incorporates the elements that are unique for a particular emergency response. Furthermore, given that objects and agents inherit the activities, attributes and attribute values from their direct and indirect parents, this approach provides us with the opportunity to construct an inheritance based hierarchal template of the elements involved in an emergency and the concurrent emergency response.

This chapter will present the generic model -or template- of emergency response which acts as the main modelling architecture for the AWS emergency response simulation; a reusable template consisting of the elements that are shared between different emergency responses. Emergency specific details and deviations are added when a particular scenario is modelled (for example, the proof of concept simulation in Chapter 8). The major benefit of the template approach is that it allows the modeller to focus on the elements of interest in the emergency response model and be supported by a predefined inheritance structure of the classes and instances in the model. As a consequence, this will likely decrease modelling time for variations of the same scenario and decrease modelling time when developing new emergency response scenarios. Figure 4.1 illustrates the concept of inheritance. It shows the direct and indirect classes from which a pump-water-tender (`PWT_FDAlmelo_Centre_obj`) of the fire department of Almelo (the Netherlands) that is stationed at fire station Almelo Centre, inherits its attributes and attribute values. Within all of these classes, activities, attributes and attribute values that are unique for that particular class are defined. These activities, attributes and attribute values are then inherited by its children. General (constant) characteristics of an object are defined at a high class level, while object specific characteristics are defined at a lower class level. A more detailed characterization of the object thus emerges at the lower levels.

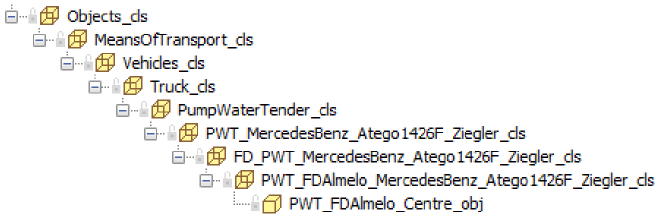


Figure 4.1: *Direct and indirect parent classes of a pump-water-tender stationed at the fire station Almelo centre in the Netherlands.*

The pump-water-tender stationed at the Almelo centre fire station (`PWT_FDAlmelo_Centre_obj`) is a pump-water-tender (`PumpWaterTender_cls`) of a particular type and build (`PWT_MercedesBenz_Atego1426F_Ziegler_cls`) that is owned by the fire department of Almelo (`PWT_FDAlmelo_MercedesBenz_Atego1426F_Ziegler_cls`). This pump-water-tender shares attributes with similar pump-water-tenders owned by other fire departments in the Netherlands (`FD_PWT_MercedesBenz_Atego1426F_Ziegler_cls`) and is a type of truck (belonging to the `Truck_cls`). A truck can be classified as a type of vehicle (`Vehicles_cls`), which is a means of transport (`MeansOfTransport_cls`). Finally, all means of transport can be classified as objects (`Objects_cls`). The `PWT_FDAlmelo_Centre_obj` inherits attributes and attribute values from all of these parent classes. Other pump-water-tenders that are owned by other fire departments or are of a different type and build, automatically inherit other activities, attributes and attribute values, distinguishing them from other pump-water-tenders, but share general characteristics of pump-water-tenders, trucks, vehicles, means of transport and objects.

Figure 4.2 shows the inherited attributes (inherited attributes and relations) and attribute values (inherited initial facts and beliefs) for the pump-water-tender that is stationed at Almelo centre fire station (`PWT_FDAlmelo_Centre_obj`).

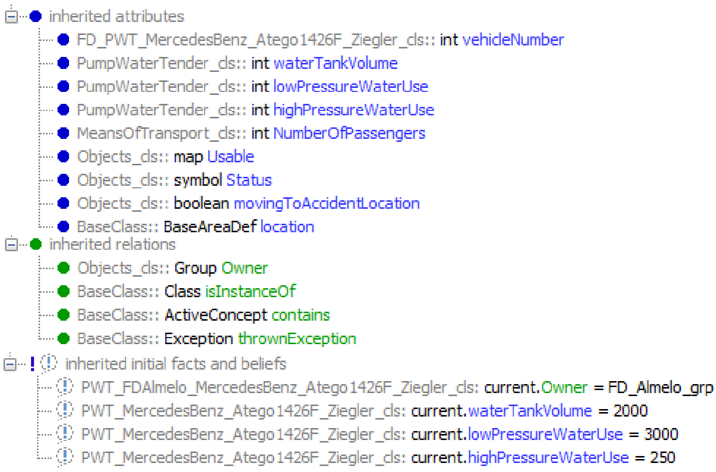


Figure 4.2: *Inherited attributes and attribute values for PWT_FDAlmelo_Centre_obj.*

Since the `PWT_FDAlmelo_Centre_obj` has the "means of transport" class as an indirect parent, it automatically inherits the ability to carry a number of passengers (since all means of transport have this ability). This is expressed by the line `PumpWaterTender_cls::int NumberOfPassengers` in Figure 4.2, meaning that the "number of passengers" value is expressed using an integer (indicated by `int`) and the attribute is inherited from the `PumpWaterTender_cls`. In a similar manner, the pump-water-tender that is stationed at Almelo Centre fire station is able to have attributes about: the size of its water tank; water usage when the pumps are operated at low or high pressure; can have a vehicle number; is suitable for certain types of use; has an operational status; is moving to the accident location; has a current location; has an owner; and can contain objects (i.e., the objects in the standard packing list (Projectgroep Modificatie Bepakkingslijsten, 2008)) or agents. The template structure thus predefines a large set of the characteristics about an object without having to redo this for all specific instances.

However, when unique details about a specific pump-water-tender are needed to achieve the goals derived from the focus of the simulation, these can be specified at the instance level. The pump-water-tender `PWT_FDAlmelo_Centre_obj`, for example, has a unique vehicle number (number 3131) that is used for identification purposes. As mentioned earlier, a more detailed model of a specific object or agent thus emerges when moving towards the lower end of the inheritance hierarchy. Although creating a highly detailed model increases the richness and possibilities of the template, it also greatly increases the effort that has to be put into constructing and maintaining the model in order to keep it up to date.

Returning to the discussion on system development issues presented in Chapter 2 (generic versus specialized systems, and the level of detail needed in a simulation, a balance has to be found concerning the level of detail needed in the template model for it to be complete, but not overly complete, without compromising the goals and the focus of the simulation. Since the focus of the AWS is to simulate emergency response as a function of the information that is exchanged, the level of detail within the template model thus can be reduced to the level on which is communicated, decreasing the number of attributes and states of an object or agent to general attributes that appear in the literature and more detailed attributes used in communication. Grounding the AWS template model in emergency response practice with the combined use of literature and empirical data from actual communication thus is beneficial for balancing the level of detail incorporated in the AWS template. The empirical grounding that is used in this chapter, is based on communication during emergency response exercises at the operational level (during a traffic accident exercise) and the COPI level (during a major fire incident exercise) of emergency response. A more detailed description of these exercises and the data collection and analysis methods used was given in section 2.3.1.

The research (sub)questions that will be addressed in this chapter are:

- How can we construct a grounded AWS template that is able to function as a reusable template for emergency response modelling in Brahms?
  - Which classes have to be included in the AWS template for it to be a model of the reoccurring elements in emergency response?
  - How can the reoccurring elements be ordered using the inheritance structure?
  - What level of detail should the model have?

The upcoming sections will use the hierarchal approach to construct a grounded descriptive model of the reoccurring elements in emergency response that minimally should be incorporated in the AWS. Incorporating the emergency response model requirements presented in Table 3.3, first the organisation, role and task model will be presented in section 4.1. Next, the agent model will be presented (in section 4.2) followed by the object model (in section 4.3) and the geography model (in section 4.4). Finally, section 4.5 will present the concluding remarks.

4.1 The AWS Emergency Response Organisational Template

The type, severity and prospects of an emergency determine the composition of the emergency response organisation. Following the principle of "scaling up", the emergency response organisation starts relatively small but adaptively scales up or down given the situation at hand. Scaling up in this respect entails that more of the same units, specialized units or coordinating layers are added to the organisation, where scaling down refers to downsizing the emergency response organisation. However, despite the emergency dependent fast changing composition of the emergency response organisation, the organisations that are involved, the roles that exist in these organisations and the tasks that the members of the organisations can (and will) perform, are relatively stable throughout different emergency responses. During an emergency response it is not likely that new organisations will emerge -besides already known facilitating organisations such as coach companies or large stakeholders such as a power company- that actively will take part in the response. Furthermore, it is not likely that, apart from minor variations, new major roles or tasks emerge in the emergency response organisation besides the ones already defined. The unique character of the emergency response is defined by the order in which predefined tasks are executed by actors from known organisations that have a predefined role in the emergency response organisation.

This stability at a higher level than the "actual emergency responder in the field", provides us with the opportunity to build a template of the total emergency response organisation that propagates down to the emergency responder in the field. By doing this, the AWS automatically provides a modelled agent with the attributes and tasks that belong to its organisation and role. This section describes the organisational template model used in the AWS: the organisations involved in emergency response in section 4.1.1 and the roles and tasks that can be fulfilled by emergency responders in section 4.1.2.

4.1.1 Emergency Response Organisational Structure

Depending on the emergency, the emergency response organisation consists of members from the fire department, police department, medical services, local government, regional government, national government and the public prosecutor that are organized in multidisciplinary teams. Figure 4.3 provides an overview of the chain of command and the different multidisciplinary teams that can be active during an emergency response situation in the Netherlands.

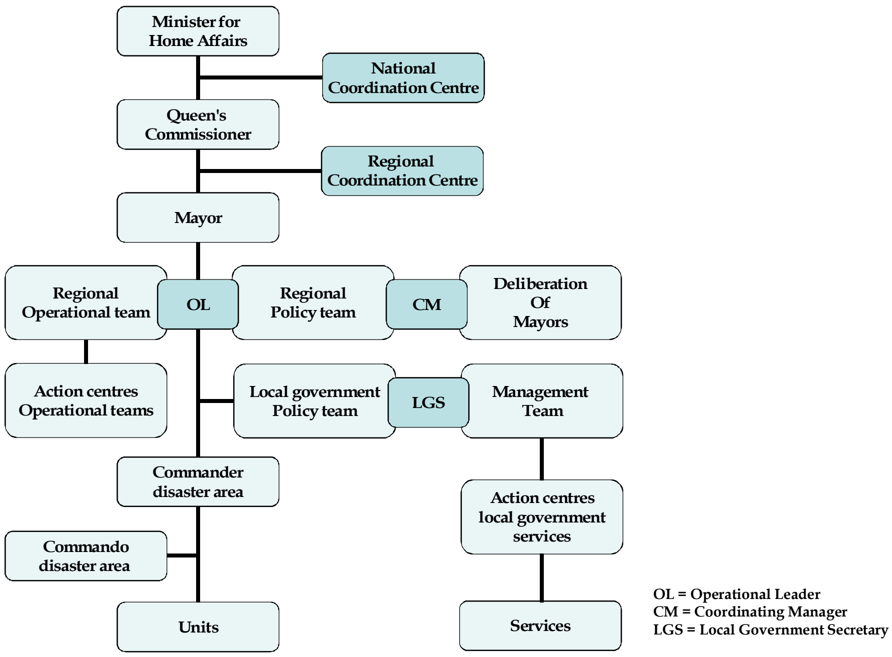


Figure 4.3: *Organisational structure of the emergency response organisation (based on the emergency Response organisational structure as presented by the Dutch Ministry of the interior and kingdom relations in the Handbook for disaster management preparation (2003, p A3-2)).*

Following the GRIP (which stands for coordinated emergency response procedure) for scaling up, it is determined (based on the impact of the emergency) which parts of the emergency response organisation are, or will become active. Within this procedure, five compositions of the emergency response organisation are defined (see Table 4.1).

Table 4.1: *Relation between the composition of the emergency response organisation, the impact of the emergency and the GRIP scaling up procedure.*

GRIP	Impact of the Emergency	Emergency Response Organisation
0	Minor	Units
1	Source area only	Units + COPI
2	Source and effect area	Units + COPI + ROT
3	Endangerment of large groups of people	Units + COPI + ROT + GBT
4	Effecting multiple municipalities or shortage of means	Units + COPI + ROT + RBT

When an emergency can be dealt with monodisciplinary (such as a small kitchen fire), coordination of the emergency response is handled by the units involved (GRIP 0 phase). When the fire intensifies and coordinated effort is needed, but the effects of the emergency are limited to the source's location, the multidisciplinary "Commando Disaster Area" (COPI) comes into existence (GRIP 1 phase). The COPI consists of members from the fire department, police and medical services. If the smoke (a consequence of the initial fire) from the fire affects the directly surrounding area (the effect area) and measures have to be taken to deal with these consequences, the multidisciplinary "Regional Operational Team" (ROT) will also become active (GRIP 2 phase). Coordination between these teams is done by the "Commander disaster area" who resides at the emergency location and the "Operational Leader" which often resides at the regional coordination centre.

If the kitchen fire were to grow into a large building fire that threatens the safety and health of large groups of people within a municipality, the "Local Government Policy Team" (GBT) is added to the emergency response organisation to coordinate the effort from the municipality (GRIP 3 phase). When, furthermore, smoke drifts across the municipality border or the municipality doesn't have the means to handle the emergency, more than one municipality will become involved in the emergency response organisation (GRIP 4 phase). Their efforts will then be coordinated by the multidisciplinary "Regional Policy Team" (RBT).

In order to facilitate reasoning and communication about the dynamic organisational structure of the emergency response organisation, the AWS template firstly has to provide the opportunity for the agents to reason about attributes of the emergency itself (since this determines the organisational structure) and, secondly, has to facilitate multi group membership, since agents can both be a member of their own emergency response discipline and be a member of multidisciplinary teams and units, each with specific goals, knowledge and resources. The attributes of the emergency are presented in section 4.1.1.1, while the AWS organisational template describing the organisations involved in the emergency response using the principle of multiple group membership are presented in section 4.1.1.2.

4.1.1.1 Attributes of the Emergency

As stated in the first sentence of section 4.1, the type, severity and prospects of an emergency determine the size and the composition of the emergency response organisation. During an emergency response it is therefore likely that the attributes of the emergency are subject of communication between emergency responders in order to come to grips with the situation one is faced with and, consequently, tailor the emergency response organisation to the emergency. However, to make reasoning about, and reasoning with these attributes in the AWS possible, these attributes of the emergency have to be included in the AWS template.

Within the AWS template, the emergency concept is seen as the overall impression of the emergency response situation, summarizing the general attributes of the emergency and the response to it. In order to determine which attributes of the emergency have to be included in the AWS template to sufficiently represent the emergency concept, the communication between emergency responders from two emergency response exercises described in section 2.3.1 were analyzed on the presence of these attributes.

From the 170 distinct topics that were extracted from the communication during both emergency response exercises (with a total of 2478 information elements being exchanged), 40 topics included information elements that related to general attributes of the emergency. Given the fact that within the topic clusters of duplicate information elements could be present, information elements within these topics were further analyzed and clustered by similarity to reveal the general attributes of the emergency. This resulted in 10 main emergency response attribute categories shown in Table 4.2.

Table 4.2: *Attribute categories observed during emergency response exercises.*

Attribute categories	Explanation	Examples (translated from Dutch)
Emergency type	what kind of emergency is the emergency response organisation faced with	- "Dispatcher, you called in a traffic accident" - "Dispatcher, update, very large accident, multiple vehicles involved"
Emergency response location	where is the emergency situated	- "6278, can you turn out to an accident, with entrapped victims on the provincial road N347 between Goor and Lochem, close to hectometre marker 69,2"
Effect area	Status of the indirect consequences of the emergency	- "So we haven't got an effect area where acute health problems may surface" - "The biggest problem we have is the soot is settling everywhere in an area of about 1,5 to 2 kilometres in length."
Source area	status of (parts) of the emergency at the source	- "the fire in the fourth vehicle has been put out" - "Dispatcher, update, accident under control" - "Source area is effect area"
Victims	total number, condition and location of victims	- "Two victims have been moved to the hospital" - "T3 victims can go to the sports centre at the laan van oostindiegangers" - "Up to this point we have got 10 victims"
GRIP	GRIP status of the emergency	- "The current GRIP status is one" - "We just got a message from the officer on duty that he called out a GRIP 3 status"
Assembly of the emergency response organisation	Present assembly (people / material) of the emergency response organisation	- "Ambulance is present" - "In total four pump-water-tenders are on the scene"
Prospect of the emergency	most likely path that the emergency will follow	- "How long do people from the fire department expect it will take" - "Fire will most likely be confined to the basement and maximally will jump to the building above"
Emergency response procedures	Current active emergency response procedure	- "we made a decision, and we will follow the emergency plan; lpg filling station scenario 2"
Needed services / assistance	Material or people needed to mitigate the emergency	- "So I want to utilize the services of a fire department company and the WTS1000"

Based on these observations, an emergency can be described as a particular type that emerges in a location on which it has a direct influence (i.e., victims, buildings) and can have an indirect influence on other locations (effect area). The emergency response organisation consequently uses the values of these attributes to determine the GRIP status of the emergency and their own organisational arrangements. Once formed, the emergency response organisation will follow certain work procedures (fitting the type of emergency) and will determine if the current configuration is sufficient to mitigate the actual scenario and likely future scenarios. For the AWS to represent the communication about these general aspects of the emergency, these attributes of the emergency should be included.

However, simply constructing an emergency class and providing it with the general attributes will not do justice to the diversity that exists in the communication between emergency responders about the attributes of the emergency. First, within the general attribute categories, many sub-attributes may exist that account for the variation within these categories. As, for example, can be seen in the explanation of the "victims category" in Table 4.2, one cannot describe all victims of the emergency with a single value since this will not cover the variation of victims (i.e., severity, number). Secondly, communication may lead to the assignment of different attribute values to the same attribute (i.e., by using absolute numbers or by subjective measures such as a "large amount") that all are used in reasoning and communication. These subjective semantic representations often are used in reasoning when no detailed information is available (which is frequently the case at the start of an emergency response). Thirdly, despite that all types of emergencies are very likely to share the general attributes, emergency type specific attributes also exist (such as water level when dealing with a flooding) which are exclusively used for that type of emergency. Finally, the emergency may be composed of multiple emergency types (such as a building fire *and* a traffic accident) leading to combinations of attributes' values originating from multiple emergencies types. The AWS template thus needs to provide the opportunity to model the general attribute categories, variations that exist within these categories, multiple value assignment, emergency type specific attributes and emergencies consisting of multiple emergency types.

Figure 4.4 shows the AWS template model representing the attributes of the emergency based on the general attribute categories and by taking the remarks mentioned above into account. Within the parent emergency class (`Emergency_coc`), 18 distinct classes reside that correspond with the 18 distinct emergency types that were identified by SAVE & Adviesbureau van Dijke (2000). Within these classes, attributes can be defined that are unique for that particular emergency type, providing the opportunity to model emergency specific attributes. In addition to the 18 emergency type classes, one extra class is added that covers the `CurrentEmergency_cob` object, representing information about the current emergency. Since the `CurrentEmergency_coc` is able to inherit attributes from one or any combination of more than one emergency type classes, the AWS is able to deal with the inheritance of attributes when modelling emergencies consisting of different emergency types.

The general attributes of the emergency –based on the observed attribute categories from the two empirical emergency response exercises- are defined in the `Emergency_coc` class. These attributes propagate down to all 18 emergency types mentioned above and the current emergency class. Except for the GRIP attribute (which is defined as an integer), all general attributes of the emergency are defined as map attributes. A map attribute is a collection type that allows the assignment of multiple values to a single attribute. An example of an agent's beliefs about the total number and severity of victims (the triage attributes) within the emergency, using map attributes is shown in Figure 4.5.

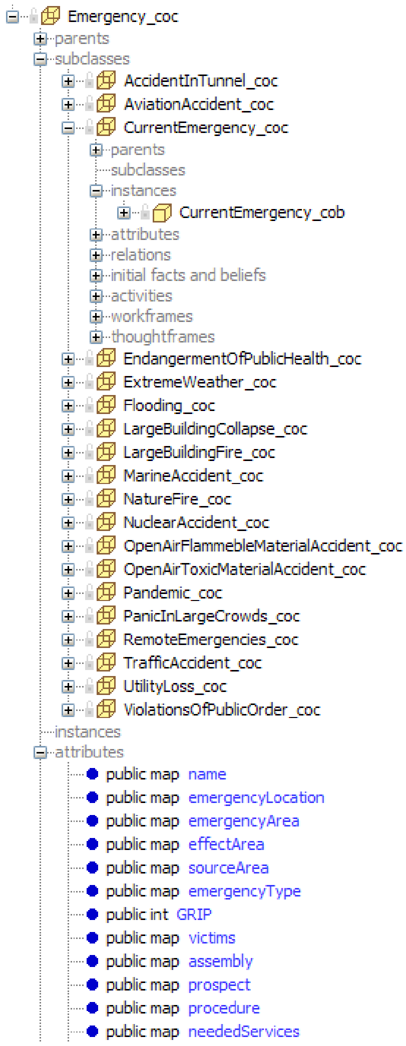


Figure 4.4: *Emergency types and attributes in AWS template.*

```
(CurrentEmergency_cob.victims("Number T1") = 5);
(CurrentEmergency_cob.victims("Number T2") = 23);
(TrafficAccident_cob.victims("Number T1") = 2);
(TrafficAccident_cob.victims("Number T2") = 6);
(LargeBuildingFire_cob.victims("Number T3") = 152);
(CurrentEmergency_cob.victims("Cluster T1") = "Almelo_Ambtstraat_IntersSchoolstraat_are");
(CurrentEmergency_cob.victims("Number T3") = 152);
(CurrentEmergency_cob.victims("Semantic Number T3") = "large amount");
```

Figure 4.5: *Agent beliefs containing map attribute values about victims.*

In Figure 4.5, it can be seen that the agent has information about the total number of T1 victims (5 unstable victims that need immediate medical attention), T2 victims (23 victims that need to be treated within six hours) and T3 victims (153 stable but injured victim). Furthermore, the agent has information about the origin of these victims (152 T3 victims caused by the building fire, 2 T1 and 6 T2 victims caused by the traffic accident). Finally, the agent has knowledge about the location of a cluster of T1 victims located at the intersection of the Ambtstraat and the Schoolstraat in Almelo the Netherlands. The map attribute, furthermore, provides us with the opportunity to assign attributes such as “Semantic number T3” representing a semantic representation of the number of T3 victims.

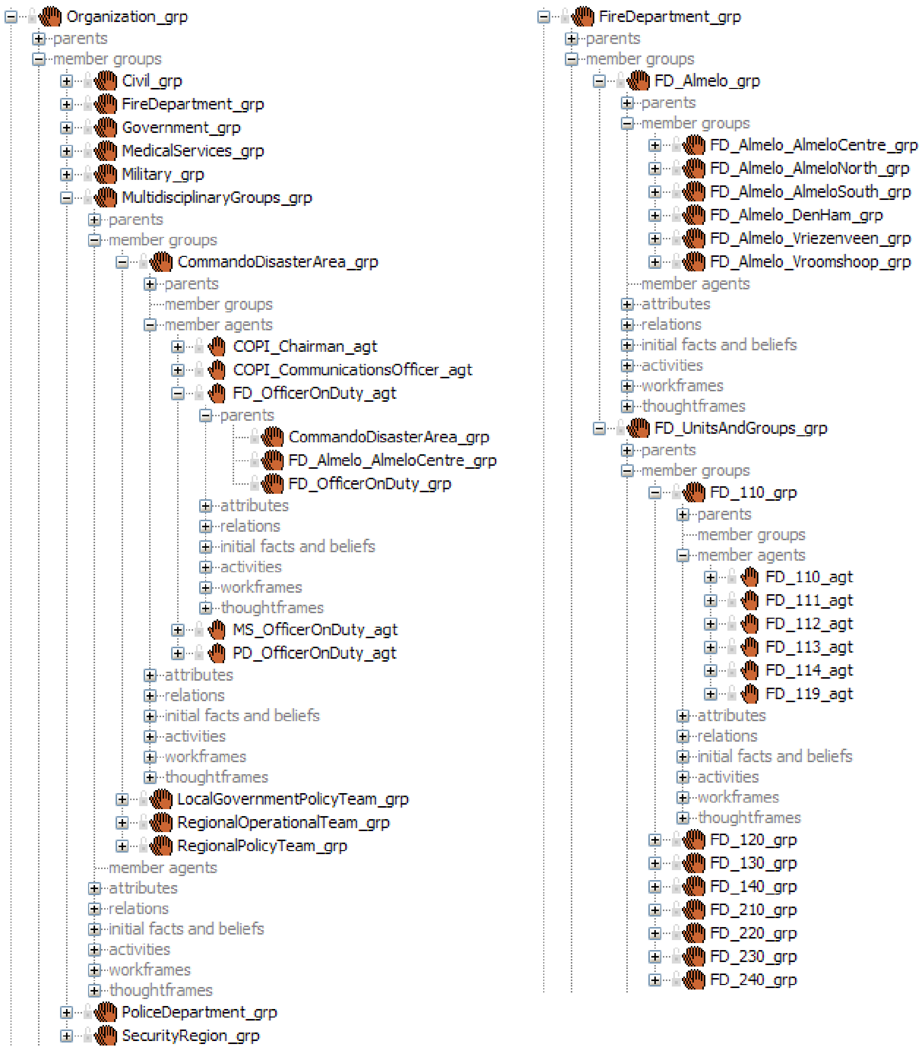
With the use of map attributes, the AWS template can model the general attribute categories of the emergency described in Table 4.2, as well as the variation that may exist in these categories in actual emergencies. Moreover, it can handle the assignment of multiple values (i.e., semantic, absolute). Using multiple class membership and the inheritance structure, the AWS also handles information exchange that is unique for a particular emergency or for emergencies that consist of one or multiple emergency types. Using this representation of the emergency, the AWS template thus enables the modeller to define the attributes of the emergency, and model the behaviour of agents so they can react to different types of emergencies (i.e., protocol activation).

4.1.1.2 Organisational Model and Multiple Group Membership

As was indicated in the previous section, the attributes of the emergency determine the size and the composition of the emergency response organisation. The emergency response organisation consists of representatives from multiple organisations performing roles and tasks in their own organisation, as well as in multidisciplinary groups and units. Besides being a member of their own organisation, an emergency responder also is a member of the multidisciplinary group, of the group of emergency responders involved in the response and of a group of people performing the same role. Each of these groups can have group specific information, tasks and goals. Thus, in order to define the emergency response organisation, multiple group membership is of key importance.

This section presents the organisational model representing the emergency response organisation that consists of multiple sub-organisation and teams. The different roles and tasks that are performed during emergency response are presented in section 4.1.2.

Figure 4.4 illustrates how multiple group membership within the emergency response organisation (a) and the fire department (b) is incorporated in the AWS template.



(a) Emergency response organisation (b) Fire department group

Figure 4.4: *Snippet from the AWS emergency response organisational template showing multiple group membership.*

As can be seen in Figure 4.4(a), the different organisations that are involved in the emergency response all are a child of the `organization_grp`. This overall group contains groups representing the standard parties involved in emergency response; the `PoliceDepartment_grp`, the `FireDepartment_grp` and the `MedicalServices_grp`. Furthermore, since members from these groups also can be an active or passive member of the security region, the `SecurityRegion_grp` is modelled. The template furthermore provides the opportunity to model civil groups (such as large companies having their own specialized emergency response equipment), divisions of the (local, regional and national) government and the military organisation. Finally, the multidisciplinary groups are represented by the `MultidisciplinaryGroups_grp`.

When unfolding the `MultidisciplinaryGroups_grp`, it can be seen that it subsumes the `LocalGovernmentPolicyTeam_grp`, the `RegionalOperationalTeam_grp`, the `RegionalPolicyTeam_grp` and the `CommandoDisasterArea_grp`. Further unfolding of the last group, shows that it always consist of a chairman (which is a member of either the police department, fire department or the medical services), a communications officer, and representatives from the police department, fire department and the medical services. Looking more closely at the representative from the fire department, it can be seen that this agent is a member of the `CommandoDisasterAera_grp`, a particular fire station (in this case `FD_Almelo_AlmeloCentre_grp`) that in turn is a child of the `FD_Almelo_grp` (see Figure 4.4 (b)), and finally that this particular agent has the role of officer on duty for the fire department (see section 4.1.2 for a further elaboration of the "role groups"). Figure 4.5 shows the multiple group inherited beliefs of the officer on duty.

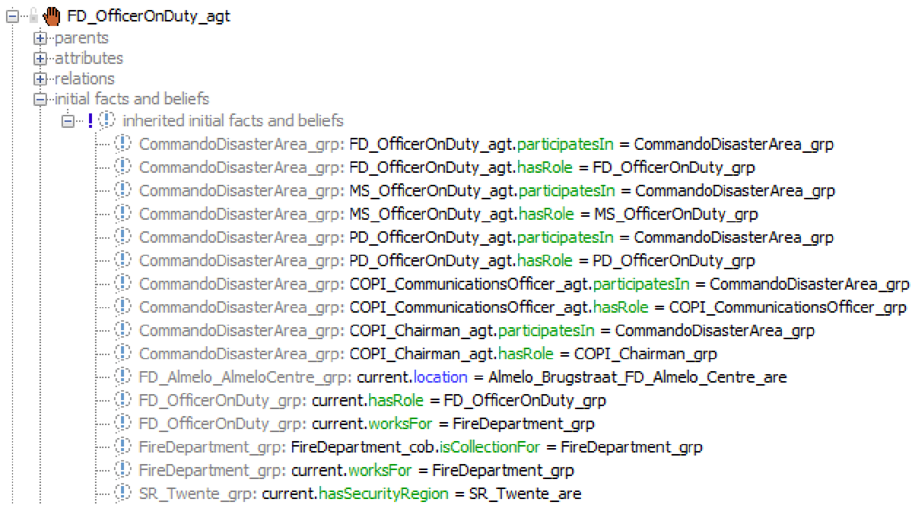


Figure 4.5: *Inherited beliefs due to multiple group membership for the Officer on Duty during the emergency response.*

The template automatically provides the officer on duty with knowledge about the composition of, and the roles within the commando disaster area group. Furthermore, in this example, the officer on duty inherits knowledge about his security region and his starting location at the beginning of the simulation. Adding a group membership in Brahms is done by adding a group to the "memberof" list of an agent. Figure 4.6 shows the code needed to achieve the inherited beliefs presented in Figure 4.5.

```
agent FD_OfficerOnDuty_agt memberof CommandoDisasterArea_grp, FD_OfficerOnDuty_grp, FD_Almelo_AlmeloCentre_grp
}
```

Figure 4.6: *Code needed for multiple group membership for the officer on duty in Brahms*

Returning to the group representing the fire department where multiple group membership within an organisation can be demonstrated, Figure 4.4 (b) illustrates multiple group membership for the `FireDepartment_grp`. All members of the fire department can be narrowed down to a particular fire department, which in turn can be narrowed down to a fire station, inheriting attributes and activities from all these groups. The fire stations, in turn, supply the units to deal with specific segment of the emergency response. During all emergency responses, fire department units (pump-water-tenders) are named based upon the arrival sequence (independent of the fire department membership). The first unit on the scene is referred to as the 110, the second as the 120, the third as the 130 and the fourth as the 140. Since four units form a platoon, the second platoon (units 5 to 8 based on arrival) is referred to by the numbers 210 to 240. The firemen that accompany a unit have particular roles within the unit and are referred to by using a number that is linked to the unit name. The members of a unit thus are members of a particular fire station/department and perform a specific role in the unit.

However, not all information elements that are exchanged concern *individual* agents. Table 4.3 displays the agent and group related topics that account for 60% of the information elements that are exchanged during the emergency response exercises. As can be seen, a substantial part of the total communication of information elements that concerned agents and groups, makes reference to *general groups* of agents (such as COPI, Fire Department, Emergency Responder, Unit 110, Police Department) and attributes of groups instead of attributes of individual emergency responders that make up the group. Communication about groups accounts for 52% of all agent and organisation related information elements that are communicated. Generally speaking, these communications concern abstract references to a certain group; do not concern all its members; and furthermore it is not known to which specific group members it refers. An example of such a reference regarding the police is “the police is surveying the area”. What we know is that someone from the police is surveying the area. However, it remains unclear who is doing it.

Table 4.3: *Top 60% of the agent and organisation related topics extracted from the emergency response exercises.*

Topic	Total
COPI	44
H-OvD	43
AC	33
AGS	32
OvD-B	31
Fire Department	31
OVI	30
Emergency Responders	29
Victim	28
110	28
Police Department	27
Total	356

If it was clear that Agent#1 was surveying the area, this communication could be modelled by sending the belief `Agent#1.currentActivity(“surveying the area”) = true`. However, since Brahms cannot make reference to groups, the organisational and group structure is rebuild as a conceptual class that has similar attributes as the original groups. By doing this an agent can communicate `PoliceDepartment_cob.currentActivity(“surveying the area”) = true`, making an abstract group reference to the police department activity, without making reference to a single agent performing the activity.

Summarizing, using multiple group membership, the AWS template thus provides the opportunity to easily model agents in multidisciplinary groups and multirole groups within multidisciplinary teams and monodisciplinary organisations. The inheritance structure automatically provides the agents with organisation or group specific beliefs that can be used for reasoning or communication for the individual emergency responder. Furthermore, by using a parallel organisation inheritance structure in the conceptual object class, communication about abstract references to organisations can be modelled in the AWS.

4.1.2 Roles and Activities

In general, one can state that organisations consist of people that perform activities related to the role they have within the organisation. Roles, in this respect, structure the activities that are performed, and can lead to role exclusive knowledge about, for example, procedures and goals. This section will present the part of the AWS template that structures the activities and roles within the AWS. However, before we can start, it has to be clear what is understood by roles and activities within the context of the AWS.

As Shea (1981) points out, activities within an organisation can be structured using different levels of work, structured in a hierarchy. By describing the secretarial job, he points out that jobs consist of duties, that consists of tasks, that consist of elements of tasks, that consist of motions. A job refers to the total collection of duties and tasks an individual performs. An example of a job within the emergency response organisation is a fire department unit commander. The associated duties concern a large segment of work that structurally occupies a significant portion of the employees time. When applied to the unit commander, a duty for example can be responding to an emergency situation. Next, tasks concern units of work activity that form a significant and consistent part of the duty. The duty of responding to an emergency situation includes tasks such as monitoring the pager system for emergencies, getting into the pump-water-tender and moving to emergency location. When moving to the location, different tasks are performed by the unit commander, such as obtaining extra information from the dispatcher, informing the unit about the turn-out, and determining the initial plan of attack. The task of getting extra information can, for example, be broken down into getting the walkie-talkie, getting in contact with the dispatcher, ask information about weather conditions, ask for information about possible victims. Shea (1981) refers to these activities as elements; a further detailed description of a task without referring to actual motions, movements or cognitive processes. When elements are broken down to the lowest level -the motion level- one describes the actual physical or mental acts. For the element of using the walkie-talkie, motions include raising the arm to the shoulder, pressing the activate bottom with the hand that is positioned at the shoulder, hold the button, say name of receiver, say “here”, say unit number, release activate button, etcetera.

The role that an emergency responder has in the emergency response organisation can best be described as the job of that emergency responder, describing the total of duties and tasks. However, other than the jobs described by Shea (1981), roles in the emergency response organisation are short lived, given the temporary character of the emergency response organisation. Strictly speaking, one could argue that the role of the emergency responder could be described as a duty; a significant part of work that structurally occupies a significant portion of the employees time. However, for the development of the AWS template the perspective is taken of the emergency response organisation and not of the organisation in which the emergency responder operates in his day to day routine. A role therefore is seen as the emergency bound job of an emergency responder.

Within the context of the AWS, an activity in theory can originate from all levels of work (except the job level) described by Shea (1981). However, firstly, the level of detail of the activities is determined by the focus of the simulation. Given that this focus is on simulating general emergency response and providing general task information to the information distributer, it is very unlikely that activities from the element and motion level will add value to the simulation. Secondly, the level of detail of the activities within the AWS is determined by the level of detail in which is communicated about activities at the unit commander and COPI level. Apart from detailed communication about motions when, for example, it has to be explained how a neck brace has to be put on, it is very unlikely that this kind of detailed communication concerning the motions and elements will take place at the COPI and unit commander level. Within the AWS, an activity therefore is defined at the duty and task level, thus covering those parts of the work that are in the scope of the simulation.

As was shown in section 4.1.1, during emergency response more of the same or different groups, units or coordinating layers are adaptively added to the organisation to respond to the specific need of the emergency, each performing their role in the total emergency response organisation. Using the emergency response organisation structure as is shown in Figure 4.3, during a GRIP 4 situation minimally 46 distinct roles can be identified belonging to agents involved in the emergency response (see Table 4.4). This list, however, is the minimal list, given that other parties that are involved may join the emergency response organisation.

Table 4.4: Selection of groups and roles in the Dutch emergency response organisation.

Group	Roles
Fire department	- Fire department unit commander, Attack squad, Water squad, Driver-pump-operator
Police department	- Police car driver, Police motor driver, Police officer
Medical services	- Ambulance driver, Ambulance nurse, First ambulance driver, First ambulance nurse, Mobile medical team doctor, Mobile medical team nurse, SIGMA leader, SIGMA member, SIGMA driver, Wounded transport assistant, Wounded transport coordinator, Head casualty collection point
COPI	- Officers on duty (from fire department, police department, medical services), Leader COPI, COPI communications officer
Operational Team	- Operational team staff members (from fire department, police department, medical services, local government), Operational team communications officer, Leader operational Team
Local policy team	- Mayor, Public prosecutor, Local government policy team members (from fire department, police department, medical services, local government), Local government policy team communications officer
Regional policy team	- Coordinating official, Mayors, Public prosecutor, Commander regional fire department, Regional medical services coordinator, Police superintendant, Member local government policy team, Regional policy team communications officer

The roles shown in Table 4.4 were implemented in the AWS template. Figure 4.6 displays a snippet of the roles within the emergency response organisation incorporated in the AWS.

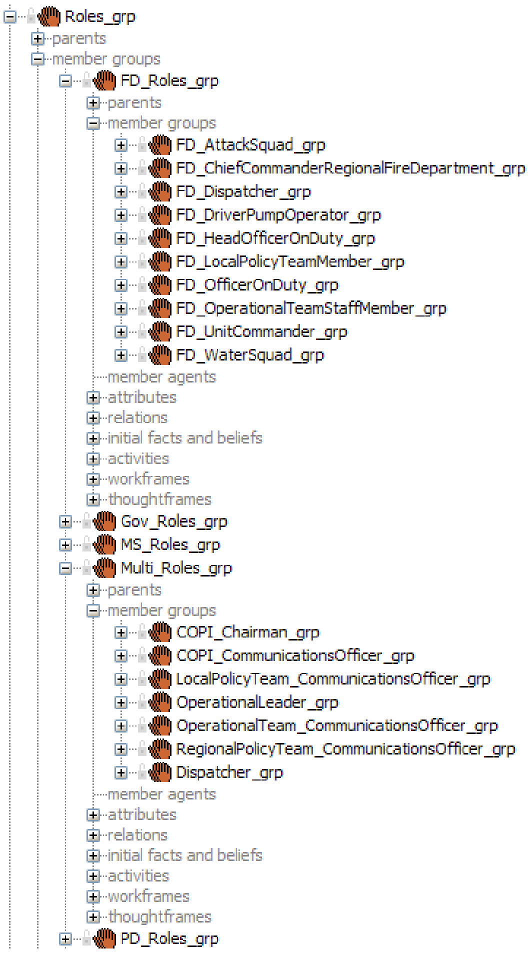


Figure 4.6: *Snippet of the role model of emergency response in the AWS.*

Except for the `Multi_Roles_grp`, that contains the roles that can be performed by members from all standard disciplines involved in the emergency response (fire department, police department, medical services), roles are clustered by the organisations that provide the emergency responders that perform that particular role. `FD_Roles_grp` contains the roles typically performed by members of the fire department (such as the officer on duty of the fire department role `FD_OfficerOnDuty_grp`), `Gov_roles_grp` contains the roles performed by the government, `MS_Roles_grp` contains the roles of the medical services and `PD_Roles_grp` the roles of the police department. Within each of the role groups, attributes and attribute values that are unique for a particular role can be defined to structure role specific knowledge and activities of the agents that are instances of these roles.

In Figure 4.7 an example is shown how attributes defined in the role group are used to structure the activities of the emergency responders in the simulation using role based preconditions.

```
workframe wf_cac_moveToSafeHavenLocation {
   display: "Evacuate unit from dangerous area";
   when(knownval(current.currentTask("Evacuate unit from dangerous area") = true) and
        knownval(current hasRole FD_UnitCommander_grp))
         do {
            cac_moveToSafeHavenLocation();
            conclude ((current.previousTask("Evacuate unit from dangerous area") = true), bc:100, fc:0);
            conclude ((current.currentTask("Evacuate unit from dangerous area") = false),bc:100, fc:0);
         } // close do
      } //close workframe wf_pac_moveToSafeHavenLocation
```

Figure 4.7: *Role restriction in the preconditions of a workframe.*

Figure 4.7 shows the workframe that is used to trigger the `"evacuate unit from dangerous area"` activity (`cac_moveToSafeHavenLocation()`). This activity is triggered when two enabling conditions are satisfied. The first condition (`current.currentTask("evacuate unit from dangerous area") = true`) refers to the belief that in order for a responder to evacuate its unit, the emergency responder first has to come up with the necessity that this should be an activity that has to be performed. This belief can either be the consequence of reasoning -the responder can conclude that evacuation of the unit is needed based on information that reaches him- or it can be communicated to the responder by others. All agents in the model can derive this belief, given the right available information. The order to evacuate, however, should only come from an agent who is a unit commander. This is expressed in the second enabling condition: `current hasRole FD_UnitCommander_grp`. Since the unit commander is a member of the `FD_UnitCommander_grp` role group, the unit commander automatically inherits the `current hasRole FD_UnitCommander_grp` belief, satisfying the second enabling condition thus triggering the `"evacuate unit from dangerous area"` activity. By specifying the roles and the corresponding role belief, the AWS is able to structure task execution, making it a function of the roles of the agents.

Which activities are defined in the AWS model depends on the AWS activity model. In the AWS template, activities are defined in a separate group from which the `Roles_grp` and the `Organization_grp` are children: the `ActivityTemplate_grp`. Within this group, all general activities (such as moving from A to B) and role specific activities (such as securing the water supply to the pump-water-tender) are modelled. In theory, all agents that exist in the model thus are able to perform all activities in the model. However, by using the role attributes mentioned earlier, restrictions are put on the actual firing of the activities. By defining all activities in one group however, the opportunity is created to create flexibility in the assignment of roles, including possible multiple roles due to absence.

Summarizing, by using roles, the AWS is able to structure role specific knowledge and activities, while also maintaining a flexible structure which provides the opportunity to model multiple role assignments to the agents that perform the activities within the AWS. Chapter 5 will go further into the aspects of the order in which the triggered activities will be executed.

4.2 Agents

While reading above about the complex organisational structures, roles and activities, one could almost forget that the focus of the AWS is on the agents within the model and their beliefs. However, by specifying most of the agents' attributes and attribute values at the organisational and role levels, an agent can simply be modelled by placing it in a combination of groups (organisation, or role). To complicate matters, non work related attributes - not specified in the organisation or role group- exist that may influence the agents behaviour or will be used to communicate about. As specified in the model requirement in Table 3.3, these include physical characteristics, identifying characteristics and psychological characteristics. The presence of physical characteristics enables the agent to, for example, get injuries and enabling communication about the physical state of agents. Identifying characteristics describe the personal particulars (such as name, home address and age). Finally, psychological characteristics provide the opportunity to monitor elements that cannot be observed directly during emergencies or emergency response exercises (such as workload and communication load).

Since workload and communication load are central concepts in this thesis, the development and implementation of the workload and communication load in the AWS is described in separate chapters: Chapter 6 presents the AWS workload model and Chapter 7 presents the AWS communication load model. This section is limited to the presentation and implementation of the physical and identifying attributes that are shared by all agents in the model. Incorporating the physical and identifying characteristics in the AWS provides the modeller with the opportunity to model communication about these characteristics, use these characteristics on the agent level for reasoning or use it as an enabling condition for the execution of activities.

Given the vast amount of physical and identifying characteristics that can be used to describe the physical attributes of humans, the AWS incorporates an initial set of characteristics that are most relevant in emergency response situations. To narrow down the attributes to emergency response relevant attributes, the AWS incorporates an initial attributes set, based on the casualty card (raad van RGF'en, 2006). This card is commonly used in emergency response, so personal and medical attributes of agents systematically can be recorded to facilitate triage and the treatment process.

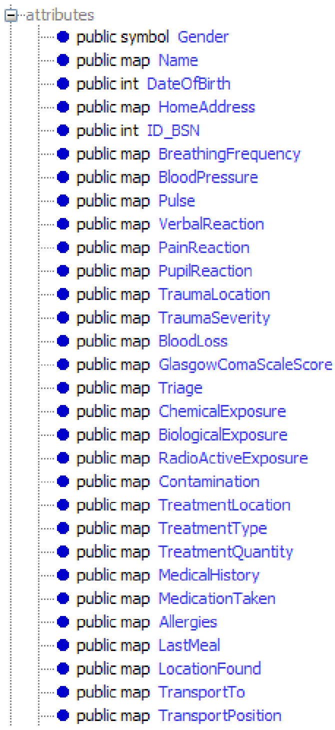
Based on the attributes mentioned in the casualty card, Figure 4.8 provides an overview of the physical (victim) and identifying attributes and how these are modelled in the AWS. The identifying attributes concern personal characteristics such as name, address, and date of birth. Physical characteristics concern the vital signs (such as pulse, blood pressure, respiratory rate); characteristics determining the Glasgow coma scale score (pain reaction, pupil reaction, verbal reaction); a description of possible traumata (location and severity of the trauma, possible chemical/biological/nuclear exposure); the agent's history (medical history, medication taken, possible allergies, time of last meal); and finally the agent's triage classification and information about how the victim should be transported and to which location.

Agent Attributes

Identifying Attributes:
Gender, Name, Date of birth, Home address, Home town, ID number

Victim Attributes:
Breathing frequency, Blood pressure, Pulse frequency, Verbal reaction, Pain reaction, Pupil reaction, Trauma (location, severity), Blood loss, Glasgow coma scale score, Triage, Exposure (chemical, biological, radioactive), Contamination status, Treatments (location, type, quantity), Medical history, previous Medication taken, Allergies, Last meal, Location found, Transport (to, position).

(a) Agent Attributes



(b) Agent Attributes in AWS

Figure 4.8: *Agent attributes and their mapping in the AWS.*

By defining most of the agent attributes within the AWS with the use of `map` attributes, the opportunity is created to provide an accurate description of an agent's injuries, while providing flexibility in the number and type of values that can be assigned to the attributes.

Besides describing injuries, these attributes can also be used as input for behavioural models of activities. The breathing frequency, for example, strongly determines the speed in which the air in a compressed air canister is used up and so influences the time a fire fighter can use a gasmask, influencing the workflow of the emergency responder. Secondly, these attributes can be used to make reference to multiple individual agents with similar attributes. When an emergency responder, for example, has to communicate the ID number (ID_BSN attribute) of all specific T0 victims, he/she has knowledge of the (Triage("T0") attribute) and the emergency responder can select the agents from who he/she has these beliefs and then communicate these beliefs to the receiver of the message.

By applying the same principle as when referring to multiple individual agents, it also is possible to, for example, select a specific agent based on his/her actual name or role name without having to refer to its (abstract) model name. For example, since the composition of the emergency organisation, the division of roles and the naming of fire department units based on their role that are involved in an emergency response are stable throughout emergencies, agents at the instance level are predefined within the AWS template (and have a predefined agent name). In Figure 4.4(b) it can be seen that the first unit on the scene

(FD_110_grp) consists of six predefined agents that are named according to their role and unit. The number 110 refers to the agent performing the role of unit commander in unit 110; numbers 111 and 112 refers to the agents performing the role of attack squad in unit 110; numbers 113 and 114 refers to the agents performing the role of water squad in unit 110; finally, number 119 refers to the agent who has the role of driver and pump operator in unit 110. The unit number and the role number, unfortunately, do not hold information about which emergency responder actually is represented by that number. The only information that is available is that unit 110 represents the first unit and consists of agents that have certain roles. One does not know which actual persons are linked to a specific unit or a specific role, making it hard to make reference to specific emergency responders. Using the identifying attributes one is able to do this.

By incorporating the attributes provided by the casualty card in the AWS template, the modeller is provided with the opportunity to construct an accurate, yet flexible, description of an agents' injuries; model the initial treatment; use physical attributes as input for behavioural models and make reference to non predefined collections of individual agents or single agents using these attributes. However, despite the possibilities created to model attributes at the agent level, this is only needed when adding highly detailed information to the model is crucial for the goal of the simulation. Adding highly detailed information about agents can be seen as very time consuming activity. The extent to which attributes on the agent level are modelled, must be determined by the focus of the simulation. Since the focus of the AWS in on modelling the communication and workflow during emergency response, the level of detail is determined by the detail at which is communicated. For the AWS it therefore is necessary to have the opportunity to model agents at varying levels of detail, dependant on the level of detail in which is communicated about the agent attributes. Since the attributes specified in the casualty card are used to identify agents and describe their current physical condition, the agent model provides the modeller to opportunity to use these attributes.

4.3 Objects

On several occasions throughout this chapter, snippets of the object model already were shown. The chapter started with an illustration of the inheritance principle that is used throughout the entire AWS by describing the representation of a specific pump-water-tender that is stationed at the fire station of Almelo centre in the Netherlands. Later, objects were encountered which were used to model abstract concepts such as the general attributes of an emergency or abstract groups of agents. Within the AWS, objects have these two functions. They can either refer to concrete (physical) objects or to abstract conceptual objects encountered in the emergency response situation. The AWS object model, consisting of a model representing the concrete objects (section 4.3.1) and an object model representing abstract conceptual objects (4.3.2), that both can be communicated about during emergency response situations, is detailed in this section.

4.3.1 Concrete Objects

As was stated in section 3.2, enumerating the requirements of the AWS to model emergency response situations, concrete objects refer to the non living elements within the emergency response situation that are situated and adaptively can engage in interactions with their surroundings or can undergo state changes. The goal of the AWS template regarding these concrete objects is to construct a general model of the objects that are used, used up, encountered or communicated about in emergency response situations. However, given the vast amount of objects that can be encountered and the changing presence of these objects in emergency responses, the initial template model presented in this section will limit the concrete object classes to those classes that are shared mostly between emergencies. This initial template model will, therefore, focus on the objects that are available to the emergency response organisations to deal with the emergency response situation. Specific classes and instances that represent the objects that emerge during emergency responses, are added later to the template when these are used in communication or determine the workflow of the emergency responders.

Figure 4.9 shows an overview of the initial concrete object classes that are incorporated in the AWS template. As can be seen, the parent concrete objects class extends into four subclasses. These subclasses comprise the objects and attributes of objects that are used for communication (Communication objects class); that are used as equipment by the emergency services (Equipment class), that are used as a the basic means of transport (Means of transport class), and that can be used as add-ons to the basic means of transport containing a collection of specialist equipment (Trailers class).

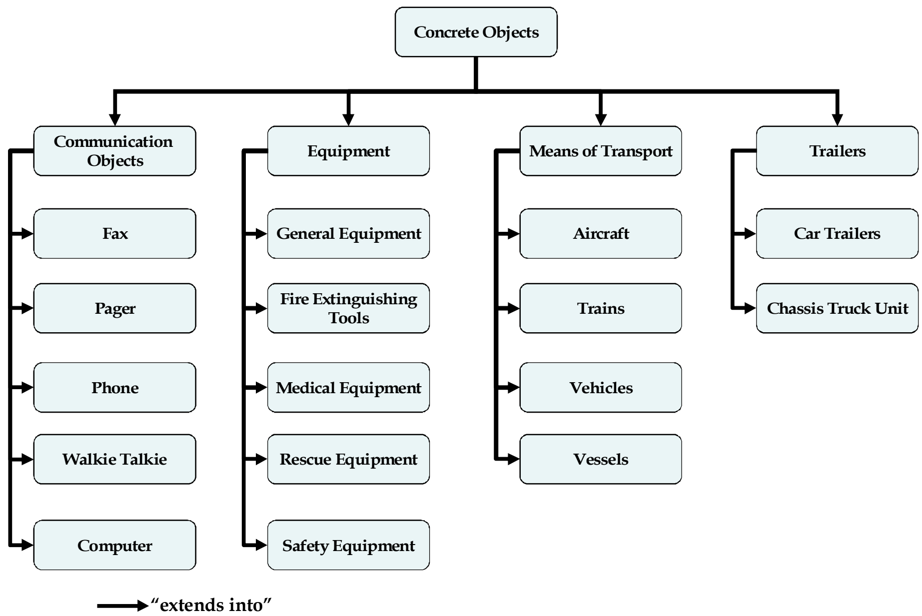


Figure 4.9: *Concrete object classes in the AWS model.*

In the following sections (4.3.1.1 to 4.3.1.4) the implementation of these subclasses in the AWS is described.

4.3.1.1 Communication objects

Modelling the distribution of information is crucial when one wants to determine the communication load (see Chapter 7) and the workflow (see Chapter 5) of the emergency responders. During an emergency response, a large amount of communication is mediated by communication devices. In order to facilitate the modelling of communication between emergency responders about the attributes of the communication devices (such as communicating a phone number), these devices must be able to be modelled in the AWS. However, besides being able to model communication about communication devices used in emergency response, it also is important to be able to model the resulting information flow when these devices are used. A phone, for example, is used to transmit information from a single agent to another single agent, where a walkie-talkie is used to transmit information from a single agent to one or more agents simultaneously. Consequently, information is able to impact a single agent's workflow in the case of the phone or more agents' workflows in case of the walkie-talkie. Furthermore, from a communication load perspective, the phone demands cognitive resources from a single agent to process the information, while the walkie-talkie demands cognitive resources from a group of receivers in order to process the sent information.

As can be seen in Figure 4.9, an initial set of communication object classes that are incorporated in the AWS refer to communication devices that are commonly used in organisations. These include phones, fax machines and computers. More specific to the Dutch emergency response organisation are the paging system P2000, extending the pager class, and the C2000 system, extending the walkie-talkie class. The attributes that where communicated about concerned the phone numbers of the phones carried by the officers on duty and the channels that were used in the C2000 system. In the AWS, it thus minimally should be facilitated that agents can carry (mobile) communication devices and that beliefs (residing in agents) and facts (residing in objects) of phone numbers and walkie-talkie groups can be modelled.

To be able to model the information flow mediated by communication devices, the AWS should additionally incorporate knowledge preconditions and technological preconditions. These preconditions are used to select a single individual or multiple individuals as receivers when using a communication device. Knowledge preconditions refer to specific knowledge that must be present in the agent in order to use a communication device to communicate with another agent. An example of a knowledge precondition is knowledge about the phone number of the phone owned by the receiver, since it is not possible to call someone without having that person's phone number. Knowledge preconditions reside in the agent. Technological preconditions, on the other hand, refer to preconditions that have to be met in order for the communication object to fulfil its function. Examples of these preconditions are if the device is turned on, or if the device is connected to its communication network.

As can be seen in Figure 4.10, `FD_110_agt` carries both a phone (`current contains FD_110_agt_Phone_obj`) and a walkie-talkie (`current contains FD_110_agt_WalkieTalkie_obj`). He furthermore has a belief about his own phone number (`FD_110_agt_Phone_obj.numberOrGroup("number") = "0615842356"`). He also has a belief that the unit commander of unit 120 owns a phone

(`FD_120_agt_Phone_obj.Owner("FD_120_agt") = true`) and has the phone number at which he can reach the unit commander of unit 120. The `FD_110_agt` has the belief that his walkie-talkie is tuned to group 405 (`FD_110_agt_WalkieTalkie_obj.numberOrGroup("currentGroup") = 405`).

Agent Beliefs

```
agent FD_110_agt memberof FD_110_grp, FD_UnitCommander_grp {
    initial_beliefs:
      //Phones
      (current contains FD_110_agt_Phone_obj);
      (FD_110_agt_Phone_obj.numberOrGroup("number") = "0615842356");
      (FD_120_agt_Phone_obj.numberOrGroup("number") = "0615842323");
      (FD_120_agt_Phone_obj.Owner("FD_120_agt") = true);
      (FD_120_agt_Phone_obj.CommunicationObjectType("Phone") = true);
      //Walkie Talkie
      (current contains FD_110_agt_WalkieTalkie_obj);
      (FD_110_agt_WalkieTalkie_obj.numberOrGroup("current group") = "405");
}
```

Object Beliefs

```
object FD_110_agt_Phone_obj instanceof Phone_cls {
   initial_beliefs:
      (current.numberOrGroup("number") = "0615842356");
      (current.Status("on / off") = "on");
      (current.Status("broken") = false);
      (current.Status("connected with communication network") = true);
      (current.Owner("FD_110_agt") = true);
}
 object FD_110_agt_WalkieTalkie_obj instanceof WalkieTalkie_cls {
   initial_beliefs:
      (current.Status("on / off") = "on");
      (current.Status("broken") = false);
      (current.Status("connected with communication network") = true);
      (current.Owner("FD_110_agt") = true);
      (current.numberOrGroup("current group") = "415");
}
```

Figure 4.10: *Agent and Object beliefs regarding communication objects.*

It should however be noted that the agent beliefs mentioned, represent the *assumed* actual state of the devices. The *actual* state of the objects is represented by the beliefs of the objects themselves. By using discrepancies between the values of the same object attribute incorporated in the object and the agent, communication errors can be modelled in the AWS. When, for example, the unit commander thinks he is using walkie-talkie group 405, he in fact is broadcasting on group 415. System preconditions such as Status ("on / off"), Status ("broken"), Status ("connected with communication network") will limit the device in receiving and sending information.

Summarizing, by incorporating the communication object classes and the attributes of communication devices that surfaced during the emergency response exercises combined with the attributed needed to define the knowledge and technological preconditions that are necessary to structure the information flow, both the information flow and communication about communication objects can be modelled in the AWS.

4.3.1.2 Equipment

Instances extending the means of transport class and the trailer class are able to carry equipments (specified with the "contains" relation) that are used during emergency response situations. To support modelling of the communication about the equipment used during emergency response (carried by the means of transport and trailers or otherwise present or used at the emergency response location), these objects have to be included in the AWS. An example of equipment related communication between the unit commander of unit 130 (UC 130) and the officer on duty (100) extracted from the emergency response exercises is shown in Figure 4.11.

Sender	Receiver	Message
UC 130	100	130 for the 100
UC 130	100	Over
100	UC 130	130, 100 here
100	UC 130	Over
UC 130	100	We went to the truck. There is one person trapped there
UC 130	100	Can we have the rescue tools from the rescue vehicle?

Figure 4.10: *Snippet from a dialogue between the unit commander of unit 130 and the officer on duty regarding the availability of rescue equipment.*

Within the AWS equipment class, a general distinction is made between emergency response specialist tools that are used by the fire department to extinguish fires (such as a street water cannon); general not emergency response related equipment (such as screwdrivers); medical equipment (such as bandages or the contents of a first aid kit); rescue equipment (such as cribbing tools and hydraulic spreader and cutters); and safety equipment such as a chemical suit.

When looking more closely at, for example, the tools that are used by the fire department to extinguish fires (Figure 4.11), it can be seen that this class further extends into object classes containing specialist equipment. Since these classes are subclasses of the general object classes, instances can contain beliefs that reflect their current status, usability and location. The status attribute describes the status of an object in a similar fashion as the status of the communication device was modelled in the previous section. One can, for example, describe and communicate about the availability status of an object, who is using the objects, or if the tool is used up or broken. Returning to the dialog, one can thus communicate about the availability status of the instances residing in the `RescueEquipment_cls` that is contained by a specific rescue vehicle (specified with a "contains" relation).

Improvisation, however, plays a large role in emergency response (Mendonça, Jefferson, Harrald, 2007), and requires responders to use the available equipment in a way for which they originally were not intended to be used. When faced with a situation where resources are limited, emergency responders have to explore alternative ways to use the available tools to reach their goal. The Usable attribute (`public map Usable`) in the AWS template describes communication about the possible applications of tools, describing its intended use and its possible use. Typical communications that occur in those moments are, for example: “can we use this low pressure attack hose to lift the car?”. This communication is modelled as a belief being communicated between agents: `lowPressureAttackHose_obj.Usable("lift car") = unknown.`

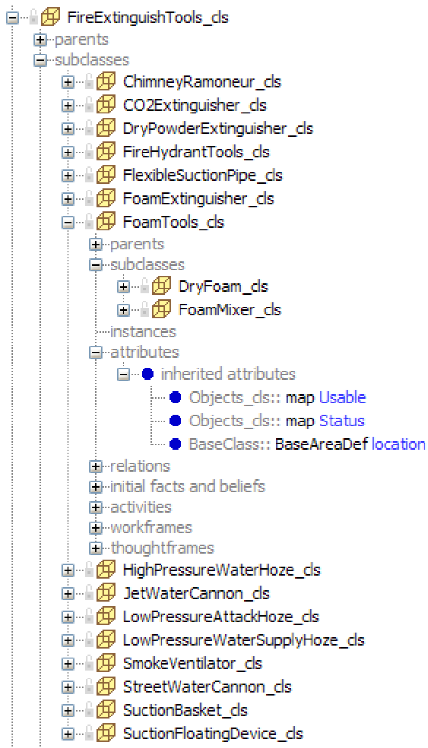


Figure 4.11: *Hierarchal view of the class representing the fire extinguishing tools in the AWS*

However, when communicating about the possible use of objects or about objects in general, communication may also make reference to the general class of objects instead of a particular instance. When emergency responders in the previous example communicate “can we use *a* low pressure attack hose to lift the car?”, this communication is modelled as following belief communicated between agents: `lowPressureAttackHose_cob.Usable("lift car") = unknown.` The `_cob` here is an indication that communication referred the general class of low pressure attack hoses instead of one particular instance. To facilitate modelling this level of abstraction in the communication, the AWS incorporates an object conceptual class incorporating references to collections of objects.

Furthermore, objects are situated. Communication about the location of a certain object can be modelled using the location attribute. When, for example, an emergency responder wants to know where the jet water cannon is located, a belief is transmitted such as `jetWaterCannon_obj.location = unknown.` This, in turn, can trigger the receiver to share his knowledge about the location of the jet water cannon.

By using the object classes and attributes presented above, the AWS can model communication about objects used in emergency response at various levels of detail. Furthermore, by using the Usable attribute, the AWS can model improvisation and communication about possible uses of objects.

4.3.1.3 Means of Transport

During emergency responses several means of transport, ranging from police cars to specialized fire trucks and trauma helicopters owned by the emergency response organisations, rush to the scene, transporting specialized material and emergency responders.

The means of transport class within the AWS, is aimed at providing the conditions to model the different modes of transport that are used during emergency response and thus support the modelling of information exchange regarding these objects. In addition, emergency responders communicate about attributes of these means of transport and the means of transport also fulfil the function of containing objects and agents and also performing the task of moving between locations with the agents and objects. This section provides a snippet of the means of transport that are predefined in the AWS and the attributes that enable the modelling of communication about these objects. Also the question how agents and objects can be moved with the use of means of transport is answered.

Resulting from the emergency response exercises, several means of transport were the topic of conversation between emergency responders. These are:

- Ambulances to move victims
- Buses to move victims
- Cars that were involved in the car crash
- OGS (accidents dangerous substances) vehicle owned by the fire department containing specialist material to handle accidents where dangerous substances play a role
- Pump-water-tender owned by the fire department
- Officer on duty car
- Rescue vehicles owned by the fire department that contains specialist equipment for rescue work

Attributes that were communicated about were: if the vehicle was moving to the incident location; if the vehicle was standing by, if the vehicle contained victims; and, finally, the availability status of equipment (i.e., the dialogue shown in Figure 4.10 regarding the rescue vehicle). The means of transport mentioned during the emergency response exercise only concerned vehicles. However, since the emergency response organisations also may use other means of transport, three other means of transport classes are included that extend the general means of transport class. The general means of transport consists of 4 subclasses, providing the possibility to model most means of transport: the aircraft class (representing objects such as planes and helicopters); the trains class (representing rail track bound modes of transport); vehicles (representing means of transport that use the roads); and, finally, vessels (representing objects that move over water). The level of detail at which the classes and instances are modelled depends on the means of transport that are used in the model and the focus of the simulation. Figure 4.13, for example, shows how a trauma helicopter can be incorporated in the model.

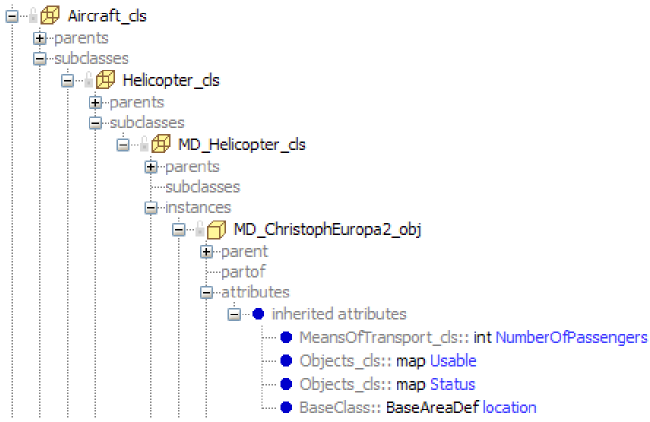


Figure 4.13: *Trauma helicopter Christoph Europa 2 modelled in the AWS*

The helicopter class extends the aircraft class. Given that a trauma helicopter is a type of helicopter (but has specific attributes), it further extends the helicopter class. The Christoph Europa 2 furthermore is an instance of the trauma helicopter class (`MD_Helicopter_cls`).

Since all these classes all are children of the object class, the attributes that are specified at the object class level propagate down to all instances of the means of transport classes. The attributes that surfaced from the emergency response exercises, can be modelled using the status attribute or by specifying a contains relation between the means of transport instance and other objects or agents. To, for example, model the communication about if a vehicle is moving towards the incident location using the "status" attribute, one can specify `object_obj.Status("moving towards incident location") = true` if this were the case. In a similar way, one could model communication about the stand by status of the object by specifying `object_obj.Status("stand by") = true` if this were the case. Furthermore, using the inherited `numberOfPassengers` attribute (from the `meansOfTransport_cls` class), one is able to model communication about the number of agents present in the vehicle. The inheritance of attributes so simplifies the modelling of attribute values at the instance level.

Figure 4.14, additionally, shows a snippet of the standard packing list of the pump-water-tender modelled in the AWS, illustrating the contains relation between objects. The snippet illustrates that the pump-water-tender contains two types of gloves, boots, chemical suits and flashlights. This results in the fact that wherever the pump-water-tender goes, the equipment goes also.

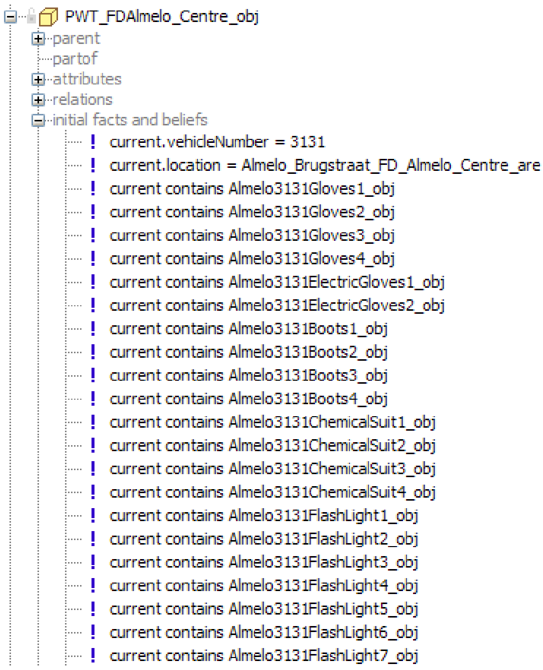


Figure 4.14: *Snippet of the packing list of the pump-water-tender stationed at the fire station of Almelo centre in the Netherlands.*

Since a similar contains relation can be specified for agents, movement with the use of means of transport can be modelled by "placing" the agents in the object. Section 3.3.1 specified this activity for the "moving to the incident location" of the emergency responders during the Hercules plane crash.

Thus, the AWS template provides the modeller with the opportunity to model communication about various types of means of transport. Furthermore, by using the "contains" relation, the AWS facilitates modelling the movement of agents and objects with the use of these means of transport.

4.3.1.4 Trailers

During emergency response situations trailers are used to transport specialist equipment to the emergency response location when no specialist vehicle (such as the OGS vehicle or a command vehicle) is available. Trailers can either be car trailers, which can be linked to a vehicle such as, for example, a pump-water-tender, or chassis truck units. The latter can be described as a container unit that is moved to the accident location by a chassis truck. The chassis truck can move one unit at a time, but once a unit is delivered at the accident location it can get another unit if needed. During the emergency response exercises, two chassis truck units were used in communication: the WTS 1000 unit and the COH unit. The WTS is a unit used to transport water over 1000 meters using hoses, while the COH unit is the commando unit at the scene.

Figure 4.15 shows the car trailer and chassis truck units incorporated in the AWS (including the WTS and the COH chassis truck unit). In the initial set incorporated in the AWS template, in total 20 different chassis truck units were differentiated and 3 car trailers. Besides the inherited attributes that apply to all means of transport objects, using the "contains" relation provides the modeller with the freedom to model the objects that are carried by these units, and model the units while being transported by a chassis truck or car. Movement of these trailer objects in the AWS thus can be modelled by "placing" the trailers in their carriers.

This gives the modeller the possibility to model car trailers and chassis truck units, and model communication about these units using the inherited attributes by their parent classes. Trailers can be moved by their carriers by either specifying a "contains" relation in advance or by placing the trailer in the carrier and move the carrier to the location where the unit is needed.

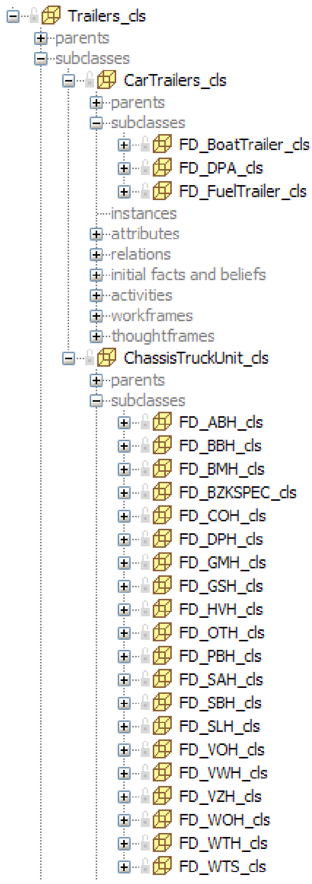


Figure 4.15: *Trailer class extending into the car trailers class and the chassis truck unit class.*

4.3.2 Abstract Concepts

As mentioned at the beginning of section 4.3, objects within the AWS refer to concrete or abstract concepts. The concrete object model that was presented in section 4.3.1 provides the modeller with the opportunity to model communication about concrete, situated objects' attributes. Furthermore, it allows the modeller to model object specific behaviour (such as the ability to move between locations), specify the ability to act as a container for other objects (using the contains relation) and for it to be used for other than the originally intended purpose. In contrast to the concrete objects, abstract objects are not situated, do not engage in behaviour and are solely used to model communication about non tangible abstract concepts.

Three examples of conceptual classes and instances that we saw earlier, concern the emergency conceptual object model, the conceptual object class used for abstract group reference and finally the conceptual object class used for referencing to collections of objects. The first provided the modeller with the opportunity to model communication and information exchange regarding the attributes of the emergency concept (section 4.1.1.1), such as a description of the current assembly of the emergency response organisation or a general description of the number and severity of the victims involved. The second abstract concept that was encountered in this chapter, provided the modeller with the opportunity to model communication that referred to groups of people without referring to individual agents (section 4.1.1.2). Communication such as "the police is surveying the area" is in this way made possible in the AWS. In a similar manner as the abstract group reference, the third abstract concept enabled the modeller to model communication about collections of objects (section 4.3.1.2) such as "can we use a low pressure attack hose to lift the car?".

However, besides the ability to model communication about the emergency concept and collections of groups and objects, the communication during the emergency response exercises that was analyzed also made reference to one other abstract concept: meetings and gatherings (bilateral or non official meetings). Attributes that surfaced in the communication were:

- Meeting time /date: when is the time the next meeting will begin, or at what time was another meeting held.
- Meeting agenda: what will be discussed during the current meeting or what was discussed during previous meetings.
- Meeting topic: what is the current topic being discussed.
- Meeting location: what will be the location where the meeting will be held, or where was the meeting held.
- Meeting Start / End / Adjourned: has the meeting officially started or ended or is it adjourned.

Coordinating meetings and gatherings are not uncommon during emergency response situations and demand a significant amount of time from the responders. During these meetings, emergency responders exchange information and make up a plan of attack regarding the situation the emergency response organisation is faced with. The absence of information of, for example, the location of the meeting can influence the workflow of the emergency responder, missing the meeting and consequently missing information distributed there. Since the emergency response organisations meet in mobile units, descriptive information about the meeting itself determines who is going to be where at what time talking about which topics. The AWS template model thus should incorporate descriptive information about meetings that can provide the modeller ways to model communication about meetings.

Based on the different multidisciplinary teams that emerge (see section 4.1.1) and the corresponding meetings that are held during emergency response situations, an initial set of meeting classes was determined to model the communication that concern meetings in the AWS. In addition, to facilitate the modelling of communication about bilateral or non official meetings (gatherings) held during the emergency response, an extra conceptual class was added to model communication about these non official meetings. Figure 4.16 shows the gatherings abstract object class as it is implemented in the AWS (`Gatherings_coc`).

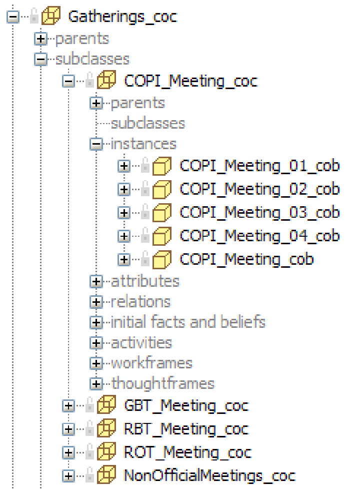


Figure 4.16: *Gathering conceptual object*

Representing the initial set of different meetings that are held in emergency response situations, the gathering abstract object class extends into five sub classes: meetings of the commando disaster area group (`COPI_Meeting_coc`); meetings of the local government policy team (`GBT_Meeting_coc`); meetings of the regional policy team (`RBT_Meeting_coc`); meetings from the regional operational team (`ROT_Meeting_coc`); and finally the non official meetings (`NonOfficialMeetings_coc`).

During emergency response communication that refers to meetings, one does not always refer to an actual meeting, but also can refer to attributes of the meeting type in general (such as the meeting interval). As can be seen, when unfolding the class representing the COPI meeting (`COPI_Meeting_coc`), the instances within this class are of two types. The numbered instances (`COPI_Meeting_01_cob` to `COPI_Meeting_04_cob`) represent the actual meetings held, while the unnumbered instance refers to a COPI meeting in general.

Using the attributes of the `Gatherings_coc` class shown in Figure 4.17, one can model communication about a meeting type in general, as well communication about specific meetings.

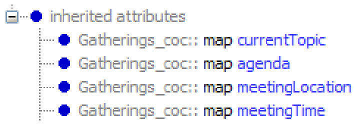


Figure 4.17: *Gatherings_coc attributes.*

The `meetingTime` attribute, for example, can be used to model communication that the starting time of the second COPI meeting (`COPI_Meeting_02_cob`) is at 10:15 by communicating the belief `COPI_Meeting_02_cob.meetingTime("Start") = "10:15"`, while on the other hand one can communicate about the COPI meeting interval being 20 minutes by communicating the belief `COPI_Meeting_cob.meetingTime("interval") = 20`. Furthermore, by using the flexible map attribute, the AWS is able to facilitate the modelling of communication about meetings that turned up during the emergency response exercises and is able to handle variations within these attributes.

With the use of abstract classes, the modeller is able to incorporate communication about attributes of these abstract classes in the model of emergency response. In addition, the modeller has the opportunity to apply the attributes to a single or a collection instance, representing similar instances, varying the level of detail used in communication.

Apart from the abstract classes presented in this section, another abstract object class is incorporated in the AWS template. This regards abstract objects that reflect the available information that can be observed by agents from their surroundings. This class of abstract objects regarding information about the locations is described in section 4.4.

4.4 Geography

As indicated in Chapter 3, the environment in which an emergency unfolds consists of geographic instances, objects, agents and relations between these elements. In the previous sections of this chapter, the objects, agents, the relations between them and their implementation in the AWS template was presented. This section will describe the geographic classes that enable the AWS to situate the emergency response and enable the geographic instances to interact with agents and objects. The geography representation that is used in the AWS serves as a *placeholder* (locating agents and objects), as an *information provider* (providing location information to its inhabitants), and as the *framework* that is used to model movement of agents and objects.

The section starts by describing and implementing the first function of AWS geography model, the function as a placeholder. The placeholder model in section 4.4.1, situates the emergency response activities; enables agents to refer to locations in communications and enables agents to assert information about the actual (static) state of the location. The actual static state refers to the stable attribute values of the location that can be observed by all agents that are located within a certain location. The attribute values regarding the actual static state of the location do not change throughout the course of the simulation (such as that a room is connected to a hallway). Building on the placeholder model, the second function –that of information provider- is presented in section 4.4.2. There, in contrast to the attributes that describe the actual static state in the placeholder model, the opportunity is created for the modeller to incorporate attributes and attribute values regarding the actual dynamic state of the location. The actual dynamic state refers to attribute values of the location that can be observed by all agents that are located within a certain location and that can change during the course of the simulation (such as the smoke density in a room). Finally, using the elements presented in section 4.4.1 and 4.4.2, section 4.4.3 presents the framework that is used to model movement of agents and objects in the AWS.

4.4.1 AWS Placeholder function

Similar to that one does not know beforehand which emergency responders will be involved in the emergency response organisation, one also does not know which locations are going to play a key role in the emergency. As was shown in section 4.1 regarding the emergency response organisation, uncertainty often remains at the instance level. During each emergency, the same organisations form the backbone of the emergency response organisation, each of them having predefined roles and tasks during the response. With the use of the inheritance structure, attributes defined at organisation and role level, propagate down to the instance level whenever an instance is modelled. A large part of the attributes at the instance level thus is defined before the actual instance is incorporated in the model.

To equip the geography model with the placeholder function, a similar approach is taken. Although one does not know exactly which actual locations will be involved in the response or are used in communication, the types of locations (placeholder classes) that can be encountered, their hierarchal structuring and the attributes defining their interconnectedness are stable. The hierarchal structuring of the placeholder classes that are incorporated in the AWS is shown in Figure 4.18.

Anticipating on possible inheritance problems that may surface, given the large differences between the factual dynamic state attributes that apply to indoor and outdoor locations, two separate hierarchal models are used in the AWS template. For example, although an instance of a country and an instance of a room share attributes such as temperature and humidity, it cannot be assumed that, even though a room within a building is part of a country, the room will be covered in snow when the country is covered in snow. Therefore, one model is used to incorporate locations that are *open* to the elements (Outdoor locations) and one model refers to the locations that are *not open* to the elements (Indoor locations). It, however, should be noted that although outdoor and indoor locations are represented in different models, the relations that exist between outdoor and indoor locations must be specified. Furthermore, since both models incorporate representations of locations, one parent class is used where shared attributes are defined (the BaseLocations class).

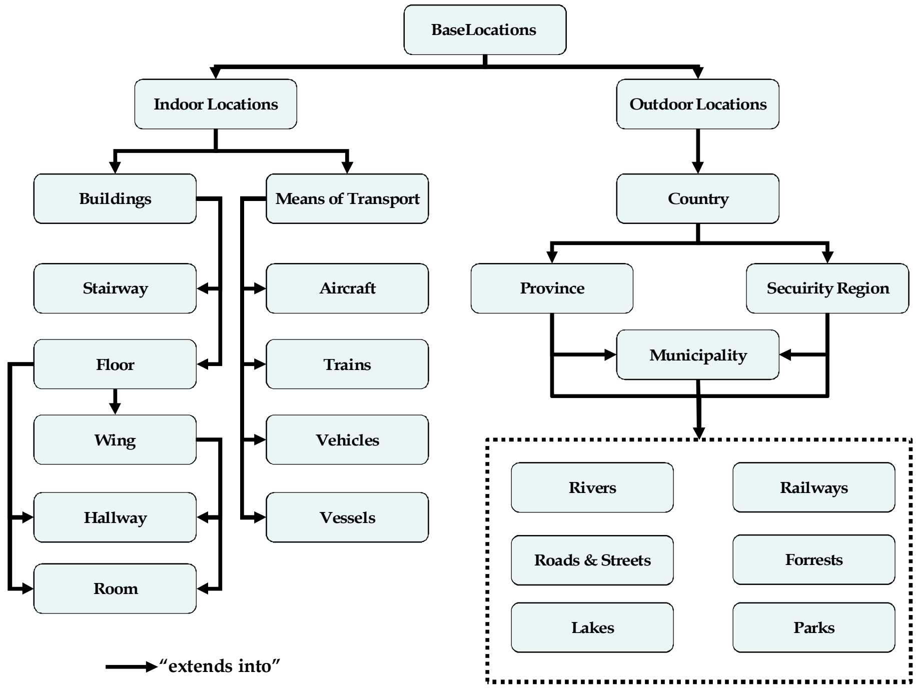


Figure 4.18: *Placeholder classes incorporated in the AWS template.*

4.4.1.1 Outdoor locations

Extending the outdoor location class, the country class is the first placeholder class that is encountered. A country concurrently extends parallel into a province and a security region. Both refer to major divisions of a country and consist of groups of municipalities. A province is a subdivision of a country that functions as the regional legislative authority positioned between the local legislative authority (municipality) and the national legislative authority (national government). The country thus consists of provinces that in turn consist of municipalities. The security region refers to a division of the country containing several municipalities and is used to structure regional emergency response. A municipality thus is a member of a province and a member of a security region. In the AWS, these parallel structures are represented by using multiple class membership. A municipality can be a member of a province and a security region as well, inhering attributes from both. The same multiple class principle is applied to rivers, railways, forests, parks, lakes and roads and streets, since these may be located in more than one city, province, security region or even one country. It should be noted that sections of these location types can be specified and located in a single province, security region or country. Figure 4.19 shows how the Hengelosestraat in Enschede is modelled in the AWS.

The Hengelosestraat (=street) is of the type "roads and streets" and is located in the Netherlands (type of "country"), and more specific in the municipality of Enschede (type of "municipality", extending the province Overijssel and the security region Twente). The instance representing the Hengelosestraat (`Enschede_Hengelosestraat_are`) is subsumed in the `Enschede_Hengelosestraat_adf` class, that, besides being able to contain the instance representing the entire street also comprises instances of subsections of the Hengelosestraat. The instances representing the entire street and the subsections of the street thus inherit the attributes specified at the street class.

To model municipality specific attributes of roads, the `Enschede_Hengelosestraat_adf` class is modelled as a child of the roads and streets in Enschede (the `RoadAndStreet_Netherlands_Enschede_adf` class) that in turn inherits attributes that apply to all streets in the Netherlands from the `RoadAndStreet_Netherlands_adf` class, and that apply to the municipality of Enschede (the `Netherlands_Enschede_adf` class). The `RoadAndStreet_Netherlands_adf` class inherits its attributes from the general "roads and street" type class (the `RoadAndStreet_adf` class).

Figure 4.19: AWS *representation of the Hengelosestraat street in Enschede, the Netherlands*

In the first minutes of an emergency, the locations used in communication often remain limited to instances that represent large areas as detailed information about the emergency location is not yet available. In this way, the use of general instances can be anticipated on. In the AWS, therefore, each of the subclasses descending from the placeholder classes, incorporates an instance representing the area represented by the subclass. This instance can be used in communication and as a placeholder for the general location.

By using the inheritance structure presented in Figure 4.18, the modeller has the freedom to model instances of outdoor locations that automatically inherit the attributes and attribute values from all their parent classes (such as municipalities, provinces, security regions, country and subclasses of these classes), and can refer to areas representing the general classes.

4.1.1.2 Indoor locations

Of course, many activities of emergency responders also occur in locations that are not open to the elements. One can, for example, think of an inside fire attack by the fire department to fight a closed fire or providing medical aid in an ambulance. To be able to situate activities of emergency responders and objects in these locations, buildings and the inside of vehicles were incorporated in the AWS. Firstly, the inside of buildings (that extend indoor locations) generally consist of floors and connections between these floors in the form of stairways (or elevators). A floor, furthermore, extends into hallways and rooms. However, when a wing is present in a building, the floor can be divided in wings that can further be divided into rooms and hallways. A room or hallway thus can inherit attributes and attribute values from the floor it is located in and from the wing it is part of. Instances of these classes are linked to a specific building. Figure 4.20, for example, shows how the fire station of the municipality of Enschede (location north) is modelled in the AWS.

Figure 4.20: AWS *representation of the fire station of Enschede (location north), the Netherlands*

Providing the opportunity to model country dependent characteristics of buildings that influence emergency response (for example, caused by different national building regulations), the Buildings_Netherlands_adf class -that extends the general Building_adf class- contains all instances of buildings that are located in the Netherlands. To model differences in buildings depending on municipalities, the Buildings_Netherlands_adf class further extends into the building classes that represent buildings within a certain municipality (Buildings_Netherlands_Enschede_adf). The fire station location north class (Enschede_HengeloseStraat_FD_Enschede_North_adf), which is a building in the municipality of Enschede, successively extends this class. To model communication such as "the officer on duty is located at the fire station north", a general placeholder about the fire station can be used resulting in the belief OfficerOnDuty_agt.location = Enschede_HengeloseStraat_FD_Enschede_North_are. However, in order to locate agents or objects or communicate about more specific placeholders (such as when communicating about agents or objects in a specific room or floor within the building), the building class extends into 5 interrelated classes representing the floors, wings, hallways, stairways and rooms within the building in which detailed location instances can be modelled.

From an inheritance perspective, the model representing a building and the model representing its outdoor environment are independent. However, in practice these models are connected: a building for example can be located in a street. To solve this issue regarding the interconnectedness of buildings and their environment, partof and parts relationship between buildings and their environment are defined. As can be seen in Figures 4.19 and 4.20, the fire station building instance and the Hengelosestraat instance are connected using the parts and partof relationship: the fire station is part of the Hengelosestraat (Figure 4.20 partof) and the hengelosestraat refers to the fire station as a part (Figure 4.19 parts). Agents located in these locations automatically have beliefs about the parts that make up the location and of which larger location it is a part of, thus connecting the two models.

The parts and partof relation are not sufficient to model all relations between locations that can be used in communication. Therefore, in addition to the parts and partof relations, the relational attributes specified by the Open Geospatial Consortium (2006) that reflect the geographic relations that cannot be expressed with the parts and part of relationships are incorporated in the AWS template (Table 4.5).

Table 4.5 *Additional geographic relations incorporated in the AWS template.*

Geographic Relation	Explanation
Equals	Specifies that the geometric properties of the placeholder instance is equal to the geometric properties of another placeholder instance (such as similar rooms in a building).
Disjoint	Specifies that the placeholder instance is disjoint from another placeholder instance (such as two houses in a street that are not connected with each other, but are located in the same street).
Intersects	Specifies that the placeholder instance intersects with another placeholder instance (such as two intersecting roads).
Touches	Specifies that the placeholder instance touches with another placeholder instance (such as two houses that are connected)
Crosses	Specifies that the placeholder instance crosses another placeholder instance (such as a fly over).

Since the attribute values of stable actual representations of a location are transformed into subjective beliefs in the agent whenever an agent is placed in a location, the attribute values regarding the relations specified in Table 4.5 are not specified beforehand. If this were the case, this would entail that whenever an agent enters a location, the agent would receive all meta information about that location, for example, knowing that a room with similar geometric properties is located at the other side of town. These relations thus are only used for agents to communicate about the relations between locations and not to inform the agents about the geography they are in.

Summarizing, using the classification shown in Figure 4.18, the AWS template gives the modeller the opportunity to construct a model of indoor and outdoor locations that enables the situation of objects and agents; provide ways to model communication in which general or specific locations are used; and finally, enables the modeller to represent communication about possible relations that exist between locations.

However, as mentioned at the start of this section, the placeholder model describes the reoccurring elements in the geography that situates the emergency response activities and makes communication concerning locations possible. The previous paragraphs presented how buildings and outdoor locations in which emergence response activities can be located are incorporated, using the geography template. As was illustrated in section 4.1.2, by providing a partial analysis of the activities of a unit commander that make up the activity of moving towards an accident location, the inside of means of transport also can be the scene where activities are executed. The activities of the unit commander, for example, were situated in the fire truck and made use of the objects and agents that resided in the fire truck.

To be able to situate activities, agents and objects and make communication possible concerning the inside of the means of transport, each instance of the means of transport class is equipped with its own location. By specifying separate subclasses for the means of transport types that were identified in the concrete object model, means of transport specific attributes can be defined that propagate down to the locations modelled in the means of transport sub class.

Wrapping up, the placeholder model that is incorporated in the AWS is used to situate emergency responders in both outside locations that are open to the elements as well as inside locations, such as buildings and the inside of means of transport that are not open to the elements. At these locations, the agents and objects are able to perform activities and assess information about objects and agents that also reside in that location as well as about the location itself. The latter aspects refer to the information provider function of the locations which will be presented in the following section.

4.4.2 Information Providing Function

The placeholder model presented above, situated the emergency responder and the objects that are used during the response. However, being in an environment consequently means being able to acquire observable information and communicate about attributes of the environment and being able to interact with that environment. For example, when an emergency responder enters a room he/she can immediately observe that an object is on fire and that the room is filled with smoke. When the emergency responder extinguishes the fire, smoke will eventually clear.

The aim of the AWS template, however, is not on constructing a detailed model of all possible interactions and their consequences for the environment. Using the general observable attributes, the AWS's aim is providing a general approximation of the consequences that environmental attributes have on the workflow. This section will therefore describe an initial set of location attributes that enable the modeller to model observable location information that, in contrast to the actual static state, can change during the course of the simulation: the actual *dynamic* state of the location.

Table 4.6 shows the attributes that were used during the emergency response exercises to discuss the actual dynamic state of locations.

Table 4.6: Actual state attributes communicated about during the emergency response excercises.

Attribute	Explanation	Examples (translated from Dutch)
Fire	Intensity, presence, likely duration, will jump over to, under control, type of fire	- "there a fire in the cellar" - "the only possible location the fire can jump over to is the location above it, the school" - "it concerns a petrol fire"
Geometric properties	Length, width, height, volume, water level	- "the size of the cellar is 20 by 20 by 2 meters"
Status	Stable, traffic flow	- "he said the filling station was stabilized" - "Traffic is congested on the s15"
Smoke	Toxic, intensity, colour, direction	- "smoke does not concern us, unless of course the smoke contains toxic gasses" - "the smoke that is released from the area is thick black smoke" - "the smoke is heading towards the school"
Sediment	Soot, snow, forecast, toxic	- "soot is coming down everywhere" - "it is a bit snowy"
Fog	Intensity, forecast	- "locally, very dense fog"
Temperature	Degree, forecast	- "it is four degrees Celsius"
Wind	Direction, speed, forecast	- "the wind is coming from the south east" - "because if the wind direction changes.." - "the wind speed is 3 meters per second, a mild wind"

Similar to what we saw with the attributes concerning the concept of the emergency, several general attributes can be derived from the communication that refer to fire attributes, geometric properties, general status properties, smoke attributes, sediment attributes and weather attributes, such as fog, temperature and wind. Figure 4.21 shows the location attributes that are incorporated in the placeholder model, providing an initial set of actual (dynamic and static) state attributes that apply to locations.

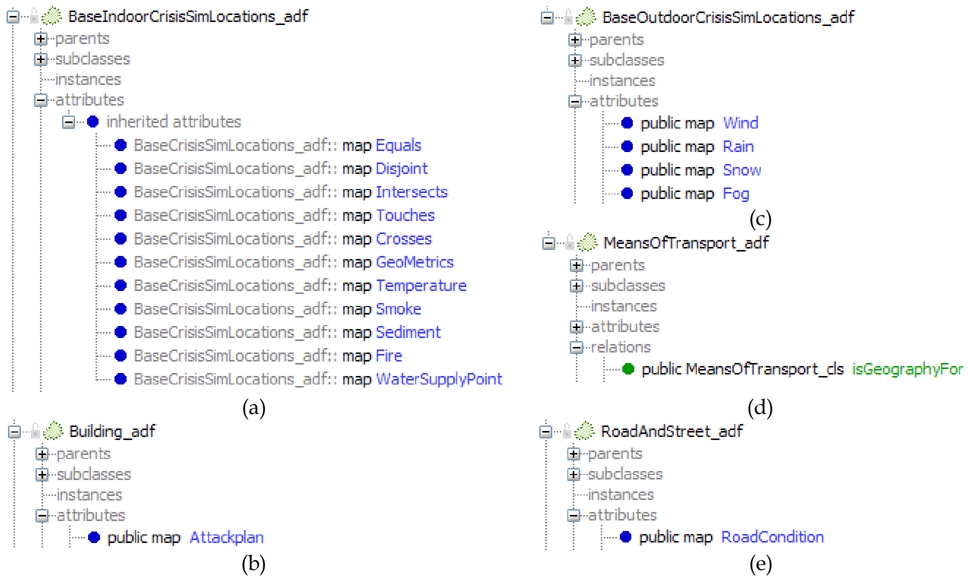


Figure 4.21: *Initial set of actual (dynamic and static) state attributes of locations in the AWS.*

As can be seen in Figure 4.21, all actual state attributes that apply to indoor locations (a) also apply to outdoor locations, since all attributes within the indoor locations class were propagated down from the Base locations class (`BaseCrisisSimLocations_adf`). The attributes defined in the base locations class concern the attributes that represent the geographic relations that can be defined between locations, geometric attributes, temperature, smoke, sediment, fire and finally water supply point. The latter was added since information about the location and capacity of water supply points are frequently communicated about during the first phase of the emergency response (Nibra, 2005a). For the same reason, another attribute of a building was added (`public map Attackplan` (b)) and an attribute concerning the road condition (`public map RoadCondition` (e)). Furthermore, since outdoor locations are open to the elements, the outdoor location class is equipped with attributes that provide the opportunity to describe the weather conditions such as wind, rain, snow and fog (c). The means of transport class, furthermore, is equipped with a relational attribute describing its relation with the object representing the means of transport in the means of transport object class (d). All attributes, except the `isGeographyFor` relation, use the map attribute collection type. As was demonstrated in section 4.1.1.1, by using the multiple value assignment to a single attribute, the modeller is able to apply these attributes in a flexible manner, using the map collection attribute.

As was indicated at the start of section 4.4, the actual static state attributes values remain stable throughout the simulation and cannot be changed. The actual dynamic state attribute values can be subject to change. A limitation in the Brahms modelling environment is that instances in the geography model can not engage in activities and hence cannot change their attribute values during the simulation. Although this is exactly the intention for the static state attributes, this makes it impossible to model dynamic state attributes of the environment with the use of the geography model in Brahms.

To solve this problem, the hierarchal structure of the placeholder model is present as a parallel conceptual object model in which each location class and instance respectively is represented as an object class and an object instance (location information object). This object concurrently is located in the area that is modelled by the location instance. Figure 4.22(a) shows the object linked to the `Enschede_Hengelosestraat_are` as an inhabitant of the `Enschede_Hengelosestraat_are`.

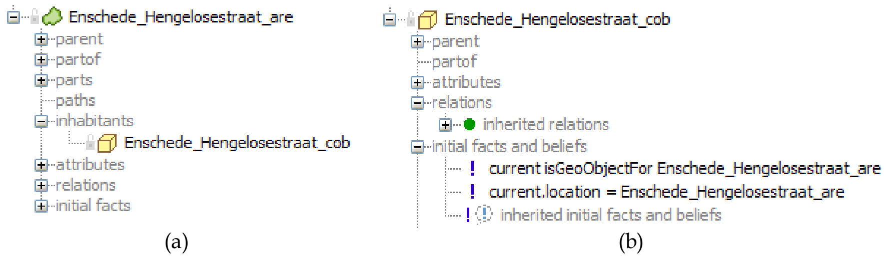


Figure 4.22: *Location information object*

The location object, furthermore, is provided with an initial fact (`current isGeographyObjectFor Enschede_Hengelosestraat_are`) defining that it is the object that is linked to a specific location (4.22(b)). The dynamic actual state of the location in parallel is modelled in the location object as facts. Since an object is able to engage in activities, the attribute value of the location (that is modelled as a fact in the location object) is able to change given that certain preconditions are met. The dynamical actual state of the location thus is modelled in a location object, while the static actual state of the location is modelled in the location instance itself. The location object thus can serve as a beacon, providing the current actual state about the location it represents to its inhabitants (agents and objects).

4.4.3 Movement

Section 4.4.1 provided the AWS template with the opportunity to model the locations that are used in communication during the emergency response and, furthermore, provided the general layout of an interconnected grid of locations in which objects and agents can be placed. Building on this grid, the dynamic and static attributes that are typically encountered in communication about locations during emergency response situations were implemented. This provided the emergency responder that is modelled in the AWS to assert, communicate about and influence, attribute values of the location.

In emergency response practice, emergency responders are able to acquire information about their surroundings by either communicating with other agents (or objects such as a computer) or by moving through the locations involved in the emergency and acquire the information first hand. In the AWS, the first situation can be incorporated by implementing a communication activity that is predefined in the Brahms software. This entails specifying the receiver, the duration, the priority of the communication and the beliefs that are communicated.

Movement also is incorporated in the Brahms software. By specifying paths between locations that take a certain amount of time to complete, one is able to specify the start and end location and let agents use these paths for movement. The Brahms engine automatically calculates which route is taken, optimizing the route based on the time it takes to get from start to finish. The agent remains at the start location for the duration of the move activity and then is put in the end location. While this method is suited for a large range of simulation purposes, it is a less suited method for the modelling of movement in emergency response situations.

As was mentioned earlier, moving from one location to another means that the emergency responder is able to acquire new information about the locations involved in the emergency. Information encountered can be of crucial importance, potentially changing the course of the emergency response. For example, in the exercise concerning the mock COPI emergency response to a major fire incident, the coincidental discovery of a fireworks depot changed the course of the emergency response. When an agent is “moving” from one location to another it therefore is crucial that the agent is able to encounter all information that is located in its path.

Furthermore, as was indicated in section 4.1.2 and section 4.4.1, emergency responders are often involved in (communication) activities while moving with the use of means of transport. These situated activities provide them with initial information about the emergency which is used to determine the initial plan of attack. By moving with a means of transport object, the emergency responder in essence is situated in multiple locations, being able to perform activities inside the means of transport objects and being able to acquire information about the areas the means of transport object moves through.

For the AWS to adequately simulate information acquisition of this type, it has to specify move activities in detail, enabling agents to move from one location to another (with or without a certain means of transport object), while being able to obtain information about the sub locations that are encountered which is provided by the location provider objects and the locations instances described in the previous sections.

The approach taken in the AWS is to specify in advance the paths that can be taken within the particular scenario that is modelled, with the inclusion of the sub areas that are encountered. The level of detail in which the paths are modelled can be determined by the focus of the simulation. The agent or object thus moves from sub area to sub area, acquiring location information about all these sub areas. A move activity typically consists of a composite activity in which multiple move activities are specified. Figure 4.23, for example, shows the composite activity that is used to model movement of agents and objects between the fire station located in Almelo centre the Netherlands and railway station “de Riet” in Almelo.

Figure 4.23: *Composite move activity.*

The activity of moving from the fire station to the railway station is chunked in multiple workframes, each incorporating a separate move activity with its own duration. As can be seen in the unfolded workframe representing the sub-move from the fire station to the Brugstraat, the duration is 20 seconds (the fire station is located in the Brugstraat).

This activity can be executed either by agents or by means of transport carrying agents. When the activity is executed using a means of transport object, the duration is adjusted to correct for differences in speed during the move activities. Furthermore, as indicated earlier, move activities with the use of means of transport deserve extra attention, enabling the agents contained by the object to be at two locations at the same time. Figure 4.24 shows part of a Brahms simulation output concerning location use while moving with a means of transport object.

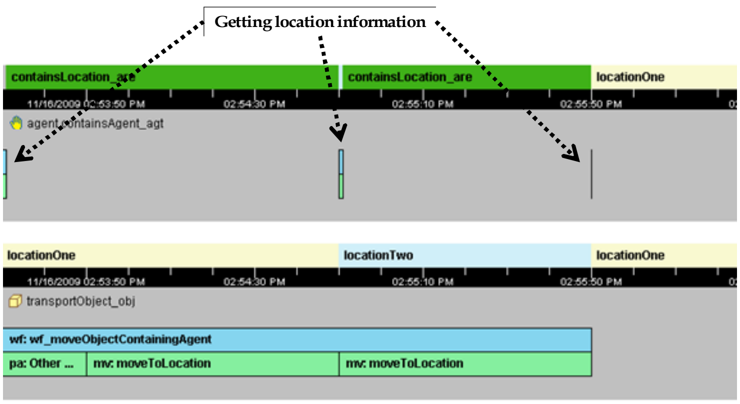


Figure 4.24: *Agent "being" at two places at the same time*

With the time line on the horizontal axes, the two large bars in Figure 4.24 show the activities that are executed by an agent (top bar) and a transport object (bottom bar). The location transitions of the agent and the object can be seen in the top half of each bar. In the transport object bar it can be seen that it moves from `locationOne` to `locationTwo` and back to `locationOne`. This fact that the object is moving is also reflected by the workframe it is involved with `wf:wf_moveObjectContainingAgent`. What not can be seen in this output is that the object contains the agent and that the inside of the object (the area where the agent resides) is named the `containsLocation_are`.

During the move activity of the object, the agent resides in the `containsLocation_are` location (in the transport object). However, when the object changes location the agents assert the facts that reside in the new location. The activity of "getting new location information" takes no time, just as looking out of the window does not take time. By incorporating this activity, the opportunity is created to model an agent that is able to perform tasks in the transport object, assert new information about the locations that pass by and simultaneously is moved by the transport object to the destination.

Summarizing, by specifying movement paths beforehand, performing at all sub locations the assessment of location information that exist in the sub location can be incorporated in the AWS. Furthermore, by positioning the agent in the location representing the means of transport, in combination with a "get information" activity when the means of transport object changes location, the modeller is provided with the opportunity to model location acquisition while the agent is being moved by the object and simultaneously performs other activities.

4.5 AWS Template conclusion

In this chapter the research questions posed in the start of the chapter were answered. These questions are:

- How can we construct a grounded AWS template that is able to function as a reusable template for emergency response modelling in Brahms?
  - Which classes have to be included in the AWS template for it to be a model of the reoccurring elements in emergency response?
  - How can the reoccurring elements be ordered using the inheritance structure?
  - What level of detail should the model have?

First, a grounded template of emergency response that is used as the main modelling architecture for the AWS simulation in which emergency response scenarios can be simulated was developed. This template consists of the reoccurring elements that are similar between emergency responses. These reoccurring elements relate to:

- The emergency itself
- The emergency response organisation
- The roles and activities that are performed by these organisations
- The agents that are involved in the emergency response
- The concrete objects that are used during the emergency response
- The geography in which the emergency unfolds

The reusability of a template is mainly determined by the level of detail that is incorporated in the template. Under specifying will lead to additional effort when building the actual model, while over specifying will increase the "payload" of the model, shifting effort towards the development phase with the danger of incorporating redundant elements. By using both empirical grounded data and literature on emergency response practice, the actual elements and the level of detail of these elements was determined.

Characteristics such as the type, severity and prospects of the emergency determine the size and the assembly of the emergency response organisation. Using data gathered concerning communication during two emergency response exercises, 10 attributes of the emergency itself were differentiated. In combination with 18 emergency types that could be identified in the literature, an initial set of attributes of the emergency was implemented in the AWS. Secondly, reflecting the emergency response organisational structure as it is seen in practice, the AWS incorporated multiple group membership (role, organisation, team) to structure the inheritance of attributes that propagate down from these groups. The emphasis on prescribing a detailed model of the stable group classes furthermore aids the modeller and can facilitate efficient modelling of instances. This is reflected in the limited number of attributes needed to model the actual emergency responders. The level of detail of the identifying and physical attributes of the emergency responders, was determined by adopting the attributes that are incorporated in the "standard" casualty card that is used by emergency personnel to describe the relevant medical and identifying attributes of a person.

Together with the workflow model that will be presented in Chapter 5, the workload model that will be presented in Chapter 6 and the communication load model that will be presented in Chapter 7, the template model will equip the AWS with the necessary elements making it possible to model a variety of emergency responses as a function of the information that is exchanged and can provide an approximation of the workflow and workload of the individual emergency responders.

5. Workflow Model

For capturing the dynamics of the workflow of emergency responders and predicting the task at hand by conducting work simulation, exception handling and task flexibility are key concepts. A rigid workflow representing the tasks that should be done only has power to represent part of the work that actually is done during mitigation. These textbook reactions, mostly documented in protocols (predefined plans of attack) and emergency plans, possess a fine level of granularity of workflows, but work practice often deviates.

This chapter presents and tests the workflow model that is used within the AWS, presenting a method to enable adaptive task simulation. Section 5.1 describes the general workflow model, linking the basic elements that are included within the AWS workflow representation. Next, in section 5.2, the implementation and applicability of the AWS workflow representation is presented. Finally, section 5.3 concludes on the viability and applicability of the AWS workflow model to model emergency response task ordering diversity.

The AWS workflow model is aimed at constructing a general on the fly workflow that consists of multiple parallel workflows of emergency responders, adaptively interacting to deal with the emergency and other emergency responders' workflows. The AWS workflow model thus constructs workflows at the emergency responder level, who are engaged in parallel activities, and by doing this, constructs the total workflow model of the emergency response.

5.1 Workflow Specifications

Emergency responders react to the situation at hand, changing the workflow due to new information provided to them in the form of situational cues or as a result of communication. By doing this, they restructure and combine parts from different workflows that are described in action plans. The actual workflow thus emerges on the fly. "Disaster plans are often written in the belief that people ought to behave according to the plan. The plans state what people "should do". A more successful approach is to design the plan according to what people are "likely to do" (Auf der Heide, 1989) To achieve this level of flexibility in a workflow system, the unit level of work should not be the complete, prescribed, rigid workflow, but the tasks of which the prescribed and observed workflow consists. The workflow model of emergency response therefore demands more than only a relative ordering of prescribed tasks. Incorporating the adaptive level of work practice of emergency responders into traditional workflow and simulation systems thus goes beyond the abilities of most systems used for adaptive workflow modelling (Kramer, Bolcer, Taylor, Hitomi, & Bergman, 2000; Berfield, Chrysanthis, & Labrinidis, 2004).

In Figure 5.1, the workflow model for an agent is shown as it is used within the AWS. It illustrates the dependency relations in the AWS workflow model between tasks, preconditions and priorities. The workflow model is based on the task as the unit level of work, with an illustrative *Task (B)* being the central task within the workflow. The relative order of tasks is that *Task (B)* is preceded by *Task (A)* and followed by *Task (C)*. The order

A-B-C represents the ordering of the tasks executed referring to workflow modelling such as it is used within business process modelling and reengineering (van der Aalst, ter Hofstede, Kiepuszewski, & Barros, 2003; Divitini & Simone, 2000; Klein, Dellarocas & Bernstein, 2000) or emergency management (de Muralt, 2007; Wang, Rosca, Tepfenhart & Milewski, 2006; Llavador, Letelier, Penadés, Canós, Borges & Solís, 2006).

The AWS supplements these types of workflow models with the principle that completing a preceding task does not have be the exclusive trigger for the next task, by adding situational workflow dependence to the workflow. *Task (B)'s* enabling conditions in Figure 5.1 are the preceding task (*Task (A)*) and the preconditions of *Task (B)*. The preconditions can be resource constraints, spatial constraints or knowledge constraints of the performer. Resource constraints refer to the availability of objects to be used and to be used up during task execution. Task execution cannot commence if these objects are not available. Spatial constraints refer to constraints put on task execution by the difference in location between the performer and the location where the task has to be performed. Finally, knowledge constraints of the performer refer to constraints put on task execution based on the availability of task performance knowledge or the availability of vital state of the world information needed to initiate a task.

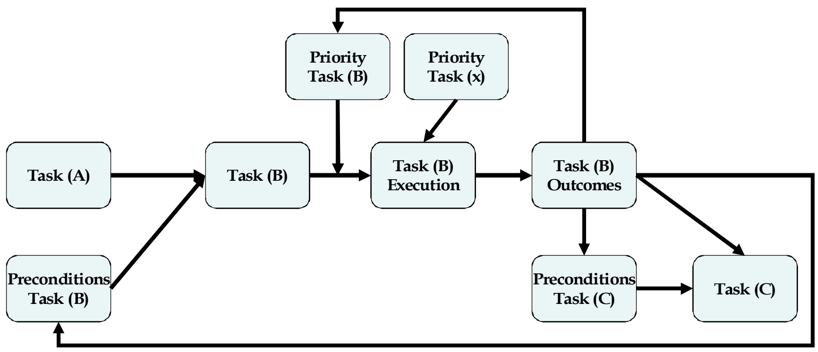


Figure 5.1: *Workflow specification of an agent within the AWS.*

These conditions determine if a task can be triggered in an actual situation. For the task "extinguish a fire", a fire-fighter first of all has to know that there actually is a fire (knowledge constraints). The fire-fighter furthermore needs to be at the actual location of the fire (spatial constraint) to start extinguishing. When the fire-fighter is not at the location of the fire, the fire-fighter first has to move to the fire location (task dependency). Finally, when the fire-fighter is at the location of the fire he/she has to possess fire extinguishing material to actually put out the fire (resource constraints).

Task (B) can only be triggered if the enabling conditions of *Task (B)* are met. However, it is not a necessity that every task is preceded by another task. Tasks can exist that only have specific preconditions and no "preceding task" conditions and vice versa. This increases the adaptive level of the AWS, since these tasks can be triggered at any moment when the preconditions are met. Furthermore, by specifying the relative task order by only specifying the preceding tasks as conditions, the standard textbook workflow can be modelled without affecting the adaptive level of the simulation.

When the task enabling conditions are met for a *Task (B)*, it is triggered (ready for task execution) and put on a “to do list”. Within the AWS triggering a task thus is not the same as executing a task. Task execution rather is phased, using a “to do list” as a stack of tasks that are ready for execution. During emergency response, multiple tasks from multiple workflows can be triggered simultaneously. Actual task execution order is determined by the priority of the task compared with the priority of other triggered tasks. When, for example, the priority of other triggered tasks is higher than the one for *Task (B)*, these tasks are executed first, delaying task execution of *Task (B)*. However, the execution of the higher prioritized tasks can affect the enabling conditions for *Task (B)*, resetting the triggering status of *Task (B)*, and disabling it for execution.

If all task enabling conditions for *Task (B)* are met and the priority of *Task (B)* is higher than other triggered tasks, the road is clear for task execution. After and during the execution of the task, this successively leads to task outcomes; the outcomes successively influence the values of the parameters of the preconditions of other tasks. When both *Task (B)* and *Task (C)* are dependent on one specific resource, such as fire extinguishing foam, and all of the foam is used up while carrying out *Task (B)* – which is the preceding task condition for *Task (C)* – then, even though *Task (B)* is completed successfully, *Task (C)* cannot be triggered since its “available foam quantity” precondition cannot be met. When *Task (B)*, for example, is executed without success, this also can influence its own priority and its preconditions (when materials for example only can be used once). Task execution can furthermore be interrupted by other tasks that are triggered and have a higher priority than *Task (B)*. *Task (B),* however, is not abandoned, but interrupted, enabling task switching. Tasks can be executed and reused multiple times in different contexts, by different executers. The task “extinguishing fire”, for example, only needs to be modelled once for it to be executed by all people that are able to “extinguish fire”, in all situations where the task is triggered.

Summarizing, a task is triggered when its enabling conditions are met and that task has the highest priority compared with other triggered tasks. When another task receives a higher priority during execution of the initial task, the initial task is interrupted. When executed, the outcomes of a task can influence the parameters of the preconditions of other tasks, with the consequence of triggering tasks or blocking triggering of other tasks..

The workflow representation within the AWS corresponds with requirements of an enriched workflow representation for emergency response postulated by Berfield et al. (2004), but also adds complementary / corrected specifications. According to Berfield et al. (2004), a workflow representation must include, with AWS additions specified:

- Pre conditions have to be specified as a condition for a task to be executed.
    - AWS supplement: pre conditions can find their origin in both the setting of the emergency response and the tasks performed.
- Post conditions have to be specified as the outcome of task execution.

  - AWS supplements the consequences of the task outcome of task execution on parameters of the variables in the precondition with influence on the priority of the current and future tasks.
- Causal links between the consequences of task execution and the task and between the preconditions and task execution have to be established.
  - AWS supplements this by adding causal links between tasks, being able to follow textbook procedures when the situation does not demand an adaptive workflow.
- In- and out-parameters have to be specified within the workflow model to test the parameter values within the preconditions.
- Temporal constraints have to specify the start and stop time of a task, and the duration of a task.
  - AWS uses both task and context information as the trigger for tasks within the workflow. Time is not used as a trigger for task execution. The start time therefore is relative, increasing the adaptive level of the simulation. The end time is determined by the duration of the task without interrupts. When interrupted, the end time of the interrupted task corresponds with the time taken until the interruption, plus the interruption time and the remaining duration of the interrupted task.
- Resource constraints have to be modelled that specifies the material and agents that are needed to fulfil a task.
- Significance of a task indicating that a task has to be executed within the workflow.
  - AWS supplements this with by adding the priority of the task as an indication that task can be vital within the workflow. The priority attribute can be set dynamically, changing the priority and workflow on the fly, increasing the adaptive level of the workflow.

Based on these requirements for a workflow model describing emergency response, it can be concluded that the general workflow model used within the AWS workflow model meets the requirements for an enriched workflow representation.

The workflow in emergency response practice is characterized by the combination of adaptive execution of elements of textbook workflows and the execution of emergent work. From the workflow model presented in Figure 5.1, four basic workflow types can be derived into which the AWS workflow model can become simultaneously engaged in: the textbook workflow, the precondition workflow, the priority workflow and the interrupt workflow, describing the different relations between tasks within the total workflow.

Textbook workflows describe a workflow where tasks follow each other in a predefined order. The predefined textbook order describes a situation where a certain task always is followed by another task and task order cannot be changed. As can be seen in Figure 5.2, *Task(1)* is followed by *Task(2)*, that in turn is followed by *Task(3)*. Execution of *Task(1)* is an enabling condition for *Task(2)* and execution of *Task(2)* is an enabling condition for *Task(3)*. The textbook workflow enables the AWS to cope with rigid elements in the adaptive simulation.

The second workflow type describes a workflow that is similar to the textbook workflow in the sense that task order is rigid. The difference between the textbook workflow and this workflow resides in the fact that other task enabling conditions exist that are influenced by the preceding tasks. In Figure 5.2, this is illustrated by workflow 2: the *Precondition workflow*. *Task(1)* is followed by *Task(2)*, that in turn is followed by *Task(3)*. In correspondence with the textbook workflow *Task(1)* is an enabling condition for *Task(2)* and execution of *Task(2)* is an enabling condition for *Task(3)*. The precondition workflow however, deviates from the textbook workflow in the sense that there is another enabling condition that is influenced by the execution of the tasks in this workflow. *Task(1)* affects an enabling condition of *Task(2)*, and *Task(2)* affects an enabling condition of *Task(3)*. The precondition workflow enables the AWS to cope with rigid elements within the adaptive simulation that are dependent on a shared resource, situations where task outcomes deplete resources needed or when influencing enabling conditions for the following task are influenced by the preceding task. When the shared resource is depleted, the following task cannot be executed since its enabling conditions are not met (sufficient resources), stopping the task sequence. However, when the enabling conditions are met, the task sequence can proceed.

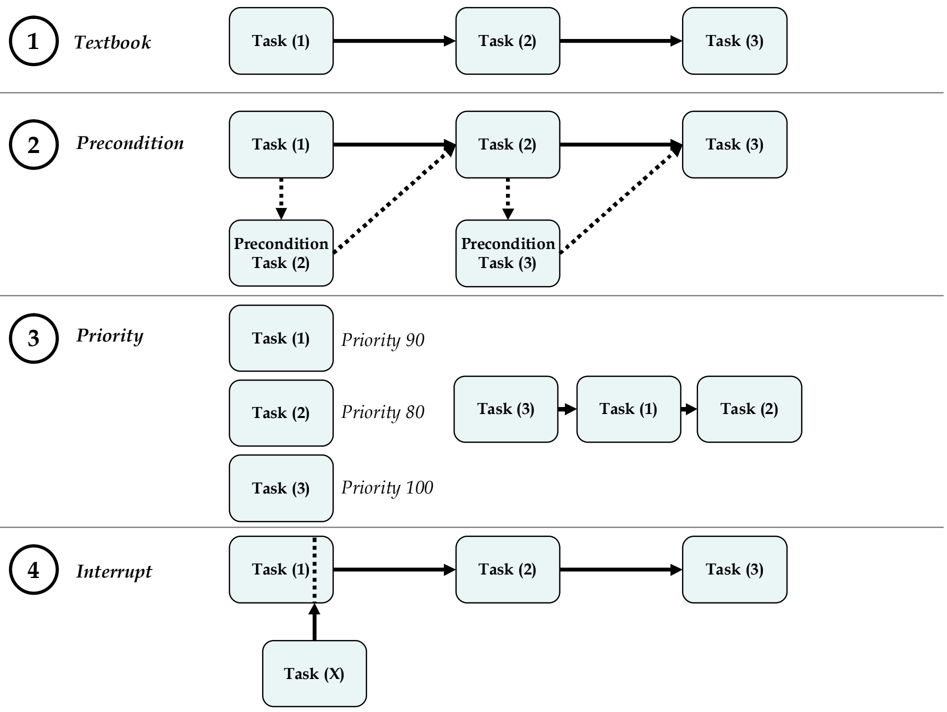


Figure 5.2 : *Relations between tasks within the workflow model*

The third workflow type, *the Priority workflow*, shown in Figure 5.2, refers to situations where multiple tasks are triggered that differ in their assigned execution priority. *Task(3)* has the highest priority and is executed first, *Task(1)*, with the second highest priority, is executed second and *Task(2)*, with the lowest priority is the last task that is executed. The priority workflow enables the AWS to adaptively structure tasks based on on the fly changes in task priority. Task execution with multiple triggered tasks with the same priority is based on a first come first serve basis, based on the order in which the enabling conditions of the tasks were met.

When the task with the highest priority is being executed and a new task arises that has a higher priority, the current task is interrupted by the new higher priority task. When the interrupting task is finished, the interrupted task can be taken up again, or be abandoned. The *Interrupt workflow* type that is indicated with 4 in Figure 5.2, enables the AWS to deal with immediate high priority task execution (such as an evacuation) in the simulated scenario, overruling all other tasks. Figure 5.2 illustrates the interrupt workflow by interrupting *Task(1)* with *Task(X)*. The total workflow duration when task execution of *Task(1)* is taken up again is prolonged with the duration of *Task(X)*.

The AWS thus structures the tasks based on a predefined task order, the enabling conditions, first come first serve and task priority enabling it to deal with both the rigid structures in the workflow as well as a highly adaptive workflow situation. Applied to emergency response, it can be stated that the on the fly character of the work within an emergency response has to deal with sets of tasks being triggered, that correspond with the scenario at hand, some with higher priorities than others (workflow type 3). The order in which these tasks are executed deviates from person to person (workflow type 3), but some tasks are always performed in a predefined manner (workflow type 1), but can have interdependencies with other tasks (workflow type 2). When an urgent situation arises, the emergency responder has to react immediately, stopping his present task and starting the new task (workflow type 4).

In the upcoming section (5.2), the implementation of the workflow model in the AWS is presented.

5.2 AWS Workflow Model Implementation

The workflow model implementation within the AWS uses the hierarchal inheritance structure of the Brahms modelling environment, as described in Chapter 4, to make the task planning activities available for all agents within the model. Differences however can exist at the agent level, enabling parallel agent specific workflow formation. This section presents the workflow planner used within the AWS and will, by addressing agent differences, test if the implemention of the four basic workflow types is successful in representing the diversity in emergency response workflows in practice.

5.2.1 AWS Workflow Planner

The hierarchical structure within the Brahms modelling and simulation environment allows the modeller to specify activities and attributes that are shared by all agents within the simulation. Using this principle, within the AWS each agent is able use the AWS workflow planner to schedule activities. The scheduling is based on agent specific beliefs. This allows agents to perform tasks in a decentralized manner while accessing the centralized AWS workflow planner.

For the AWS workflow planner to be able to create an on the fly workflow, information about the active tasks, the triggered tasks and task priority of these task provide the minimally needed information. These task attributes provide the AWS workflow planner with the information needed to construct the task structure and workflow. Active tasks are the tasks that are currently performed (`currentTask`). With the exception of the interrupt workflow type, this binds only to one specific activity. This is the actual task that is worked on by the agent. Triggered tasks represent the task of which the enabling conditions are met, but that are waiting to be performed (`nextTask`). This could be seen as a task stack or the "to do list" of a particular agent. Finally, task priority (`taskPriority`) refers to the urgency task execution has.

Residing in the `SimBaseGroup`, the workframe presented in Figure 5.3 is the code needed for planning the workflow types 1-2-3 shown in Figure 5.2. To incorporate the interrupt workflow, a separate workframe is written. The code needed for the interrupt workflow is shown in Figure 5.4. The combinations of these planning activities cover the needed workflow types to model emergency response.

```
workframes:
   workframe wf_planning {
      display: "Select task with highest priority";
      repeat: true;
      priority: 1010;
      variables:
         collectall(string) snextTask;
         forone (int) pri;
         forone (int) pri0;
      when (
            knownval (current.nextTask(snextTask)=true) and
            knownval (current.taskPriority(snextTask)= pri ) and
            knownval (current.nextCandidateValue = pri0) and
            knownval (pri0 < pri))
      do {
            conclude ((current.nextCandidate = snextTask), bc:100, fc:0);
            conclude ((current.nextCandidateValue = pri), bc:100, fc:0);
         }
   }

   workframe wf_planning02 {
      display: "Initiate task with highest priority";
      repeat: true;
      priority: 9;
      variables:
         forone (string)snextTask;
         forone (string)snextCandidate;
      when (
            knownval (current.nextTask(snextTask)=true) and
            knownval (current.nextCandidate = snextCandidate) and
            knownval (snextTask = snextCandidate))
      do {
            conclude ((current.nextTask(snextTask) = false), bc:100, fc:0);
            conclude ((current.currentTask(snextTask) = true), bc:100, fc:0);
            conclude ((current.nextCandidateValue = 0), bc:100, fc:0);
            conclude ((current.nextCandidate = unknown), bc:100, fc:0);
         }
   }
```

Figure 5.3: *AWS workflow planner*

The rationale behind the AWS workflow planner presented in Figure 5.4 is that when a person knows that he/she has an activity on the “to do list”, he/she knows the priority of that task and he/she is not performing any other task at that moment, the task on the “to do list” is removed from the “to do list”and is performed. Which task is removed from the “to do list” is based on the priority of that task and the first come first serve principle. The AWS workflow planner repeats this activity until there are no more tasks on the “to do list”.

For a task to interrupt an activity that is performed, it has to be a task that is on the “to do list” and have a higher priority than the task that is currently performed. The workframe that is added to the AWS workflow planner is an activity that monitors the activities in the “to do list” and activates emerging tasks that have a higher priority than the task that currently is performed, causing the higher priority task to interrupt the current task.

```
workframe wf_planning03 {
   display: "Interrupt workflow";
   repeat: true;
   priority: 1009;
   variables:
      forone (string)snextTask;
      forone (string)snextCandidate;
      collectall (string)scurrentTask;
      forone (int) pri;
      forone (int) priNext;
   when (knownval (current.currentTask(scurrentTask) = true) and
         knownval (current.taskPriority(scurrentTask) = pri) and
         knownval (current.nextCandidate = snextCandidate) and
         knownval (current.nextCandidateValue = priNext) and
         knownval (priNext > pri))
   do {
         conclude ((current.nextTask(snextCandidate) = false), bc:100, fc:0);
         conclude ((current.currentTask(snextCandidate) = true), bc:100, fc:0);
         conclude ((current.nextCandidateValue = 0), bc:100, fc:0);
         conclude ((current.nextCandidate = unknown), bc:100, fc:0);
      }
}
```
Figure 5.4: *AWS workflow interrupt.*

As shown in the code in Figure 5.4; when a person is performing a task (`current.currentTask(scurrentTask) = true`), he knows the priority of the task he is performing (`current.taskPriority(scurrentTask) = pri`) and he has tasks on his “to do list” (`current.nextCandidate = snextCandidate`) that have a higher priority (`current.nextCandidateValue = priNext`) than his current task (`priNext > pri`), the task with the higher priority is removed from the “to do list” (`current.nextTask(snextCandidate) = false`), and is performed (`current.currentTask(snextCandidate) = true`), causing an interrupt of the current active task.

The AWS workflow planner initiates tasks that are triggered, and reacts to the beliefs specified at the agent level. Section 5.2.2 discusses the belief differences that are used as input for the AWS workflow planner to elicit the different workflow types specified in section 5.1.

5.2.2 AWS Workflow Model Test

To test if the workflow types specified in section 5.1 can be represented by the AWS using the AWS workflow planner presented in section 5.2.1, the workflow types were modelled and simulated using the Brahms modelling and simulation environment. Within this environment four different agents were created that are engaged in three tasks (*Task1*, *Task2* and *Task3*). Depending on the beliefs an agent has about the tasks and agent specific differences as consequences of task execution, the agent must become engaged in one of the four workflow types. In Figure 5.5, the initial beliefs of the agents at the start of the simulation are shown and Figure 5.6 shows the initial workframe that initiates execution of *Task1* and determines task outcomes. A similar set of beliefs and workframes was specified for each agent and each task. Variations within these initial beliefs and task outcomes determine the type of workflow that emerges.

```
initial_beliefs:

    (current.taskPriority("Task 1") = 100);
    (current.taskDuration("Task 1") = 30);
    (current.previousTask("Task 1") = false);
    (current.currentTask("Task 1") = false);
    (current.nextTask("Task 1") = false);

    (current.taskPriority("Task 2") = 100);
    (current.taskDuration("Task 2") = 30);
    (current.previousTask("Task 2") = false);
    (current.currentTask("Task 2") = false);
    (current.nextTask("Task 2") = false);

    (current.taskPriority("Task 3") = 100);
    (current.taskDuration("Task 3") = 30);
    (current.previousTask("Task 3") = false);
    (current.currentTask("Task 3") = false);
    (current.nextTask("Task 3") = false);
```

Figure 5.5: *Initial agent task beliefs*

Concerning the initial task beliefs, per task the agent has a belief about the priority of that particular task (`taskPriority`), about the duration of the task (`taskDuration`), if the task was already executed by the agent at an earlier moment (`previousTask`), if the task is an active current task (`currentTask`) and if the task is on the “to do list” of the agent (`nextTask`). For this testing scenario, all tasks shown Figure 5.5 have arbitrary values assigned for priority and duration and true/false values set to false. The tasks have an initial priority of 100, have a duration of 30 seconds, are not performed by the agent, are not active and are not on the “to do list”.

```
workframe wf_Task1 {
   display: "executing Task 1";
   variables:
      forone (int) pri;
      forone (int) dur;
   when (knownval (current.currentTask("Task 1") = true) and
         knownval (current.taskPriority("Task 1") = pri) and
         knownval (current.taskDuration("Task 1") = dur))
   do {
      Task1(pri, dur);
      conclude ((current.currentTask("Task 1") = false), bc:100, fc:0);
      conclude ((current.previousTask("Task 1") = true), bc:100, fc:0);
   }
}
```
Figure 5.6: *Initial workframe used for task execution*

The workframe shown in Figure 5.6 describes the enabling conditions and consequences of *Task1*. Separate similar workframes do exist for *Task2* and *Task3*. The enabling conditions consist of a person knowing that *Task1* is the current active task (`current.currentTask("Task 1") = true`) and that the priority (`current.taskPriority("Task 1")`) and duration (`current.taskDuration("Task 1")`) of *Task1* is known. *Task1* can only be executed if it first was triggered, had the highest priority of all triggered tasks (or was the first on a first come first serve basis compared with other same priority tasks) and if the `current.currentTask("Task 1")` value was set to true by the AWS planning tool. Since the enabling conditions about the priority and duration are part of the agent's initial belief set, the `current.currentTask("Task 1")` in essence is the only enabling condition that differentiates between active and non active tasks.

When these enabling conditions are satisfied, *Task1* is executed with the priority and duration values set in the enabling conditions. After task execution, the `current.currentTask("Task 1")` value is set to false and its `previousTask("Task 1")` value is set to true, indicating that *Task1* is not performed anymore, but was performed during the simulation at a certain time.

The next sections will show how variations within the initial task beliefs, task enabling conditions and task consequences can result in instantiations of the four workflow types discussed in section 5.1.

5.2.2.1 Textbook Workflow

The textbook workflow represents connected tasks that have to be executed in a fixed order. Triggering only one task is sufficient to initiate the connected tasks after execution of the first. In this example, triggering of *Task1* should result in a chain of tasks, *Task1*-*Task2*-*Task3*. Within the AWS this corresponds with setting `current.nextTask("Task 1")` to true. This triggers the AWS workflow planner and sets `current.currentTask("Task 1")` to true, initiating *Task1*. When *Task1* is completed, it is concluded that the `current.nextTask("Task 2")` is true, initiating the AWS workflow planner again, resulting in execution of *Task2*. The final task does not set the `current.nextTask` value, halting the simulation.

The textbook workflow can thus be initiated by setting the `nextTask` value of the first task to true and as a consequence of task execution setting the `nextTask` value of the following task to true.

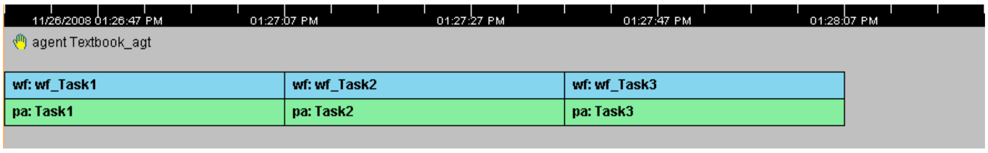


Figure 5.7: *Textbook workflow*

Figure 5.7 shows the graphical output of the agent performing the textbook workflow. As expected, first *Task1* is performed, then *Task2* is performed and finally *Task3* is performed, correctly representing the textbook workflow type.

5.2.2.2 Precondition Workflow

The precondition workflow task ordering rules are similar to the ones used in the textbook, priority or interrupt workflow. It however differs by the fact that the tasks contain an enabling condition that is affected by the other tasks. The demonstration of the precondition workflow within the AWS is based on the textbook workflow presented in the previous section, which results in a *Task1-Task2-Task3* workflow.

To test the precondition workflow, a precondition belief was modelled to function as an extra enabling condition for the tasks. The added enabling condition represents a resource depletion situation, where execution of a task consumes a particular quantity of a resource and thus needs to have access to that quantity before the task can be executed. Execution of a task thus influences the possibility for execution of a subsequent task that uses the same resource.

To test if this can be modelled with the AWS, the initial value of a precondition belief was set to 10 units and was depleted with 3 units when a task was performed. In the enabling condition of the tasks the rule was entered that it can only be performed if, after the task would be performed (looking ahead), the value would be greater than 3 units. This rule results in an impasse of *Task3* since *Task1* and *Task2* used 6 units out of the 10 units and, when looking ahead, the value after execution of *Task3* would be 4-3=1, which is smaller than 3. *Task3* is not triggered, but remains an active current task and shall be started when the precondition value after some task execution will be greater than 3 units.

The precondition workflow thus can be initiated by specifying an enabling condition that is influenced by preceding tasks.

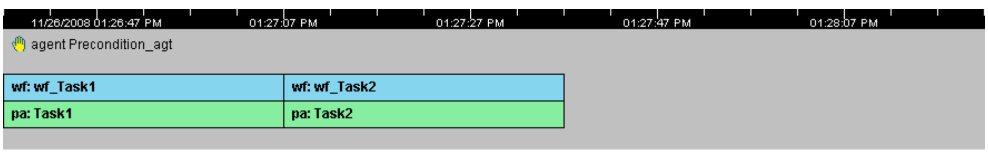


Figure 5.8: *Precondition workflow*

Figure 5.8 shows the graphical output of the agent performing the precondition workflow. First *Task1* is performed, then *Task2* is performed and *Task3* is not performed due to the fact that the enabling conditions are not met. The AWS correctly represents the precondition workflow type.

5.2.2.3 Priority Workflow

The priority workflow type represents a workflow where task arrangement is based on the relative priority of the tasks. Within the priority workflow, several tasks are on the "to do list" (triggered) and are awaiting execution. When multiple `nextTask` values are set to true, two basic situations can come into existence. Firstly, priorities (`taskPriority`) can differ between all tasks on the "to do list", and secondly, some or all tasks on the "to do list" can have the same priority.

To test the first situation, *Task1* was given a priority of 90, *Task2* was given a priority of 80 and *Task3* was given a priority of 100. The `nextTask` values of these tasks were set to true. The expected resulting workflow is *Task3-Task1-Task2*.

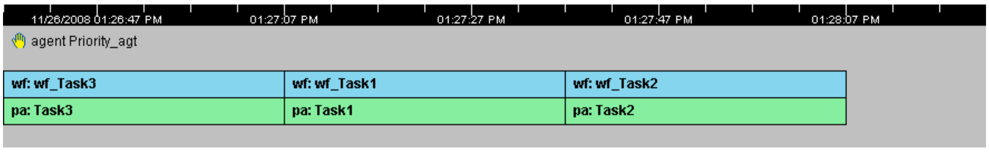


Figure 5.9: *Priority workflow with different priority tasks*

Based on the simulation results the graphical output of the simulated run that is shown in Figure 5.9, illustrates that the AWS can correctly represent the priority workflow based on tasks with different priorities.

When structuring tasks based on tasks having the same priority, the AWS on the fly selects and executes a task from the "to do list" until there are no more tasks on the "to do list" to work on. Task selection is based on a first come first serve basis, depending on the position of the modelled task in the code. The expected workflow in this situation is *Task1-Task2-Task3*.

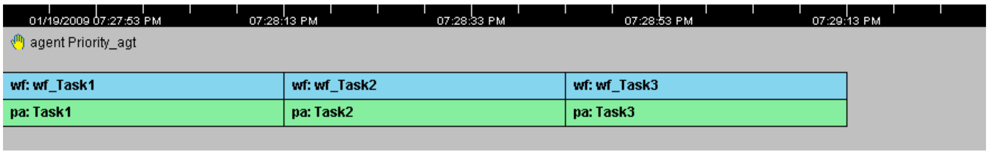


Figure 5.10: *Priority workflow with same priority tasks*

Figure 5.10 shows the graphical output of the simulated run based on a priority workflow with same priority tasks that uses the first come first serve rule.

Based on this output, it can be stated that the AWS correctly represents the priority workflow type with similar and different priorities within the triggered tasks. The priority workflow can be initiated by setting or changing the priority values of the tasks.

5.2.2.4 Interrupt Workflow

The interrupt workflow type represents the situation where more urgent tasks arrive on the "to do list" than the task that is currently performed and this task needs to be interrupted by the new high priority task.

In order to test if this type of workflow can be represented in the AWS, the textbook workflow is used to model a series of tasks. The order in which the tasks are performed is *Task1-Task2-Task3*. While performing *Task1*, *TaskX* emerges on the "to do list" that has a higher priority than Task1 and causes an interrupt.

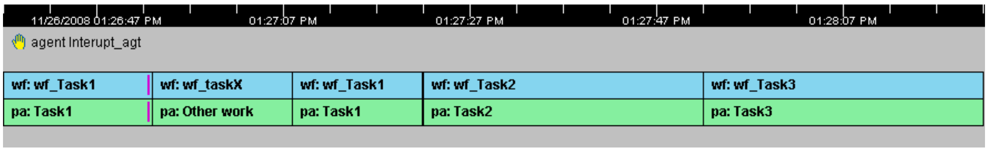


Figure 5.11: *Interrupt workflow*

As shown in Figure 5.11, execution of *Task1* is interrupted by *TaskX*. After *TaskX* is finished, *Task1* continues with its execution of the remaining part.

An interrupt workflow can thus be initiated by letting a textbook, priority, or precondition workflow type being interrupted with a higher priority task on the "to do list" during task execution.

5.3 Conclusion

Based on the simulation results shown in sections 5.2.2.1 to 5.2.2.4, it can be concluded that the AWS workflow model can sufficiently represent the workflow types presented in section 5.1 that are needed to make an enhanced workflow model for tasks that are performed during emergency response.

It is shown that the AWS workflow model is able to deal with the adaptive and agile work structuring that is commonly seen in emergency response practice. The AWS workflow model can structure agent activities based on agent specific variations in the enabling conditions (precondition workflow), incorporate rigid elements in the simulation by including the possibility to use predefined task execution order (textbook workflow), and can use the subjective priority of tasks to structure activities (priority workflow) or interrupt to interrupt the workflow (interrupt workflow). These workflow types furthermore exist simultaneously within the AWS, making a fusion of or combinations between these workflow types possible on the fly.

The generic character of the AWS workflow model, which is used and reused by all agents representedin the model, provides us with the opportunity to use the AWS workflow model in an incident independent manner. The AWS workflow model -just as the AWS template model presented in Chapter 4- can be used and reused in simulations of different types of emergency responses.

To be able to test the applicability of the AWS workflow model to form a correct and valid representation of an adaptive workflow as it is seen in practice, the AWS workflow model is tested using empirical data. The outcomes of these tests are shown in a proof of concept simulation incorporating all aspects of the AWS in Chapter 8.

6. Workload Model

The adaptive workflow model presented in Chapter 5, in combination with the emergency template presented in Chapter 4, provides the AWS with a template based simulation of emergency response that incorporates the multiple protocols that can be active concurrently in combination with situational adaptive task execution. However, as mentioned in the introductory chapter, the price paid for flexible restructuring of work in emergency response practice, lies in the proportion of effort spent on communication and the possibility of an unbalanced distribution of workload. This chapter will present and test the workload model used in the AWS. The goal of the workload model is to monitor the impact of task execution on the available processing capacities of the emergency responders modelled in the adaptive simulation. Together, the workload model and the model concerning the impact of the information that is exchanged during emergency response (Chapter 7), are used to approximate the available processing capacity of the emergency responders .

The research questions that will be addressed in the chapter are:

- How can we build a generic grounded model of the workload, processing capacity and the influence of workload and processing capacity on the workflow that lets us on the fly approximate the workload and processing capacity of individual emergency responders in the AWS?
  - Which theoretical concepts and relationships underlie workload and processing capacity?
  - How do these concepts and relationships behave in emergency response practice?
  - How can we implement these concepts and relationships in the AWS in a generic fashion?
  - How does the workload model influence the workflow of the individual emergency responder?

This chapter answers these questions by first addressing the relationship between workload, processing capacity and performance (section 6.1). Next, the factors that make up and define workload will be presented (section 6.2), leading to the workload model used in the AWS. Finally, based on the results of the questionnaire administered to 26 experts in the field of emergency response, section 6.3 presents the quantifiable grounded model and its implementation within the AWS.

6.1 Workload, Processing Capacity and Performance

Workload refers to the amount of expended capacity caused by task execution (O'Donnell, & Eggemeier, 1986) and is determined by the relationship between the available resources of the performer and the demands posed upon the performer by the task (Wickens & Hollands, 1999). Figure 6.1 is an illustration of the relationship between the resources demanded by a task (task load) on the horizontal axis and the available resources of the performer for task execution on the vertical axis (based on Wickens & Hollands (1999)).

Workload thus refers to the performer's allocated resources for task execution in order to meet the resources demanded by the task. When the resources demanded by the task and the workload of the performer are the same, the "demand = supply" line emerges (workload = task load). The performer is able to allocate the exact amount of resources demanded by task, regardless of the difficulty of the task. In this situation, workload is equal to the demands posed by the task. In Figure 6.1, given a task demand of X, the workload of the performer is represented by line C.

The resource demand-allocation relationship however, firstly is not assumed to be linear and secondly is influenced by the limited capacity of the performer to allocate resources. Non linearity refers to the sub optimal allocation of resources by the performer. In order to adequately perform task X, line C is the minimal amount of resources that have to be allocated. Allocating less will result in performance decline. To avoid the risk of performance decline, due to, for example, small fluctuations in the demands, the performer will tend to slightly over allocate resources, represented by line B. Given the assumption of non-linearity, actual workload is slightly higher than the optimal "demand = supply" relationship. For task X in Figure 6.1, the workload of the performer is represented by line C ("demand = supply") + line B (over allocation of resources).

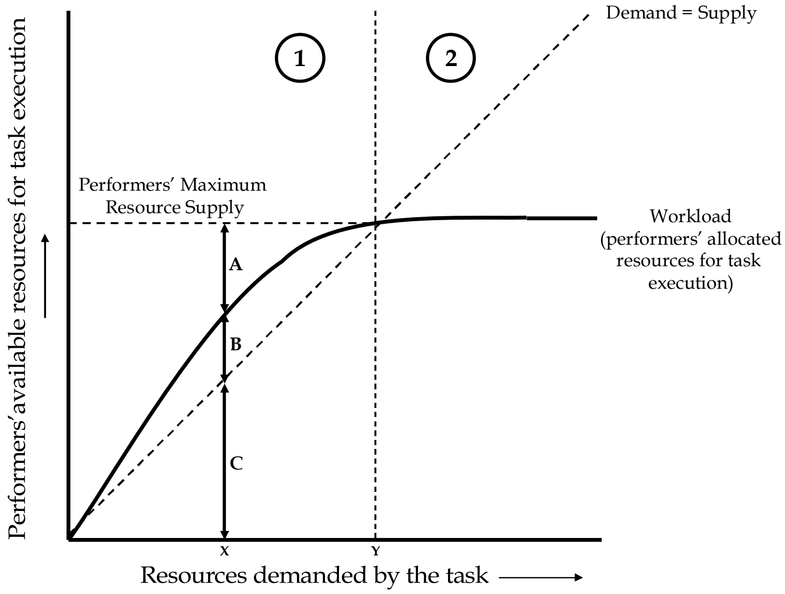


Figure 6.1: *Relationship between resources demanded by a task and resources supplied by the performer based on Wickens & Hollands (1999)*

In the first area (Area 1), the resources that are allocated by the performer (B + C) exceed the resources demanded by the task (C) and are less than the resources that maximally can be supplied by the performer (A + B + C) to adequately perform task X. In Area 1, the performer experiences rest capacity. The amount of rest capacity is represented in Figure 6.1 by line A. Rest capacity (A) refers to the performer's maximum resources that can be supplied (A + B + C) minus the resources supplied to task execution (B + C).

Up until the break-even point, in which task demands are equal to the maximum resources that can be allocated by the performer, the performer experiences rest capacity. The performer is able to allocate more than sufficient resources to meet the demands posed upon the performer by the task. A task, therefore, can be adequately performed in Area 1. The second area in Figure 6.1 (Area 2), on the other hand, represents a state of affairs where the task demands exceed the resources that can be allocated by the performer for the task to be adequately performed. In Area 2, the performer will allocate maximum resources for all tasks whose task demands surpass the task demands of task Y (where task demands are equal to the maximum resource that can be supplied).

When task demands exceed the maximum resources that can be supplied by the performer, the performer will experience maximum workload and increasingly decreased task performance (speed, accuracy, success). Performance is not affected when the resources that can be allocated exceed the task demands, but does affect performance negatively when task demands exceed the maximum resources that can be allocated by the performer.

According to Wickens & Hollands (1999), workload thus is determined by the interplay between the task demands and the performer's ability to allocate sufficient resources for task execution. The availability of resources to the performer is addressed by the concept of limited processing capacity (Kahneman, 1973), which states that an organism has a general reservoir of resources (processing capacity) that can be drawn from to address task demands. It should be noted that the processing capacity is not equal to the performer's maximum resource supply, which is shown in Figure 6.1. The maximum resource supply refers to the maximum resources that can be allocated by the performer for a single task, while the processing capacity refers to an overall reservoir of resources used for execution of multiple tasks. Thus, while the performance model of Wickens & Hollands (1999) addresses *single task performance*, the limited processing capacity model by Kahneman (1973) addresses *performance of a set of tasks*.

In the latter model, a decrease of processing capacity is due to the cumulative historic workloads posed by executed tasks on the performer, resulting in an amount of processing capacity that is available for remaining task executions (available processing capacity). This relationship is expressed by the following formula:

$$APC_{t0} = APC_{start} - (WL_{t-1} + WL_{t-2} + \dots + WL_{t-n})$$

The present available processing capacity (APC_{t0}) is a result of the available processing capacity before task executions (APC_{start}) minus the sum of the workloads of the executed tasks up until that moment. WL_{t-1} thus refers to the workload caused by execution of the previous task, and WL_{t-2} to the workload caused by execution of the task before. Depending on the task execution sequence of interest, the value of the APC_{start} can be either the initial starting processing capacity (before actually starting task executions) or the available processing capacity at a given moment in time. The available processing capacity thus is an indication of the cumulative workload allocated to all, or a subset, of past task executions by the performer.

Besides providing an indication of historic demands posed on the performer, the available processing capacity also provides an indication to what extent future task demands can be met and gives an indication of overall future performance. The available processing capacity in this sense can be related to fatigue; the decrease of the available resources that can be allocated to task executions.

The hypothetical relationship between the available processing capacity and overall performance is illustrated in Figure 6.2. Task execution typically starts in Area 2. In this area the available processing capacity for task execution exceeds the resources demanded by the task. Task demands in this area can be adequately met by the performer. Overall performance is not affected, and remains high, when the available processing capacity is higher than the depletion level (0).

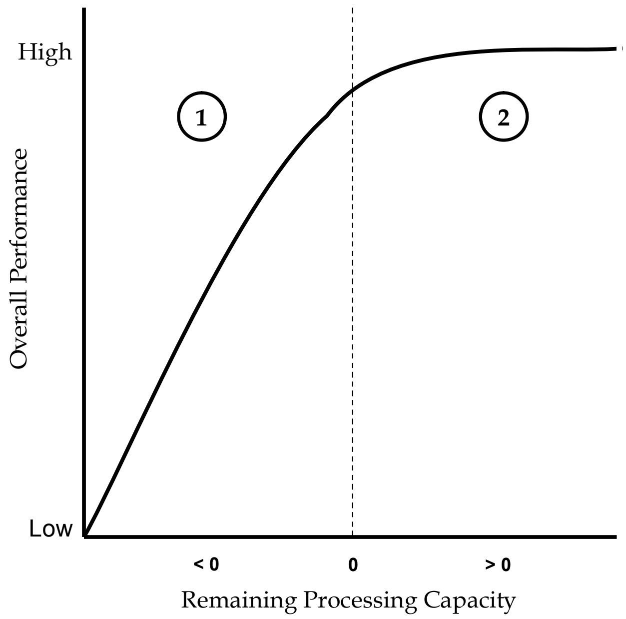


Figure 6.2: *Hypothetical relationship between a task performer's remaining resources for task execution and overall performance.*

Due to the cumulative impact of task executions, the depletion level of the processing capacity can be reached. When a task is executed at this point, the available processing capacity will get into the red (Area 1), resulting in decreased overall task performance. The performer exceeded his capacity at this moment.

After exhausting the available processing capacity, the performer needs to recover the processing capacity in order to be able to execute new tasks. Referring to fatigue, the performer has to recover from the impact of the task. The processing capacity, after exhausting it, is not immediately set to "full", but needs some "refill" time. When the remaining processing capacity exceeds the depletion level again, task execution can resume. Immediate execution of concurrent tasks, however, will lead the performance capacity getting in the red again, resulting in the needed recovery afterwards. High work rates with minimal recovery, will thus result in an accumulation of needed recovery times and prolonged overall duration of tasks. Furthermore, the magnitude by which the performer exhausts his capacity, determines the overall performance degradation. Overall, performance degrades more when performers exhaust their depleted processing capacity with a high workload task than with a low workload task, since the processing capacity needs to be refilled after task performance, and refilling duration increases due to the magnitude to which one exhausts the processing capacity.

Summarizing, workload is the result of the interplay between the demands of a task and the resources that can be allocated by the performer for that task. Due to the workload associated with past task executions, the processing capacity declines and due to recovery the processing capacity is able to recuperate. Workload, in this sense, is a portion of a person's limited capacity needed to perform a task. To assess the size of this portion, one has to know to what extent the task poses a demand on the performer and, secondly, to what extent the individual performer can comply with the various demands.

The demanded resources by the task, stem directly from mental task complexity and physical demands. In a literature review on task complexity, Campbell (1988) identified four fundamental task attributes that increase the mental task demands: presence of multiple paths to a desired end-state; presence of multiple desired end-states; presence of conflicting interdependence, and presence of uncertainty or probabilistic linkages. The degree in which one or more of these attributes are present within a task, determines its mental complexity and, consequently, the mental resources it demands from its performer.

Physical demands refer to the physical-loading factors in the work situation responsible for physical responses (de Zwart, Frings-Dresen, & van Dijk, 1995) and are expressed by the attributes of the task that evoke the response. The attribute of a weight that has to be lifted in this sense is an indicator of the physical demands posed by the "lifting" task. Lifting a rock weighing 10kg is considered easier than lifting a rock weighing 100kg. Furthermore, lifting the 10kg rock more than once poses more physical demands than lifting it once. The number and the quantity in which the physical loading factors are present, thus determine the physical demands of a task.

While both mental and physical task demands can relatively easy be determined by characteristics of a task, there is difficulty in assessing the resources that are allocated by the task performer. One can simply not tap into the processing capacity to determine the difference between the level of available processing capacity before and after task execution. Furthermore, although workload is a direct effect of objective demands posed by the tasks, inter-performer differences exist that determine the actual workload of the performer. In contrast to the task demands, the workload thus is hard to assess and is variable between performers given the same task. However, an estimation of the workload can be made by methods and theories that are used to provide an approximation of the workload of the performer. These give an indication of the elements that make up inter-performer workload differences. Section 6.2 will present the elements that are seen as most influential on the experienced workload.

6.2 Workload Influencing Factors

Kahneman (1973) points out that the impact on the performer due to task execution (workload) on the reservoir (processing capacity) fluctuates due to moderating elements that reside within the task execution domain. This view is supported by Parasuraman & Hancock (2001) and Hilburn & Jorna (2001), who state that workload is a subjective experience of the performer, caused by the universal load of the task (task load) and personal characteristics and external task environmental characteristics that moderate the relationship between the universal load of the task and workload (see Figure 6.3), such as skill, training, experience, fatigue or time pressure.

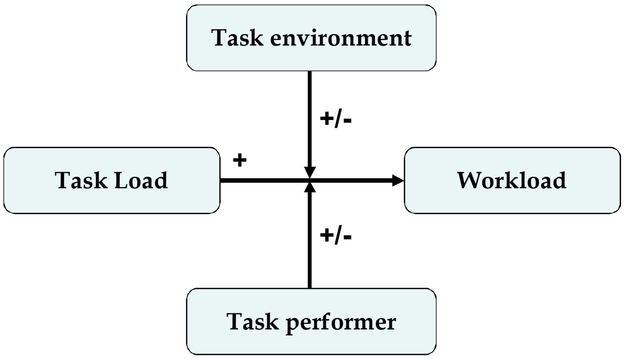


Figure 6.3: *Illustration of the relationship between task load and workload that is moderated by characteristics of the task environment and characteristics of the task performer.*

As illustrated by Figure 6.3, as a consequence of an increasing task load, workload of the performer will also increase (following the curve shown in Figure 6.1). The workload of the performer, however, also depends on the characteristics of the task environment in which the task is executed, and the characteristics of the performer who executes the task. These characteristics can both increase or decrease the workload for a performer.

Based on this distinction, this section will further specify task load and the person and task environment characteristics that moderate the relationship between task load and workload. This is needed for determining what elements should be included in the AWS to monitor the workload of the emergency responders in the simulation model. This will lead to the model for workload estimation to be described in section 6.3.

6.2.1 Task Load

The first element that has to be included in the workload model is task load. Task load refers to the demands posed by the task on the performer when performing the task, which is in line with the Wickens & Hollands (1999) task demands. In this dissertation, following the viewpoint of Hilburn & Jorna (2001) and Campbell (1988), task load is seen as a relatively stable attribute of the task, independent of the performer or the external factors. As mentioned, task load is the main determinant for workload; however, the relationship between task load and workload is moderated by personal and external factors.

Although Campbell (1988) and de Zwart et al. (1995) address the elements that influence mental and physical task load in depth, practical difficulties arise when using their definitions as a starting point to determine mental and physical task load in the AWS simulation of emergency response. Although successful in estimating task load in emergency response on a subset of tasks (see Griefahn, Künemund, & Bröde, 2003, Sothmann, Saupe, Raven, Pawelczik, Davis, Dotson, & Landy, 1991, Siliunas, 1991, Lemon, & Hermiston, 1977, Levin, France, Hemphill, Jones, Chen, Rickard, Makowski, & Aronsky, 2006), the most serious disadvantage, interestingly, resides in their thoroughness, focussing on a subset of emergency responders or emergency response activities, making it virtually impossible to collect grounded data to extrapolate the methods that are used to all tasks of all emergency responders.

A more general, but also valid, specification of task load and workload is provided by the NASA task load index (NASA-TLX, Hart & Staveland, 1988). The NASA-TLX is a widely used instrument to assess the task load and the workload of the performer. The NASA-TLX is a multidimensional instrument that provides an indication of the general workload of a performer, while differentiating between the relative contributions of six workload sources. In Table 6.1 the rating scale definitions and end-points of the NASA-TLX are shown.

From the NASA-TLX, all scales, or elements in the scales, are used in the AWS workload model, except “performance”. Performance is not included as one of the potentially influencing factors of workload because it is a highly subjective measure applied *after* task performance that interacts with the other factors. The subjectivity resides in the fact that it is not based on actual performance, but on perceived performance. In this sense, a person can perform badly, but can indicate that he or she performed well. While the other factors directly influence workload, subjective performance does not. Secondly, while the other factors can be determined in advance (which is essential in the processing capacity view), performance can only be determined in hindsight, making it not applicable within the processing capacity view on performance. Thirdly, perceived performance interacts with other factors such as biased perception of performance due to temporal demands or high task load.

The NASA-TLX addresses task load by incorporating scales about the mental and physical demands posed by the task on the performer. Mental demands of a task within the NASA-TLX address the severity of a range of cognitive activities, such as thinking or deciding, and perceptual activities such as looking and searching. Physical demands of a task relate to severity of the range of physical activities within a task, such as pushing and pulling.

Table 6.1: *NASA-TLX rating scale definitions and endpoints.*

Rating Scale Definitions		
Title	Endpoints	Descriptions
Mental demand	Low / High	How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?
Physical demand	Low / High	How much physical activity was required (e.g. pushing pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
Temporal demand	Low / High	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
Performance	Good / Poor	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?
Effort	Low / High	How hard did you have to work (mentally and physically) to accomplish your level of performance?
Frustration level	Low / High	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Mental and physical activities within a task are mutually exclusive, but complement each other in providing a general indication of task load. Tasks, however, can differ to what degree they possess mental and physical activities and how demanding these activities are. Table 6.2 illustrates this by listing task types, ranging from a task that consists only of mental activities, to a task that consists only of physical activities. The mental task load of these tasks has a constant value of 5 (simple task), while the physical task load has a constant value of 20 (hard task). These values correspond with the endpoints used in the NASA-TLX, ranging from 0 (low load / simple task) to 20 (high load / hard task). A simple task thus can contain a small proportion of hard physical activities.

Table 6.2: *Task type, proportion and load of mental and physical tasks and task load*

Task type	% Mental	Mental load	% Physical	Physical load	Task load
Mental task	100	5	0	20	5 ((1x5)+(0x20)
80/20	80	5	20	20	8 ((.8x5)+(.2x20))
60/40	60	5	40	20	11 ((.6x5)+(.4x20))
40/60	40	5	60	20	14 ((.4x5)+(.6x20))
20/80	20	5	80	20	17 ((.2x5)+(.8x20))
Physical task	0	5	100	20	20 ((0x5)+(1x20))

Task load is determined by the weighted sum of the mental and physical load of the activities within a task. The NASA-TLX provides a valid, accepted, and, most important of all, practical method that can be used to assess task load within the AWS.

6.2.2 Task Environment: External Moderating Factors

A general moderator of the relationship between task load and workload is provided by the NASA-TLX. Besides giving an indication for task load, the NASA-TLX takes notice of an external, task environment related load source: time pressure. As Staal (2004) states "Time pressure limits the time available to perform a given task. This limit is a physical boundary that does not require any psychological explanation in understanding its direct effects on performance. However, this limitation often evokes a corresponding psychological reaction such as anxiety that has secondary or indirect effects on performance." One of the psychological effects of time pressure is that it increases the workload given a task, since it compresses the distribution of the demands over time (Luczak, 1971). Workload is higher when performing the same task under high versus low time pressure conditions and moderates the relationship between task load and workload.

The NASA-TLX addresses the workload increase due to time pressure by referring to the working pace, working rate and the added frustration. The increase of the working pace (less time to execute a single task) reduces the time span, compared with the time span normally needed for task execution. Compressing the task duration increases the workload. The increase of the working rate (less time between tasks), on the other hand, decreases the opportunity to recover between task executions and restore the processing capacity. The Subjective Workload Assessment Technique (SWAT), developed by Reid & Nygren (1988), complements the elements described in the NASA-TLX by mentioning the decreased ability to recover within a task due to the lack of spare time. Recovery during task executions is also hampered. Finally, the added frustration due to time pressure within the NASA-TLX refers to the subjective increased state of arousal which is caused by time pressure, leading to increased workload. Taken together, time pressure increases the workload by condensing the task and heightening arousal and, furthermore, decreases the opportunity to recover between and during task execution.

While time pressure is a general moderator of the relationship between task load and workload, being applicable to all tasks, task specific moderators can also be distinguished in the task conditions. Heat (Guidotti, 1992), noise (Becker, Warm, Dember, & Hancock, 1995) or vibrations (Newell & Mansfield, 2007) or task location, for example, increase workload for the performer. The task of extinguishing a fire in this sense differs in workload when it is performed outside or inside a building (change of location, increased heat and noise of the fire). In this dissertation these factors are not considered as external moderating determinants of workload, but are seen as variations of a task with its own task load, since these elements have such a major influence on the task itself, that they transform the nature of the task, making it a distinct task. While time pressure is a universal moderating factor, which applies to all tasks, working conditions are task specific. This specificity is in contrast with the aim of the AWS to construct a general model to approximate workload. As a consequence, in this dissertation the influence of the task conditions is not considered as a moderating factor. Their influence however is included in the task typology.

Finally, in his review on stress, cognition and human performance, Staal (2004) pays attention to other external task environmental workload moderating elements such as:

- the presence of others
- social facilitation
- pro social behaviour
- group member status

- personality
- emotional awareness

The relationship between these elements and workload, however, is poorly understood (Staal, 2004) and including them in the model would require too much guesswork. In order to provide a general estimation of workload resulting from external moderating elements that moderate the relationship between task load and workload, time pressure and task conditions therefore are adequate.

It is concluded that time pressure, in combination with the task conditions (which are embedded in the tasks themselves), are the main moderating elements that influence the relation between task load and workload. The AWS workload model, however, will only incorporate time pressure separately as a moderator, for reasons expressed earlier.

6.2.3 Task Performer: Person Moderating Factors

With regard to the person specific mediating factors for workload, experience or elements that can be directly related to experience (repetition, skill acquisition) are often seen as a major source of diversity for the subjective experience of task load (Hilburn & Jorna, 2001, Loft, Sanderson, Neal, & Mooij, 2007). Shiffrin & Schneider (1977) showed that experience with performance of a task results in automatic or subconscious processing of the task, and, in turn, results in a decrease of workload and time needed for task performance. Within the framework of human performance proposed by Rasmussen (1983), the relation between automaticity and workload is further addressed (see Figure 6.3). Depending on the level of repetition, training or experience of the performer, a task can elicit skill based, rule based and knowledge based behaviour. The behavioural response demanding the least effort is the preferred and elicited response. This framework furthermore differentiates between signals, signs and symbols regarding information processing at the different levels of automaticity. These will come into the picture when the communication load model is described in Chapter 7. Rasmussen's framework has been used to explain workload differences in, for example, adaptive operator support (Grootjen, Neerincx, & van Weert, 2006), workload driver experience in different traffic environments (Patten, Kircher, Östlund, Nilsson, & Svenson, 2006) or system architecture (Storms, 2004).

Skill based behaviours are highly automatic and "take place without conscious control as smooth automated, and highly integrated patterns of behaviour" (Rasmussen, 1983) and are acquired by repetition, training and experience. Skill based behaviour requires little effort for the performer and is similar to automatic processing mentioned by Shiffrin & Schneider (1977). Skill based behaviour can be considered as relying on tacit procedural knowledge.

In situations where no automated response is available, one can rely on rule based behaviour or knowledge based behaviour. These types of responses require more effort than skill based behaviour. Rule based behaviour uses stored response rules that are deduced from familiar task situations or rules that are communicated from other persons' know-how. Rule based behaviour can be considered as relying on available explicit procedural knowledge.

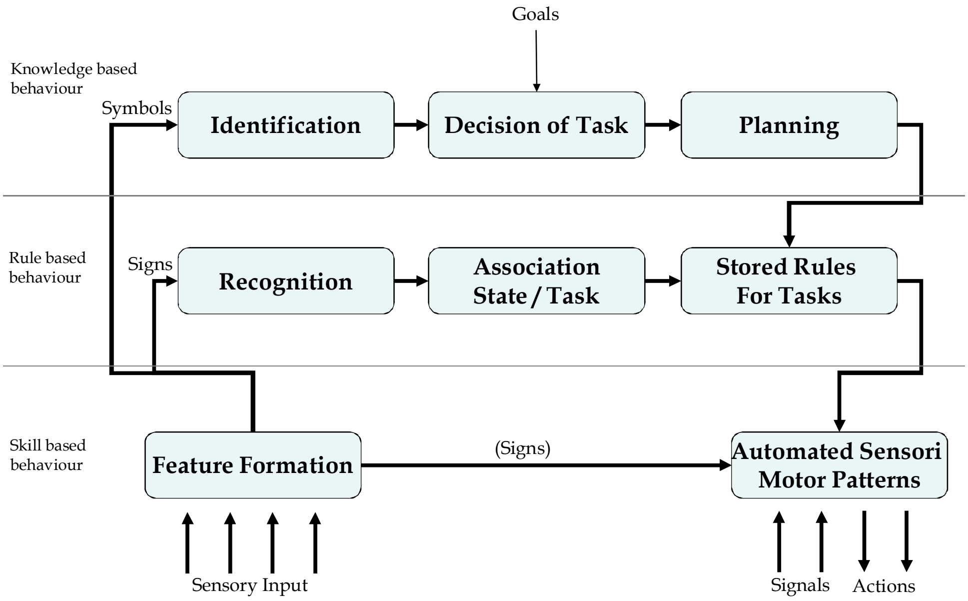


Figure 6.4: *Rasmussen's (1983) framework of human performance*

When no rules are available or can be derived from familiar task situations, rules are constructed from scratch, using knowledge based behaviour. This requires more effort than skill or rule based behaviour. Rules constructed as a result of knowledge based behaviour are stored, and can be used later in familiar situations. In all situations, skill based response has the highest preference, followed by the rule based response that, in turn, is followed by the knowledge based response, since the first requires the least effort for the performer and the last the most.

Differences between experienced and less experienced performers in workload are caused by the number of skill and rule based behavioural responses that are available to the performer. Familiarity and procedural knowledge that result from experience will result in workload decrease. When experienced performers, however, are faced with non familiar, novel tasks, they also will experience higher workload for that task, since it will elicit a knowledge based response (Patten et al., 2006). When comparing inexperienced and experienced performers under time pressured conditions, interaction effects are reported. Experienced performers' performance degrades less than the performance of less experienced performers under time pressure conditions. Reasons for these differences are sought in speed of pattern recognition (Calderwood, Klein, & Crandall, 1988) and differences in procedural knowledge of the performer (Spilker, 1994). Experience thus decreases the workload and enables the performer to cope with time pressure in a more efficient manner, decreasing the impact of the task on the processing capacity of the performer. Furthermore, due to effective coping responses that are learned from previous encounters to adequately deal with the impact of high workload and stressful situations (such as time pressure), recovery speed is also positively affected by experience, creating a double edged sword of the effect of experience on the tasks' impact on the processing capacity of the performer.

It can be concluded, that the personal attributes representing the performer's level of automaticity of task execution and experience with the task, moderates the relation between the task load and workload. An increase of experience results in an increase of task automaticity, which results in a decrease of the experienced workload. The moderating relation of experience, in turn, is moderated by time pressure. Time pressure amplifies the workload differences between experienced and less experienced performers and thus amplifies the damping effect of experience on the relationship between task load and workload. Furthermore; it can be assumed that experience passively influences recovery speed, due to more efficient coping strategies. Thus experience is a major source of diversity in workload. Therefore, experience will be included in the workload model as the person related moderator of the relationship between task load and workload.

6.2.4 Conclusion

Based on the theoretical elements that make up workload that are presented in section 6.2, a general model is constructed (Figure 6.5) that is used in the AWS to approximate the workload and the processing capacity of emergency responders included in the simulation. The model used in the AWS, is a translation of the general model of workload presented in Figure 6.3, using concrete, quantifiable predictors. The AWS workload model is characterized by:

- its differentiation between task load as an attribute of the task and workload as an attribute of the performer;
- the inequality of workload and task load ;
- the moderated relationship between task load and workload by experience and time pressure;
- the interaction of time pressure with experience, increasing the differences in workload between experienced and less experienced performers;
- differences in recovery speed between experienced and less experienced task performers;
- the impact of historic workloads on the available processing capacity;
- the ability to recover during and between tasks to "refill" the processing capacity.

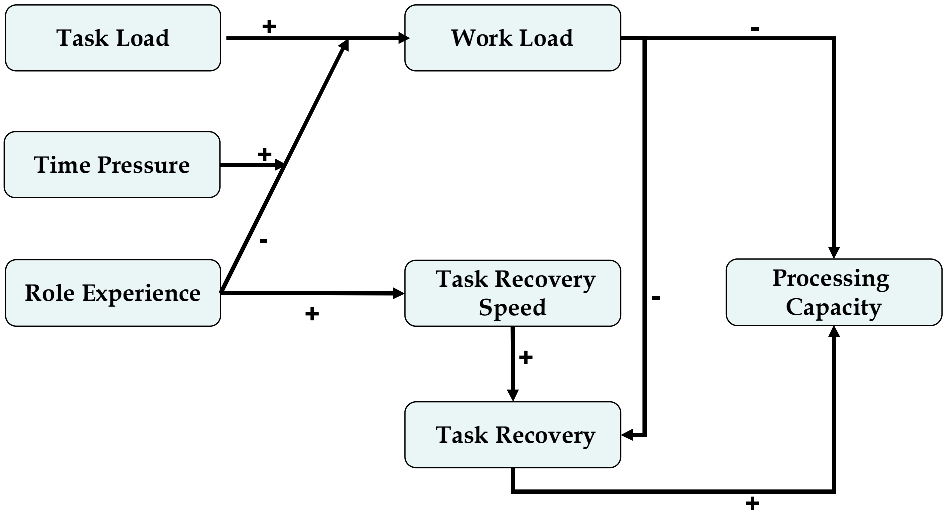


Figure 6.5: *AWS Workload model.*

Workload is determined by the interplay between the physical and mental resources that are demanded for task execution and the resources that can be allocated by the task performer. The workload of the performer is determined by the task itself and varies based on the performer's experience and the temporal demands and task conditions during task execution. Workload negatively affects the level of available processing capacity for task execution, which affects overall task performance. Furthermore, due to recovery during and between tasks, the processing capacity is able to "refill" itself. Section 6.3 will present the detailed workload model and its implementation within the AWS.

6.3 AWS Workload Model

In the model presented in Figure 6.5, workload is determined by task load. The relationship between task load and workload is negatively moderated by role experience. An increasing level of role experience hereby reduces the workload of the performer. Time pressure, furthermore, amplifies the damping effect of experience on the relationship between task load and workload. Role experience has a positive influence on the speed at which one is able to recover. Given the same task, experienced task executers thus have a lower workload, are able to recover more quickly between and within tasks and are less susceptible to the influence of time pressure than less experienced performers, completing the double edged sword of experience. An increase of workload, however, can exclude the ability to recover during that task, which is not influenced by experience. Experienced and less experienced emergency responders can only recover during relatively simple tasks. It should however be noted that the relative simplicity of a task (workload) does differ due to the level of experience. Finally, workload and recovery between or within tasks determine the processing capacity. Workload decreases the processing capacity and recovery "fills" the processing capacity again.

The devil however is in the details. When operationalizing such models and relationships into concrete values and relationship types, many decisions have to be made. This sub section will describe the implementation and the route taken to translate the theoretical causal model displayed in Figure 6.5, into a plausible model grounded in emergency response practice that is implemented in the AWS. Grounding of the model is achieved by incorporating the answers of experts in the field of emergency response on the questionnaire that addresses several elements in the model. A more detailed description of the questionnaire and the sample can be found in Chapter 2. In this section, first the operationalization of task load, workload and the moderating elements of this relationship between task load and workload is presented (section 6.3.1). Secondly, the relationship between experience, workload and task recovery is operationalized in section 6.3.2 addressing within and between task recoveries. Finally, section 6.3.3 will describe the operationalization of the impact of task execution and recovery on the processing capacity.

6.3.1 Task load, Workload, Experience and Time Pressure

As indicated at the start of section 6.2.1, task load can be seen as a relatively stable attribute of the task, independent of the performer, where workload refers to the actual impact of task load on the performers' processing capacity (workload). The relationship between task load and workload is moderated by experience and time pressure. In this section the operationalization of these elements within the AWS is presented. It should however be noted that the aim of the present section is not to define the task load and workload of all task types specified in Chapter 4 (describing the different tasks in emergency response). The aim is to provide the AWS with the necessary features for the modeller to specify the stable attributes of the task and let the AWS on the fly calculate the workload of the emergency responders during the simulation. An application of the model will be presented in a proof of concept simulation in Chapter 8.

6.3.1.1 Task load

Based on the scales in the NASA-TLX that address task load, it was concluded in section 6.2.1 that task load is determined by the weighted sum of the mental and physical load of the activities within a task. To specify a value for the physical and mental task load of a task, the AWS uses the same scales and endpoints as the ones specified in the NASA-TLX. The values range from 0 (low load / simple task) to 20 (high load / hard task). The weighted sum is determined by multiplying the physical and mental task load of a task with the percentage of the task they cover.

$$TL = (mTL \times pmT) + (pTL \times ppT)$$

Task load (TL) is the combined load of all the mental elements in a task (mTL) times the percentage in which the task consists of mental elements (pmT) plus the combined load of all the physical elements in a task (pTL) times the percentage in which the task consists of physical elements (ppT). The mental and physical elements here refer to the sub activities that are listed in the description of the "mental demand" and the "physical demand" scales that are used in the NASA-TLX. Furthermore, since mental and physical tasks are assumed to be mutually exclusive and describe all elements in the task, the sum of the ppT and the pmT always is 100 percent.

For example, the task of "extracting a hard to reach victim" is characterized by both a fairly high mental task load (for example, planning, thinking) and physical task load (for example pulling, pushing). The first has a value of 14 (`mentalTaskLoad`) while the latter has a value of 18 (`physicalTaskLoad`). Once a plan is formed about how to extract the victim, the remaining part of the task largely concerns the physical labour of actually extracting the victim. The task therefore is characterized by physical aspects dominating the task (`percentagePhysicalTask` = 70%) over the mental aspects (`percentageMentalTask` = 30%). Figure 6.6 illustrates the initial beliefs about this task.

```
initial_beliefs:

   (current.mentalTaskLoad("Extracting hard to reach victim") = 14);
   (current.physicalTaskLoad("Extracting hard to reach victim") = 18);
   (current.percentageMentalTask("Extracting hard to reach victim") = 0.30);
   (current.percentagePhysicalTask("Extracting hard to reach victim") = 0.70);
```

Figure 6.6: *Task load parameters in the AWS*

```
thoughtframe calculateWeighedTaskDemands01 {
   display: "calculating weighed task demands";
   priority: 1010;
      variables:
         forone (double) PTL; //Physical Task Load
         forone (double) MTL; //Mental Task Load
         forone (double) PPT; //Percentage Physical Task
         forone (double) PMT; //Percentage Mental Task
         forone (double) WPTD; //Weighed Physical Task Demands
         forone (double) WMTD; //Weighed Mental Task Demands
         forone (double) WTD; //Weighed Task Demands
      when (knownval(PTL = current.physicalTaskLoad(activity))and
            knownval(MTL = current.mentalTaskLoad(activity)) and
            knownval(PPT = current.percentagePhysicalTask(activity)) and
            knownval(PMT = current.percentageMentalTask(activity)) and
            knownval(WPTD = PTL * PPT) and //calculate weighed physical task demands
            knownval(WMTD = MTL * PMT) and //calculate weighed mental task demands
            knownval(WTD = WPTD + WMTD)) //calcualate weighed Task demands
      do {
         conclude((current.weighedTaskDemands(activity) = WTD), fc:0, bc:100);
         conclude((current.weighedPhysicalTaskDemands(activity) = WPTD), fc:0, bc:100);
         conclude((current.weighedMentalTaskDemands(activity) = WMTD), fc:0, bc:100);
      } // close do
}//close thoughtframe
```

Figure 6.7: *Brahms code calculating the weighted task demands (task load).*

When the task of "Extracting hard to reach victim" is performed for the first time, the task load value is calculated for the performer. For the task of "Extracting a hard to reach victim" this results in a task load of (14 x 0.30) + (18 x 0.70) = 16.8. Figure 6.7 shows the Brahms "thoughtframe" needed to complete this operation.

In the thoughtframe, first a check is made to determine if the task parameters that are needed to calculate the task load are present. When one of these parameters is not present, the task load, and consequently the workload, will not be calculated. In order for the workload and the processing capacity to be approximated during the simulation, it thus is critical that these parameters are present. When the parameters are present, the calculations needed to determine the task load value are performed which result in a `weightedTaskDemands` value, representing the task load of the current activity. Two additional values, the `weightedPhysicalTaskDemands` and the `weightedMentalTaskDemands`, are stored for use in later calculations. These values represent the weighted physical (wpTL, 18 x 0.70 = 12.6) and the weighted mental task load (wmTL, 14 x 0.30 = 4.2) of the current task. Once calculated, these parameters remain unchanged for the rest of the simulation.

Thus, by using the four values addressed in the NASA-TLX that represent task load, the AWS is able to on the fly calculate the task load of tasks that are performed by the emergency responders. In order to calculate the workload of the emergency responders, the experience of the performer and the time pressure has to be operationalized.

6.3.1.2 Experience

In section 6.2.3, addressing the person dependent moderating factors of the relationship between task load and workload, it was concluded that an increase of training, repetition and experience is likely to result in an increase of task automaticity and a decrease of the experienced workload. As indicated in Chapter 4, in emergency response, sets of tasks are strongly linked to the role the individual responder plays in the response. It therefore is likely that a responder's role experience expressed in years, has a decreasing influence on the subjective impact of the task load.

This view was supported by the group of emergency response experts who filled out the questionnaire described in section 2.3.2. From the sample, 73% agreed and 12% strongly agreed with the statement that given the same task, an increase of role experience would result in a decrease of workload in the domain of emergency response. The average score on the Likert scale (ranging from 1: strongly disagree to 5: strongly agree) corresponding with this statement was 3.96 ($N=26$, $SD=.63$).

However, as shown in Figure 6.8, when differentiating between task compositions, varying the degree in which task are composed of mental and physical aspects, the experts in the field of emergency response indicate that the beneficial effect of more role experience on the workload significantly decreases when tasks are becoming more physical ($F(2.73, 62.75) = 8.80$, $p<.001$). Task composition thus influences the degree to which role experience affects workload.

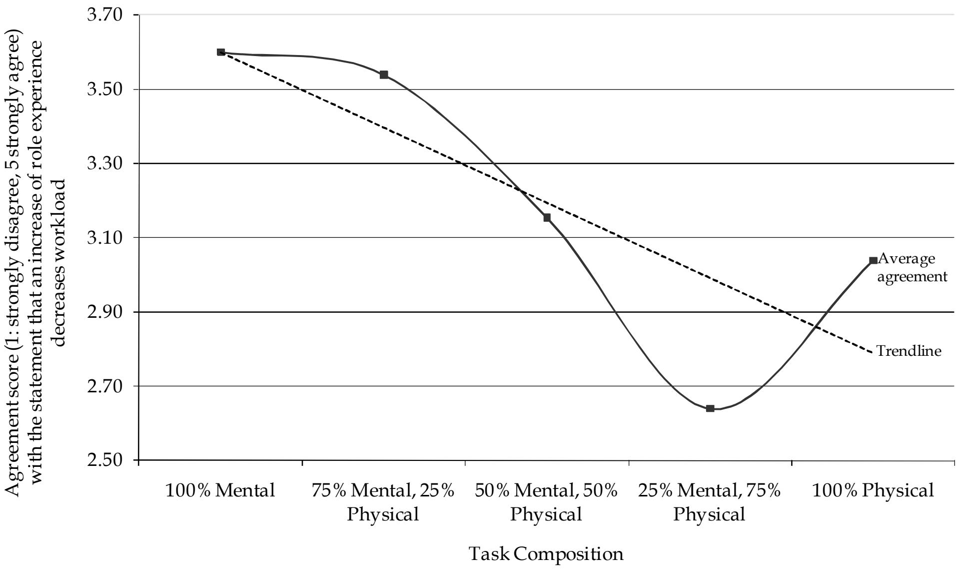


Figure 6.8: *Graphical representation of the relationship between task composition and the positive effect of role experience on workload.*

The differing beneficial effect of role experience in mental and physical tasks on workload, furthermore, is reflected in the experts' answers when asked to explicitly indicate which of the nine graph types in Figure 6.9 best represented the relationship between role experience on the horizontal axis and workload on the vertical axis.

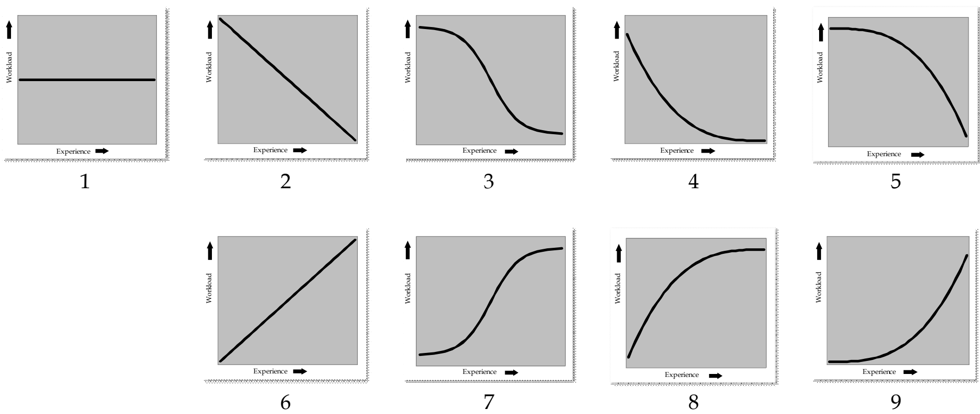


Figure 6.9: *Nine relationship types responses used in the expert questionnaire to indicate the relationship between role experience (horizontal axis) and workload.*

Two curves account for 80% of the responses. More than half of the respondents (56%) indicated that curve 3 best represents the relationship between role experience and workload for mental tasks. Curve 3 describes a situation where, at the start of a career, workload decreases slowly. During the period that follows, workload quickly decreases, and then stabilizes. In the final years, an increase of role experience only leads to a small decrease in workload. The second major group (24% of the respondents) indicated that curve 1 best represents the relationship between role experience and workload for mental tasks, implying that there is no effect of an increase of years of role experience on the mental workload of the responder. The mental workload for an experienced responder is the same as the mental workload of a non experienced responder, given the same task. Regarding physical tasks, 48% of the respondents are of the opinion that the relationship between years of role experience and workload is best described by curve number 1 (horizontal line). The second largest group of respondents (12%) are of the opinion that the relationship is best described by curve number 3. The remaining 40% is evenly distributed over the remaining 7 categories.

It can be concluded that the relationship between role experience in years and workload differs for mental and physical aspects of tasks. The AWS takes this into account by using curve number 3 to describe the moderating effect for mental task elements on the relationship between task load and workload, and the horizontal line (curve number 1) to describe the moderation effect of the relationship between task load and workload for the physical elements within a task. Not taking the final moderating element (time pressure) into account, workload can be described by the following formula.

$$\text{WLe} = (\text{wmTL} \times \text{exfmTL}) + \text{wpTL}$$

Workload (WLe) corrected only for the influence of experience thus equals the weighted mental task load (wmTL) times the experience curve of mental task load (exfmTL) plus the weighted physical task load value (wpTL). The experience curve represents curve number 3 in Figure 6.9. The formula used within the AWS to calculate the experience curve of mental task load (exfmTL) as a function of role experience in years, is displayed below.

$$\text{exfmTL} = 0.611\left(\frac{1}{1+e^{\frac{yre}{2}-4}}\right) + 0.4 \quad (yre \geq 0)$$

For each positive whole number of years of role experience (*yre*), the experience factor is calculated. Figure 6.10 displays the resulting curve when entering role experiences ranging from 0 to 20 years. The corresponding experience factor ranges from 1 for emergency responders with zero years of role experience, to 0.40 for an emergency responder with twenty years of experience, following the z-shaped curve. An emergency responder with twenty years of role experience thus is loaded by only 40% of the mental load connected to the tasks that make up the responder’s role.

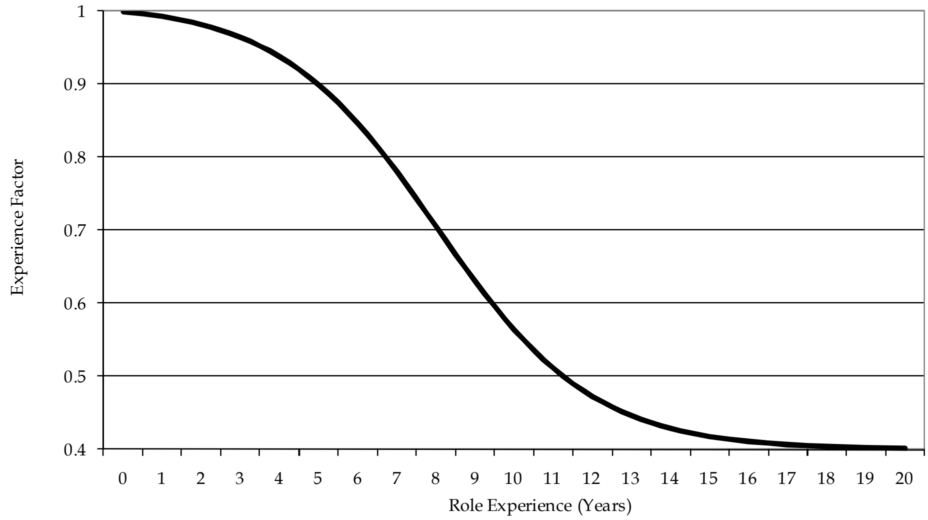


Figure 6.10: *Experience factor as a function of role experience in years.*

When returning to the example of the task of “extracting a hard to reach victim” that was presented in section 6.3.1.1, it can be seen that the weighted mental task load (wmTL) of this task was set at 4.2, and the weighted physical task load (wpTL) was set at 12.6. Resulting from these values, the task load for this task was set at 16.8. When the number of years of role experience of the agent performing the task is known, the workload corrected for years of role experience (WLe) can be calculated.

Table 6.3 shows the values resulting for WLe for five different emergency responders, who only vary in their number of years of role experience.

Table 6.3: *Workload corrected for the number of years of role experience for the task of "extracting a hard to reach victim" calculated for five different role experience values.*

Role Experience (years)	exfmTL	wmTL	wmTLe	wpTL	WLe
1	0.99	4.20	4.17	12.60	16.77
5	0.90	4.20	3.78	12.60	16.38
10	0.56	4.20	2.37	12.60	14.97
15	0.42	4.20	1.76	12.60	14.36
20	0.40	4.20	1.69	12.60	14.29

Using the years of role experience value, the experience factor for mental task load (exfmTL) is calculated. Since it was found that the number of years of role experience only influences the mental elements of the task, no experience factor has to be calculated for the load of physical elements in a task. As can be seen in Table 6.3, an emergency responder with 1 year of role experience is loaded with 99% of the weighted mental task load of the task, while an emergency responder with 10 years of role experience only is loaded with 56% of the weighted mental task load. This results in a weighted mental task load corrected for years of role experience (wmTLe) of 4.17 for an emergency responder with 1 year of role experience, and a wmTLe of 2.37 for an emergency responder with 10 year of role experience. As a result, the workload corrected for the years of role experience are 16.77 for the first and 14.97 for the latter emergency responder.

An increase of the years of role experience of the performer when executing tasks that contain mental aspects thus lead to an increase of the difference between the task load and the workload corrected for years of role experience. Furthermore, in addition to an increase of years of role experience, the difference between task load and workload also will increase due to task composition and the mental task load of the task. When a task for the larger part consists of physical aspects, the workload decreasing effect of role experience expressed in years will decline. When the task does not contain mental elements, the number of years of role experience does not reduce the workload for more experienced task executers. However, when the mental task load of a task increases, so will the difference between task load and workload corrected for role experience. An increase of experience expressed in the years of role experience thus overall will lead to a varying decrease of workload corrected for years of role experience, depending on the task composition and the mental task load of a task.

The number of years of role experience is not the only moderator of the relationship between task load and workload, since time pressure further amplifies the differences between the stable task attribute of task load and the actual impact on the processing capacity of an individual agent in the form of the workload. It achieves this by amplifying the damping effect of the number of years of role experience on the relationship between task load and workload.

6.3.1.3 Time Pressure

As was concluded in section 6.2.2, time pressure is a general moderator of the relationship between task load and workload, applicable to all tasks, but has a different effect on experienced and less experienced task executers. An increase of time pressure (less time to execute a single task), reduces the time available for task execution thus increasing the workload for both physical and mental tasks. In the AWS workload model, this is reflected by the fact that time pressure amplifies the damping effect of experience on the relationship between task load and workload.

Addressing the magnitude of the damping effect, the experts in the field of emergency response were asked to indicate which proportion of workload is determined by time pressure during emergency response activities. The respondents indicate that time pressure accounts for 26% (N= 23, SD=11.6) of the total workload in emergency response situations. Furthermore, as Table 6.4 clearly shows, the proportion of workload determined by time pressure is similar for all organisational levels of the emergency response organisation.

Table 6.4: Proportion of workload determined by time pressure at the different organisational levels in the emergency response organisation (N =23).

	Emergency Response Level				
	Operational	Tactical	Strategical	Governmental	Overall
Proportion of workload determined by time pressure	25.65%	24.29%	26.84%	27.86%	25.96%

Thus, when time pressure is present, this should account for an average of 26% added workload at all levels of the emergency response organisation. However, in line with Rasmussen's theory, more experienced task executers are less affected by the increase of workload due to time pressure than less experienced emergency responders.

In the AWS, the moderating influence of time pressure and the difference in added workload of time pressure due to differences in years of role experience, is addressed by defining a time pressure factor (tpf) that amplifies the workload corrected for years of role experience (WLe) by proportionally amplifying the WLe. More experienced emergency responders are less impacted by time pressure than less experienced responders, which is in line with the theory reported in section 6.2.3 concerning the difference between the influence of time pressure on experienced and less experienced task performers. Workload corrected for time pressure and years of role experience in the AWS is represented by the following formula.

$$WLtpe = tpf\ (WLe)$$

Workload corrected for the presence of time pressure and years of role experience (WLtpe) thus equals the workload corrected for the number of years of role experience (WLe) times the time pressure factor (tpf). In the AWS, the WLtpe value is the operationalization of workload that is presented in the model in Figure 6.5.

Using the proportion of workload indicated by the experts that is determined by time pressure (Table 6.4), the tpf value in the formula above is operationalized by selecting a random value between 1.144 (14.4%) and 1.376 (37.6%), which is derived from the average proportion of workload determined by time pressure (26%) plus and minus one standard deviation (11.6%). Using an upper and lower bound, instead of a fixed value, to express the tpf is preferred, given the relatively large deviation in the reported values for the proportion of workload determined by time pressure by the experts. As these deviations exist between the experts, it can be assumed that they also deviate in practice from person to person. The tpf for situations where no time pressure is present, equals 1.

In the example task of "extracting a hard to reach victim" the workload corrected for the years of role experience was determined (WLe) for agents with different experience values (1-5-10-15-20 years of role experience). When these agents are placed in a time pressured situation, the time pressure factor (tpf) is determined by the AWS and workload corrected for years of role experience and time pressure (WLtpe) can be calculated. Table 6.5 illustrates this for the tpf value of 1.26 and the WLe that was determined in section 6.3.1.2.

Table 6.5: *Workload corrected for time pressure the number of years of role experience for the task of "extracting a hard to reach victim" calculated for five different role experience values.*

Role Experience (years)	tpf	WLe	WLtpe	WLtpe – Wle
1	1.26	16.77	21.13	4.36
5	1.26	16.38	20.64	4.26
10	1.26	14.97	18.86	3.89
15	1.26	14.36	18.09	3.73
20	1.26	14.29	18.00	3.71

In Table 6.5 it can be seen that an increase of the number of years of role experience decreases the workload corrected for the number of years of role experience. Due to the presence of time pressure, the workload corrected for time pressure and the number of years role experience increases. The increase of the WLtpe compared with the WLe (WLtpe – Wle), which is caused by the presence of time pressure, however, differs for emergency responders with a higher number of years of role experience when compared with their less experienced colleagues. More experienced emergency responders thus experience less load increase due to the presence of time pressure than less experienced co-workers. The absolute influence of time pressure on the WLtpe furthermore decreases when tasks are relatively simple (due to a smaller WLe) compared with harder tasks or when the exf value is set at a lower random value. Time pressure has a different absolute workload increasing effect on experienced responders when compared with less experienced emergency responders. The WLtpe thus describes the impact of task execution (workload) on the processing capacity of the individual performer, taking the number of years of role experience of the performer and the presence of time pressure into account.

The sequence in which the WLtpe is calculated in the AWS is as follows. First, at the start of the simulation, the experience factor is calculated using an external java activity in Brahms resulting in an individual experience factor (exfmTL) for all agents. When a task is activated (`current.currentTask(Activity) = true`), the weighted mental task load (wmTLe) and weighted physical task load (wpTL) of that task is calculated for the individual performer. Subsequently, the time pressure factor is determined (tpf) for the performer. Finally, the workload corrected for time pressure and number of years of role experience (WLtpe) is calculated.

To be able to calculate the WLtpe for a task the agent must have beliefs about:

- the agent's number of years of role experience
- the presence of time pressure
- physical task load of the task to be performed
- mental task load of the task to be performed
- proportion of mental task elements in the task
- proportion of physical task elements in the task

The AWS can thus on the fly approximate the workload of an emergency responder for a particular task in advance when these 6 variables are specified. Since these values all are relatively stable between emergency responses and responders, one can easily adapt the model to an emergency at hand. This makes it a generic, reusable model to on the fly approximate the workload of tasks during emergency response. The aim of the AWS workload model, however, is to describe the decrease of available resources for task execution (processing capacity) due to workload and the increase of the processing capacity due to recovery during and in between tasks. The following section (section 6.3.2) will present the implementation of the between and within task recovery in the AWS.

6.3.2 Task Recovery

In the AWS, task recovery is addressed with two “recovery types”, both refilling the processing capacity. The first recovery type refers to recovery between tasks, which can be seen as refilling the processing capacity by taking a break from work, recharging the battery (in this case the processing capacity). However, when performing relatively simple tasks, one also is able to refill some of the processing capacity spent, recharging the battery by executing tasks with a low demand. This second recovery type is referred to as during task recovery. The threshold that determines if a task is low in complexity, enabling during task recovery, is determined by sub dividing the NASA-TLX scale into five zones of task complexity, mapping them on Likert scale values. The mapping of the NASA-TLX score range on Likert scale values is shown in Table 6.6.

Table 6.6: *mapping of the NASA-TLX range on Likert scale values.*

TLX Score Range	Likert Scale Mapping
0 to 4	Very simple
4 to 8	Simple
8 to 12	neither simple nor hard
12 to 16	Hard
16 to 20	Very hard

From Table 6.6 it can be derived that relative simple tasks range from a NASA-TLX score of 0 up to a score of 8. The latter therefore is taken as the threshold for the possibility of during task recovery. During task recovery thus only is possible when the WLtpe is smaller than 8.

In the AWS workload model, the degree to which one recovers is a function of the workload and the task recovery speed. The workload determines which recovery type is initiated, and determines if recovery is initiated. When no tasks are performed, and, consequently, the workload at that moment is zero, between task recovery is initiated by the performer. When the performer is executing a task, during task recovery is initiated if the task being performed is relatively simple. If this task exceeds the threshold of a WLtpe of 8, no recovery during task execution will occur. Recovery during task execution thus depends on the workload of the task that is currently executed.

Recovery speed refers to the speed at which the processing capacity is refilled, expressed in processing capacity gain per second. The AWS workload model uses two constant recovery speeds. One applies to between task recovery situations and one applies to during task recovery situations. The rationale behind the two constant recovery speeds is that it is most likely that the recovery speed when performing a task is slower than the recovery speed when not performing a task. Within the AWS, task recovery speed for between task recovery is set at a constant processing capacity gain of 0.0333 (1/30) per second, while the during task recovery speed is set at a constant processing capacity gain of 0.0167 (1/60) per second. The relationship between these values and the processing capacity will be presented in section 6.3.3. A performer thus can fully recuperate from a task with a WLtpe of 15 by taking a break of 7.5 minutes or by performing 15 minutes of relatively simple work.

As can be seen in the AWS workload model shown in Figure 6.5, recovery speed furthermore is influenced by the years of role experience of the performer. This reflects that, due to increased role experience, an experienced task executer has better coping strategies than a less experienced performer. In the AWS, recovery speed adjusted for the number of years of role experience (RSe) equals the experience factor (expf) times the standard recovery speed (RS) for the recovery type activated. This relationship is expressed in the formula stated below.

$$RSe = expf \times RS$$

The experience factor reflects the increase of the recovery speed due to the number of years of role experience of the performer.

In this section, the implementation in the AWS of the two recovery types is presented. Section 6.3.2.2 described the between task recovery model, while section 6.3.2.3 will deal with the during task recovery model. However, before the implementation of these two types of recovery in the AWS is presented, first the relationship between recovery speed and the number of years of role experience is investigated for emergency response situations (section 6.3.2.1).

6.3.2.1 Role Experience and Recovery and Recovery Speed

In order to determine the influence of the number of years of role experience on the recovery speed, the expert questionnaire included two statements concerning the relationship between the number of years of role experience and recovery speed. The experts were asked to indicate to what extent they agreed or disagreed on a five point Likert scale with the statement that recovery speed from mental tasks increases due to an increase of the number of years of role experience, and with the statement that recovery speed from physical tasks increases due to an increase of the number of years of role experience. The asnwers on these statements are shown in Figure 6.11.

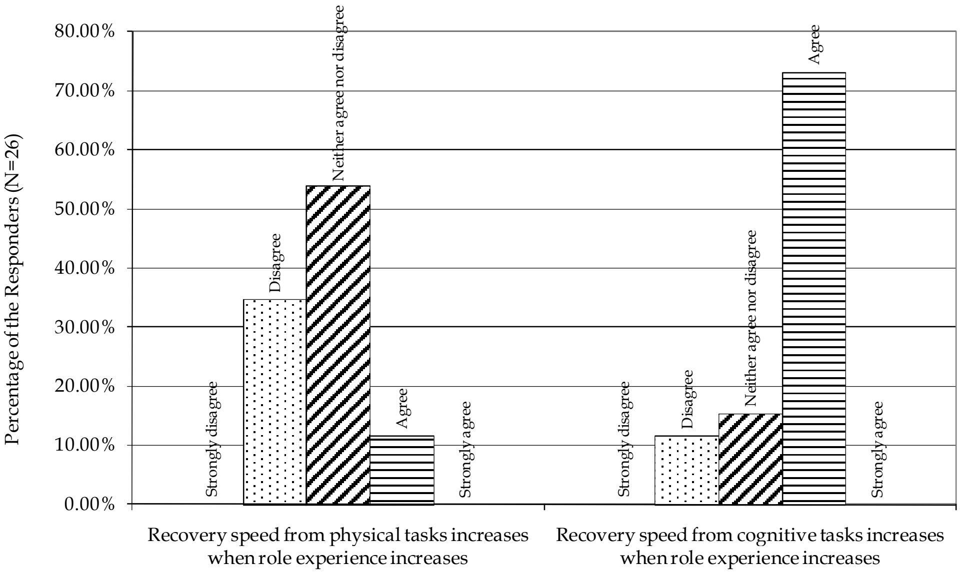


Figure 6.11: *Responses to the statement that recovery speed from mental and physical tasks increases due to the number of years of role experience of the emergency responder.*

The average score on the Likert scale, ranging from strongly disagree (Likert score of 1) to strongly agree (Likert score of 5), for the statement that recovery speed from physical tasks increases when the number of years of role experience increases is 2.77 (N=26, SD=0.65) and does not significantly differ from the neutral score of three ($t(25)=-1.81$, $p>.05$), but differs significantly from "disagree" ($t(25)=6.01$, $p<.001$). From these values it can be concluded that role experience, according to the respondents, does not influence the recovery speed from physical tasks.

The average score on the Likert scale for the statement that recovery speed for mental tasks increases when the number of years of role experience increases, is 3.62 (N=26, SD=0.70), differing significantly from the neutral score of three ($t(25)=4.50$, $p<.05$) and the "agree score" of four ($t(25)=-2.81$, $p<.01$). For mental tasks it therefore can be concluded that, although the experts state that there is a positive trend indicating that the number of years of role experience influences the recovery speed, the experts in the field of emergency response do not agree with this statement.

The implications of these findings for the AWS workload model are that the number of years of role experience do not increase the recovery speed from tasks. The recovery speed between and during tasks thus is not affected by the number of years of role experience of the performer. Implementing the experts' opinions in the AWS workload model results in the adjusted workload model illustrated in Figure 6.12. Instead of using the recovery speed corrected for the number of years of role experience (RSe), the AWS uses the constant recovery speeds which are linked to the recovery type.

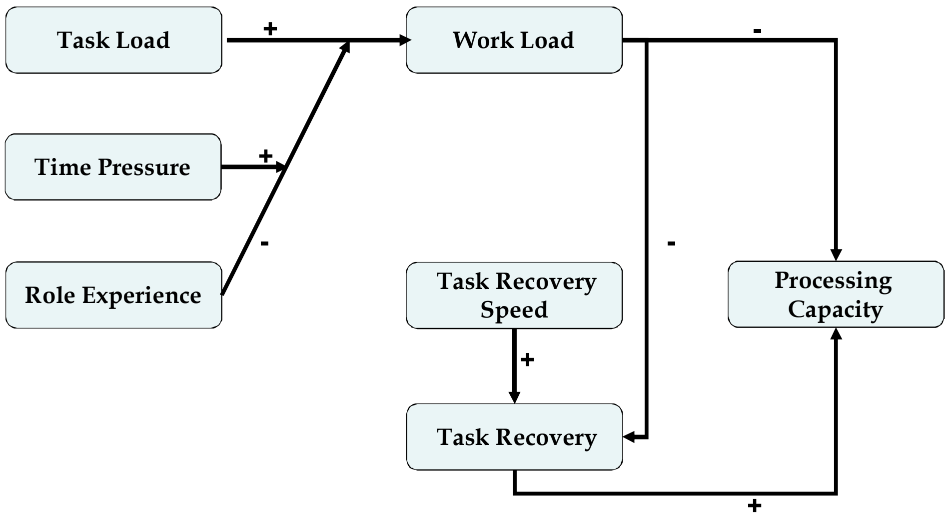


Figure 6.12: *Adjusted workload model*

Although it can be concluded that there is no direct effect of the number of years of role experience on task recovery speed, the time needed to fully recover from a task is affected by the number of years of role experience, since the number of years of role experience influences the WLtpe. Table 6.7 illustrates this effect, using the task of "extracting a hard to reach victim" as an example.

Table 6.7: *Time needed to recover from "extracting a hard to reach victim" when recovery occurs between or during tasks for five different role experience values.*

Role Experience (years)	WLtpe	Between task recovery time (BTRT)	During task recovery time (DTRT)	DTRT - BKRT
1	21.13	00:10:34	00:21:08	00:10:34
5	20.64	00:10:19	00:20:38	00:10:19
10	18.86	00:09:26	00:18:52	00:09:26
15	18.09	00:09:03	00:18:05	00:09:03
20	18.00	00:09:00	00:18:00	00:09:00

As can be seen in Table 6.7, and was demonstrated in detail in Table 6.5, the WLtpe value decreases with an increase of the number of years of role experience. As a consequence, the time needed to recover from the task of "extracting a hard to reach victim", given a constant recovery speed, also decreases. An increase in the number of years of role experience thus negatively influences the needed time to fully recover from task execution, and reduces the absolute difference between the DTRT and the BTRT. In Table 6.7, it can be seen that an emergency responder with one year of role experience has to take a break of 10 minutes (BTRT) or perform relatively simple activities for about 21 minutes (DTRT) to fully recover from the task of "extracting a hard to reach victim". An experienced emergency responder only has to take a break for 9 minutes (BTRT) or perform relatively simple task for 18 minutes (DTRT) to fully recover.

Since the decrease of the WLtpe between emergency responders with different amounts of years of role experience is determined by the mental elements of the task (the exfmTL), the trend indicated by the experts that years of role experience do have a positive influence on the recovery speed from mental elements of a task, thus implicitly is reflected in the model by the time needed to fully recover from a task. Furthermore, since the number of years of role experience moderates the relationship between task load and workload, the proportion of relatively simple tasks is larger for emergency responders with a larger number of years of role experience. From the 231 possible task compositions (when comparing whole wmTL and wpTL values), and not taking time pressure into account, a performer with 1 year of role experience encounters 44 (19%) compositions that can be labelled as simple tasks (WLe < 8), where a performer with 10 years of role experience encounters 68 (29%) simple tasks, and a performer with 20 years of role experience encounters 91 (39%) simple tasks. Since the performance of relative simple tasks initiates during task recovery, more experienced emergency responders are able to recover more frequently during their work than less experienced emergency responders performing the same task.

It can be concluded that although there is no direct relationship between the number of years of role experience and recovery speed, the recovery time needed to fully recover from a task is affected by the number of years of role experience through the decrease of the WLtpe values and the difference in proportion of simple tasks an experienced and a less experienced performer encounter.

6.3.2.1 Between Task Recovery

This section will describe the implementation of between task recovery. Between task recovery resembles a period of doing nothing or taking a break. During these periods, the workload is zero and no current tasks are active. Figure 6.13 shows the Brahms code needed to implement between task recovery.

```
workframe wf_recoverBetweenTasks {
   display: "recovering between tasks";
   repeat: true;
   priority: 1;
      variables:
         forone(string) activity; // Task
         forone(double) PC; // Processing Capacity
         forone(double) BRS; // Between task Recovery Speed
         forone(double) newPC; // Updated Processing Capacity
      when (/*not(current.currentTask(activity)=true)and*/
            knownval(PC = current.processingCapacity) and
            knownval(PC < 960.0) and
            knownval(BRS = 1/30) and
            knownval(newPC = PC + BRS))
      do {
         recoverBetweenTasks();
         conclude((current.processingCapacity = newPC), bc:100, fc:0);
      }// close do
   } //close workframe
```

Figure 6.13: *Brahms code for recovery between tasks*

The rationale behind the Brahms code presented above is, that when there is no current task and one has a belief about the current processing capacity value, and this value does not exceed the maximum value it may have (960), the new processing capacity can be calculated by adding to the recovery speed value (`BRS`) the current processing capacity, leading to a updated processing capacity (`newPC`). After these actions are performed, the activity of `recoverBetweentasks` is initiated with a duration of 1 second and the `newPC` is used as the new processing capacity. This activity is repeated up until the moment when either the processing capacity is fully restored or the emergency responder starts performing a task, overruling the priority of the task of recovering between tasks.

Between task recovery thus can be represented in the AWS, equipping emergency responders with the ability to recover between tasks. The next section will address the implementation of during task recovery in the AWS.

6.3.2.2 During Task Recovery

During task recovery is recovery during the execution of tasks and can only occur when one is performing relatively simple tasks. In the AWS the threshold that is used to distinguish between "simple" tasks (where within task recovery is possible) and "harder" tasks (where within task recovery is not possible) is set at a WLtpe value of 8. Figure 6.14 shows the Brahms code needed to implement during task recovery in the AWS.

```
thoughtframe RecoverDuringTask {
   display: "calculating recovery during task";
   priority: 1003;
      variables:
         forone(double) WLtpe; //WLtpe
         forone(double) R; //Total Recovery During task
         forone(double) dur; //Task Duration
         forone(double) RS; //Recovery Speed
      when (knownval(WLtpe = current.correctedTaskDemands(activity)) and
            knownval(WLtpe < 8.0)and
            knownval(dur = current.taskDuration(activity))and
            knownval(RS = 1/60) and
            knownval(R = RS * dur))
      do {
         conclude((current.taskRecovery(activity)= R), fc:0, bc:100);
      } //close do
} // close thoughtframe
```

Figure 6.14: *Brahms code for recovery during an activity*

In the Brahms code presented in Figure 6.14, it can be seen that the degree of task recovery during a task is calculated when the WLtpe value is known and is smaller than the threshold value of 8. Furthermore, in order to determine how much one can recover during a task, the duration of the task must be known. As a result, the approximated amount of recovery during a task (`R`) is calculated by multiplying the recovery speed (`RS`), expressed in gain of the processing capacity per second, by the duration of the task (`dur`) in seconds.

Since between task recovery is independent of a task that is being performed, its impact on the processing capacity can be calculated immediately and is updated each second. However, during task recovery is linked to a task and the duration of that task. In the AWS, the WLtpe and the possible recovery during an activity and its effect on the processing capacity is calculated simultaneously at the start of each activity. The way in which the WLtpe and during task recovery is implemented in the AWS is presented in the following section (section 6.3.3)

6.3.3 Processing Capacity

The processing capacity refers to the amount of resources that can be allocated to task execution by the performer. The available processing capacity present (APC_{t0}) is a result of the available processing capacity before task executions (APC_{start}) minus the sum of the workloads of the executed tasks up until that moment, plus the amount of between (BTR_{t-n}) and during task recovery (DTR_{t-n}) up until that moment. This results in the following formula, describing theavailable processing capacity present.

$$APC_{t0} = APC_{start} - (WLtpe_{t-n}) + (BTR_{t-n}) + (DTR_{t-n})$$

In the previous sections of this chapter the composition and the implementation of the WLtpe, the BTR and the DTR in the AWS were presented. Section 6.3.2.1 furthermore gave a sneak preview of the APC_{start} value when describing the preconditions for between task recovery.

In the AWS, the APC_{start} value represents the initial capacity of the reservoir of processing capacity that is available to the performer for task execution. The speed at which the reservoir is depleted fluctuates due to characteristics of the task (task load), the task environment (time pressure) and personal characteristics (number of years of role experience). Since these individual differences of emergency responders and situations in which these emergency responders operate already are incorporated in the WLtpe, the APC_{start} can be set at a fixed starting value equal for all emergency responders.

In the AWS, the value of the APC_{start} is set at 960. Using this value, the performer is able to fully refill its processing capacity after 8 hours (using between task recovery). When the performer is working normal shifts (even in emergency response situations) per day, no workload spill-over into the next day will be present. Performers with a depleted processing capacity that are not able to fully recover, however, will experience this spill-over effect of workload, resulting in a decreased APC_{start} at the start of the next working day. It should be noted that the processing capacity can be depleted in less than 8 hours.

When the depletion level (processing capacity = 0) of available processing capacity is passed, the performer needs to refill the processing capacity to be able to perform any new tasks. However, as was stated in section 6.1, immediate execution of concurrent tasks will bring the performance capacity in the red again, resulting in needed recovery afterwards. High work rates with minimal recovery will thus result in an accumulation of needed recovery times and prolonged overall duration of tasks. In the AWS, this is reflected by implementing a precondition into the agent workflow planner (Figure 5.3) that states that a new task can only be initiated if the processing capacity has a value higher than zero. When this precondition is not met, no new task will be initiated and between task recovery will be activated. When the processing capacity value exceeds zero, new tasks can be initiated by the workflow planner. Figure 6.15 presents a part of the results from a simulation run that illustrates the effect of exhausting the processing capacity on overall work duration.

In the simulation, whose results are shown in Figure 6.15, four tasks are modelled that are performed repeatedly. The order in which the tasks are performed is *Task1-Task2-Task3-Task4*. The impacts of these tasks (WLtpe + DTR) on the processing capacity differs; Task1 being the hardest task and Task4 being the easiest task. For this simulation, the duration of each task and between task recovery is set at 30 seconds for illustrative purposes. As can be seen in Figure 6.15, the available processing capacity declines as a result of executing the tasks mentioned. However, when *Task1* is performed for the second time, the lower limit of the processing capacity is passed. Before the concurrent task, *Task2*, can be executed, the performer first has to refill its processing capacity. After 30 seconds of recovery, *Task2* can be executed. This, however, results in the processing capacity running in the red again. As a consequence, before the performer can commence with executing *Task3,* first the processing capacity has to be positive again. This process is repeated up until the point that the performer has no tasks left to execute.

As can be seen in Figure 6.15, the magnitude by which the lower limit of the processing capacity is passed due to task performance, determines the time needed to refill the processing capacity. For example, the execution of *Task2* requires the performer to recover 5 minutes (10 recovery "tasks" of 30 seconds each), while the execution of *Tasks3* requires the performer to recover 3 minutes and 30 seconds.

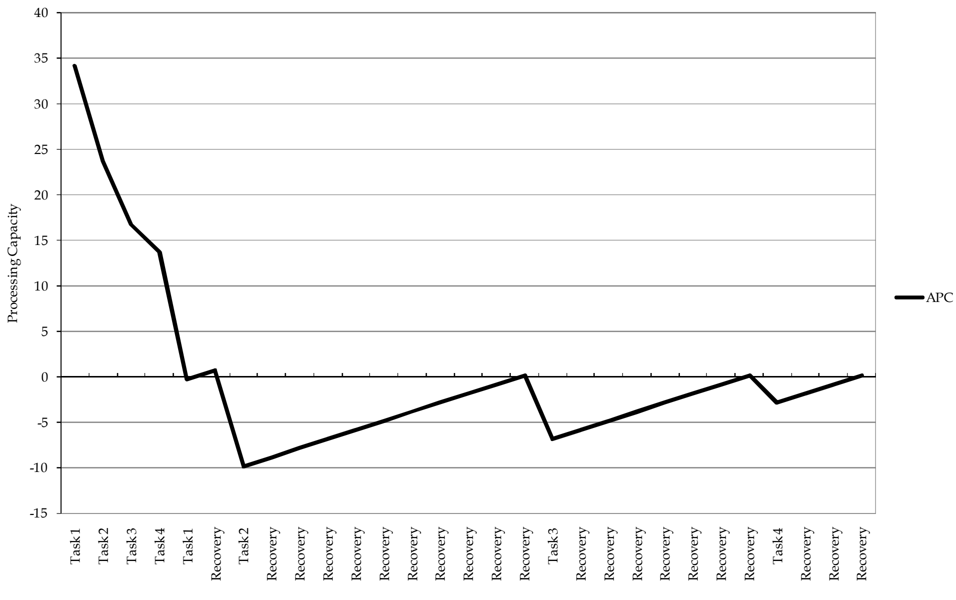


Figure 6.15: *Visual representation of the effect of overstepping the processing capacity on overall work duration*

The first cycle in which all four tasks are performed takes the performer 2 minutes to complete. However, due to the added time of needed recovery, the second cycle in which all four tasks are executed for the second time takes 12 minutes and 30 seconds. The overall duration of the cycle thus increases significantly, due to a depletion of the processing capacity.

As mentioned above, to realize this behaviour by the AWS, first a precondition has to be incorporated into the AWS workflow planner stating that new tasks can only be executed when the present processing capacity is above zero. Secondly, with each execution of a task, the new processing capacity value has to be calculated.

```
thoughtframe calculateAvailableProcessingCapacity {
   display: "calculating available processing capacity";
   priority: 1002;
      variables:
         forone(double)WLtpe;//WLtpe value
         forone(double)DTR;//During Task Recovery
         forone(double)T1;//Temp variable
         forone(double)APCc;//Current Available Processing Capacity
         forone(double)APCn;//New Available Processing Capacity
      when (knownval(WLtpe = current.correctedTaskDemands(activity)) and
            knownval(DTR = current.taskRecovery(activity)) and
            knownval(APCc = current.processingCapacity)and
            knownval(T1 = WLtpe - DTR) and
            knownval(APCn = APCc - T1))
      do {
         conclude((current.processingCapacity = APCn), fc:0, bc:100);
         } //close do
}//close thoughtframe
```

Figure 6.16: *Brahms code needed to calculate the processing capacity*

The Brahms code needed to calculate the processing capacity is shown in Figure 6.16. In the code it is specified that in order to determine the available processing capacity after the task is executed, the WLtpe value, the during task recovery (DTR) value of the task and the present available processing capacity should be known for the performer that is executing the task. When these values are known, the processing capacity after task performance is calculated by subtracting the WLtpe value from the present processing capacity and adding the DTR value to the present processing capacity.

6.4 Conclusion

This chapter answered the research questions posed at the start of the chapter, being:

- How can we build a generic grounded model of the workload, processing capacity and the influence of workload and processing capacity on the workflow that lets us on the fly approximate the workload and processing capacity of individual emergency responders in the AWS?
  1. Which theoretical concepts and relationships underlie workload and processing capacity?

2. How do these concepts and relationships behave in emergency response practice?
3. How can we implement these concepts and relationships in the AWS in a generic fashion?
4. How does the workload model influence the workflow of the individual emergency responder?

Addressing the first sub question, the relationship between workload and the processing capacity was explored resulting in the conclusion that workload mainly addresses the load associated with the execution of a single task, where the processing capacity addresses the load associated with the execution of a series of tasks. Using the analogy of a processing capacity reservoir proposed by Kahneman (1973), it was concluded that the present available processing capacity is the result of the starting level of the reservoir minus the cumulative historic workloads posed on the task executer plus the cumulative amount of processing capacity due to recovery between and during the execution of tasks. The reservoir thus is depleted by the workload associated with executing tasks and is refilled by recovering between and during tasks. The individual responder's workflow is influenced by using the depletion level as a precondition for task execution. Tasks can only be executed when the processing capacity is not below the depletion level. This results in a time delay in the workflow due to the added duration of recovery between the tasks (answering sub question 4).

The main predictor of workload is task load. Task load refers to the stable attributes of the task which define its mental and physical complexity. When no moderating factors of the relationship between task load and workload would be present, it could be concluded that task load = workload. It however was argued that the relationship between task load and workload is moderated by several factors residing in the task environment and the task performer. The main environmental moderator is time pressure and the main performer related moderator is experience.

In the resulting workload model (Figure 6.5), firstly, workload is determined by the task load. Secondly, the relationship between task load and workload is negatively moderated by the number of years of role experience of the emergency responder. Thirdly, time pressure amplifies the damping effect of experience on the relationship between task load and workload. And, finally, the number of years of role experience has a positive influence on the task recovery speed.

This workload model was grounded in emergency response by administering a questionnaire addressing the relationships in the model to experts in the field of emergency response. By doing so, sub question two was answered. The three main conclusions based on the experts' opinions, expressed in the questionnaire, were that firstly, the number of years of role experience only negatively moderated the mental elements of a task; Secondly, the number of years of role experience had no direct influence on the recovery speed, and thirdly, when present, time pressure accounts for 26 percent of the additional workload. These findings resulted in an adjustment of the AWS workload model, which is presented in Figure 6.12. According to the experts in the field of emergency response, the factors of workload, recovery and consequently the processing capacity, do not fully support the initial model. Besides grounding the theoretical model, the experts in the field of emergency also provided information about the type of relations between the elements in the model, which are used to quantify and calculate the processing capacity, workload and recovery.

Using the AWS workload model, it is possible to on the fly approximate the workload, processing capacity and the recovery of individual emergency responders by only specifying 6 variables, which all can be defined in advance. Four variables are linked to the task in order to determine task load; one variable is a value representing the number of years of role experience of the emergency responder and one value provides an indication of the presence of time pressure. Since the workframes and thoughtframes responsible for the workload and processing capacity calculations, the starting processing capacity, the presence of time pressure and the task attributes are defined at a high level in the inheritance tree, they are automatically available to all emergency responders. By modifying these values, they automatically are modified for all emergency responders. They, however, can be overruled by the children groups or individual agents, adjusting the values locally, bringing flexibility to the simulation. A generic model emerges which can be modelled easily at an agent level, representing the individual or groups of emergency responders.

To summarize, using the workload model implemented in the AWS, we have build a generic and grounded model of the workload, processing capacity and the influence of workload and processing capacity on the workflow, which lets us on the fly approximate the workload and processing capacity of individual emergency responders in the AWS. In this way the global behaviour of emergency responders, as far as the execution of tasks is concerned, is taken into account in the simulation, increasing the functional fidelity of the simulation.

7. Communication Load

The adhocratic organization (such as the emergency response organization) allocates a relatively large portion of time and effort to communication to coordinate activities in order to keep track of the current state of the emergency as well as the current state of the emergency response organization. Illustrative for the vast amount of time and effort spent on communication and coordination is the amount of information exchanged during the COPI exercise observation described in section 2.3.3. The main responsibility of the COPI, that stands for commando incident location, is to coordinate the multidisciplinary response at the incident location, therefore most, if not all, communication serves the purpose of coordination and increasing situational and organizational awareness. During the COPI exercise, a total of 1837 utterances (or messages) and 4709 information elements were recorded. For an exercise that lasted for 133 minutes, this amounts to an average of one utterance being exchanged every 4 seconds and an average of one information element being exchanged every 2 seconds (with peaks of 2 information elements per second). As a consequence of the amount of information that is exchanged during an emergency response, a large load is put on the emergency responder. The respondents to the expert questionnaire even indicated that information exchange is the number one contributor to the experienced load during emergency response, accounting for 26% (N=23, SD=6.84) of the total experienced load.

Addressing the strain put on the emergency responder as a consequence of communication (the process of handling information), the communication load model described in this chapter will enable the AWS to monitor the communication load and its consequences on the workflow of the individual emergency responder included in the simulation. Since both sending and receiving information refer to activities performed by respectively the sender and the receiver of the information, the communication load model is a specific version of the workload model presented in Chapter 6. Also, the load associated with communication has a strong link with task load and workload (Manning, Mills, Fox, Pfeiderer, & Mogilka, 2002). So the model developed in Chapter 6 will be applied to build the model of communication load. Departing from the grounded workload model presented in Figure 6.12, the communication load model that can be derived from it is shown in Figure 7.1.

The communication load model describes the process where the message load is the main predictor for communication load. Message load refers to a stable attribute of a message which defines the load of the message, independent of the person handling it and so refers to the load of the message using characteristics of the message itself. The subjective experience of the message load is expressed by the communication load. Communication load refers to the actual impact of the message load on the sender or receiver’s processing capacity. Indicative for the load associated with handling information is the load of the message that is handled by both the sender and the receiver: hence communication load can be represented by a single communication load model.

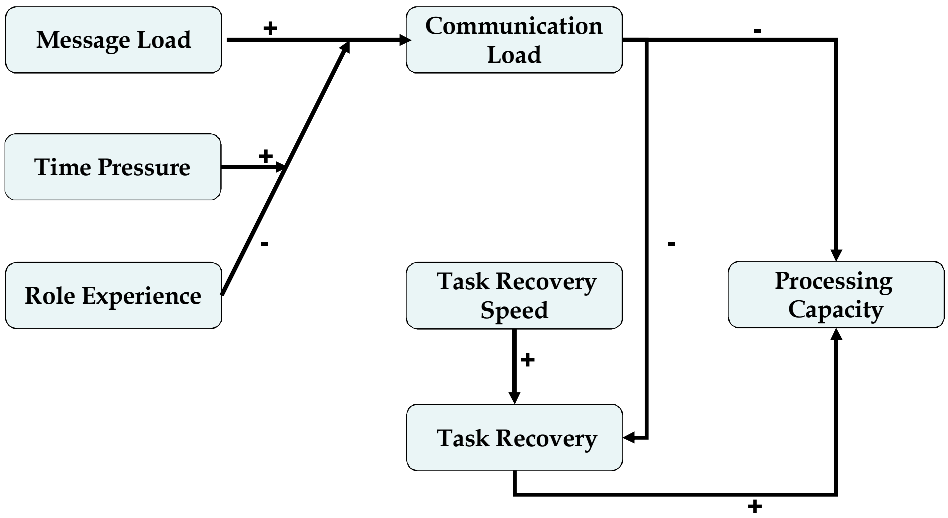


Figure 7.1: *AWS communication load model*

Given the same message, communication load associated with a message, however, can be different for the sender and the receiver. Similar to the workload model, the relationship between message load and communication load is negatively moderated by the number of years of role experience of the emergency responder. Time pressure, furthermore, amplifies the damping effect of experience on the relationship between message load and communication load. An experienced emergency response experiences less communication load than a less experienced emergency responder.

The model also differentiates between differences in load caused by situational differences for the sender and the receiver. For example, a message can be sent under low time pressure conditions, while the receiver is working under high time pressure conditions. In this case, the communication load for the receiver (given the same level of experience) is higher than the communication load for the sender of the message.

Communication load, furthermore, negatively influences the ability to recover during the task and depletes the available processing capacity of the emergency responder. Task recovery (recovery from the task of handling information) can "refill" the available processing capacity. By depleting the available processing capacity for the execution of tasks, communication load and recovery determined by handling information influence the workflow, similar to "normal tasks" that are specified in the workload model.

Since message handling takes time, communication also directly influences the workflow of the emergency responders, apart from its indirect influence on the workflow via the depletion of the processing capacity caused by handling the message. Distinct from the tasks planned by the task planner that estimates a single agent's workflow, communicating involves one sender and one or more receivers who simultaneously are involved in the task of handling the same message. Communication thus both simultaneously and directly influences the workflow of more than one agent incorporated in the simulation model.

The following sections will present, implement and ground the communication load model, focussing on those aspects in which the assessment of the load associated with the activity handling information deviates from a typical task. Section 7.1 describes the term message load. Next, the moderating role of experience on the relationship between message load and communication load and the amplification of the damping effect of experience on the relationship between message load and communication load will be presented in section 7.2. Section 7.3 will address how handling information directly and indirectly influences the workflow in the AWS. Finally section 7.4 presents the concluding remarks.

7.1 Message Load

The first step to approximate the load associated with handling messages, entails determining the load of the message, independent of the person who is handling it, or the conditions under which it is performed. Since the physical load associated with handling a message is most likely to be negligible, message load is determined by the mental load associated with handling a message.

Within the AWS, message load is specified on the NASA-TLX task load scale. Based on this scale, the message load ranges from 0 to 20, where a message with a load of 0 refers to a message that is easy to handle and 20 refers to a message that is very hard to handle. To determine the message load associated with a message, the unit of measurement that is used is the information element. Similar to the “act” that is used as the unit of analysis in the Bales group communication analysis (Bales, 1970), an information element contains a subject (topic of which is spoken) and a predicate (what is said about the topic) and is sufficiently complete in order for the receiver to interpret the information element and act on it (see also section 2.3.1.1).

Indicative for the amount of message load associated with a message is the number of information elements that are included in the message. Since a single message can contain one, or more than one information element, each added information element is likely to result in an increase of the message load, making it less easy to handle. Within the AWS, the relation between message load and the number of information elements in the message is represented using the following formula: .

$$\text{ML} = 13/10 \text{ x } e^{\left(\frac{InformationElements}{6}\right)}$$

For each positive whole number of information elements (*InformationElements*) the message load (ML) is calculated. The calculated message load ranges from 1.5 when the message contains one information element and is reaches its maximum value when the message load equals a 20 (the maximum TLX score). By dividing the number of information elements by 6, and multiplying the resulting value with 13/10 the exponential curve shown in Figure 7.2 emerges.

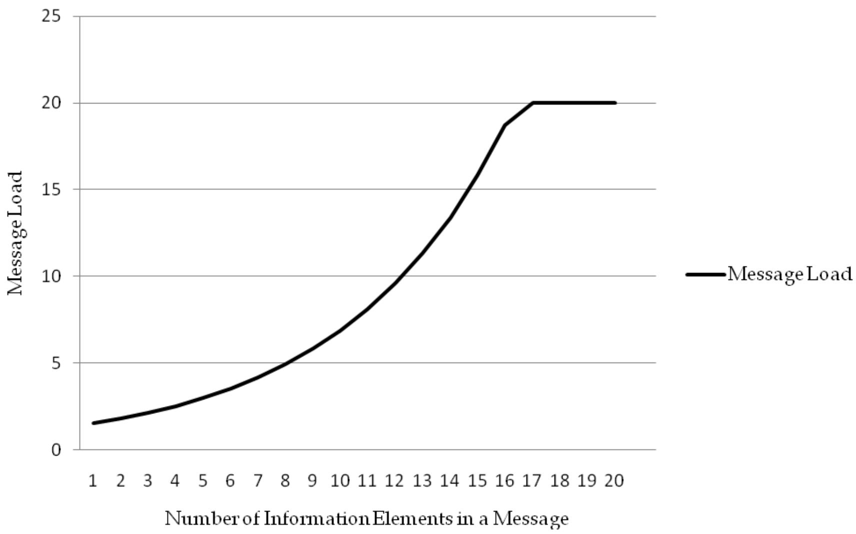


Figure 7.2: *Message load as a function of the number of information elements in a Message*

With every information element added to the message, the message load increases with an increasingly large amount. Thus, the added message load of an extra information element is less for a message containing a small number of information elements when compared with the added message load of an extra information element for a message containing a larger number of information elements. When the message already is very complex (message load of 20), a newly added message does not lead to an increased message load since it is already at its maximum.

In the AWS, the message load values associated with messages that contain up to 20 information elements are predefined using a map attribute. Whenever a communication activity is started in the AWS model, the corresponding message load of the message being communicated can be retrieved.

An example of the message load associated with the flow of information to and from the officer on duty of the fire department in the COPI exercise described in Chapter 2, is shown in Figure 7.3. During the emergency response exercise, the officer on duty handles a total of 822 messages that contain a total of 2521 information elements resulting in a total cumulative message load of 2070. The cumulative message load is the sum of all message loads associated with all messages handled by the officer on duty during the 133 minute exercise. Figure 7.3 shows the cumulative message load, broken down into five minute intervals, showing the increase and decrease of the cumulative message load over time (using five minute intervals). The average cumulative message load in these intervals is 99. Thus, by using information elements as a unit of analysis to approximate the message load of a message, we are able to provide an indication of the cumulative message load over time associated with the messages that are handled by the emergency responders involved in the emergency response exercise and consequently in the AWS.

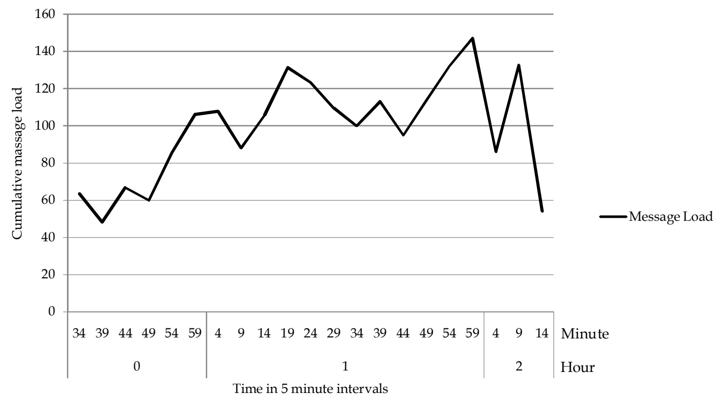


Figure 7.3: *Cumulative message load of for the officer on duty of the fire department during the COPI emergency response exercise.*

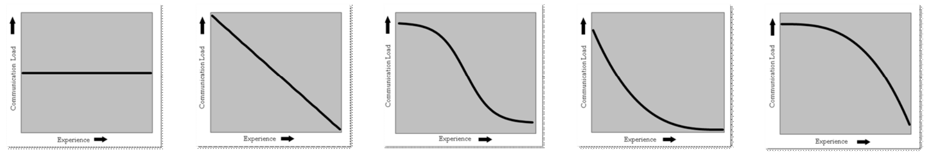
However, to determine the actual impact of these messages (the communication load) that influences the processing capacity and the workflow, experience and time pressure have to be added to the equation, since the relationship between message load and communication load is negatively moderated by the number of years of role experience of the emergency responder while time pressure amplifies the damping effect of experience on the relationship between message load and communication load. Section 7.2 will present how these two factors are operationalized in the AWS for it to be able to monitor communication load.

7.2 Experience, Time Pressure and Communication Load

In addition to that an increase of training, repetition and expertise is likely to result in a higher level of task automaticity which, in turn, leads to a decrease of the load associated with task execution (section 6.3.1.2), Rasmussen (1983) argues that experience alters the perception of information. Given the same information, a person that executes a task at the highest level of automaticity (is engaged in skill based behaviour) is able to handle information more efficiently when compared with persons that execute the same task using rule or knowledge based behaviour. The experienced performer can better extract relevant information. Supporting this view, Lloyd and Somerville (2006) indicate that fire fighters develop a “fire sense” over the years that enables them to better extract relevant information during an emergency response.

The beneficial effect of experience on information handling also is supported by the respondents to the expert questionnaire. Using 5 point Likert scales, together 50% of them agreed and 42% of them strongly agreed with the statement that experience would lead to a better assessment of the relevance of information (M=4.31, SD=0.74, N=26); 50% agreed and 38% strongly agreed with the statement that experienced emergency responders are able to more quickly assess the relevancy of information compared with less experienced emergency responders (M=4.23, SD=0.76, N=26); and finally, the respondents indicated that more experienced emergency responders experience less communication load compared with a less experienced emergency responder, when provided with the same amount of information (M=3.73, SD=0.83, N=26). Two respondents did not agree with the statement that an increase in experience level would results in a decrease of the communication load. These two respondents were routed away from the questions regarding the representation of the decreasing communication load due to experience and so did not have to answer these questions .

To explore how the decrease of communication load due to experience (expressed in the number of years of role experience) can be represented in the AWS, the experts that were of the opinion that an increase of experience decreased or did not influence the communication load (N=24), were asked to indicate which relationship type presented in figure 7.4 best describes this relationship.



(1) (2) (3) (4) (5)

Figure 7.4: *Five relationship types responses used in the expert questionnaire to indicate the relationship between role experience (horizontal axis) and communication load (vertical axis).*

From the 24 experts that made a choice between the five relationship types, the majority (67%) indicated that graph 3 shown in Figure 7.4 best described the relationship between the number of years of role experience and the communication load. In addition, 13% chose graph 5, 8% chose graph 2, 8% chose graph 4, and 4% chose graph 1.

Based on these results, it can be concluded that, in line with Rasmussen (1983) and Lloyd & Somerville (2002), experienced emergency responders are likely to be able to better and quicker assess information relevancy and, furthermore, experience less load when handling the same amount of information compared with their less experienced colleagues. From these theoretical positions, the decrease of communication load due to an increase of experience is the consequence of the experienced performer’s ability to better and quicker assess information relevance. Hence, the communication load value implicitly incorporates these experience related competencies by making it a function of the number of years of role experience. For the assessment of communication load these factors, therefore, do not have to be modelled explicitly in the AWS.

In addition to the fact that an increasing amount of experience decreases the communication load, the experts in the field of emergency response furthermore indicated that the beneficial effect of the number of years of role experience on the communication load can be represented by a z-shaped curve as is presented in Figure 7.4(graph 3).

In the AWS, the moderating effect of experience on the relationship between message load and communication load thus can be represented with the same z-shaped graph as was shown in Figure 6.10, which expressed the relationship between the number of years of role experience and the experience factor that is used to calculate the workload associated with mental activities corrected for the years of experience of the emergency responder. These findings confirm the apparent relationship between the task of handling messages and mental activities, as was indicated in the first paragraph of section 7.1. Not taking time pressure into account, communication load associated with handling messages can be represented by the following formula.

$$CLe = (ML \times exfmTL)$$

Communication load corrected for the influence of experience (CLe) equals the message load associated with handling a message (ML) times the experience factor related to the number of years of role experience of the emergency responder handling the message, as shown in Figure 6.10. Since the amplifying effect of time pressure in general on the damping effect of experience is assumed not to be affected by the task that is performed, the time pressure factor, as was presented in section 6.3.1.3, remains unaffected. Communication load corrected for both time pressure and experience, therefore, can be represented by the following formula.

$$CLtpe = tpf \times CLe$$

Communication load corrected for time pressure and experience (CLtpe) equals the time pressure factor (tpf) times the communication load corrected for the number of years of role experience of the emergency responder handling the message.

Returning to the example that was used to indicate the cumulative message load over time associated with the flow of information to and from the officer on duty from the fire department, using the formula presented above, the communication load, corrected for time pressure and experience, can be calculated. Table 7.1 shows the hypothetical cumulative communication loads (Sum), the average cumulative communication loads in the five minute intervals (M) and the standard deviations of the average cumulative communication loads in the five minute intervals (SD), while varying the presence of time pressure and the number of years of role experience (5 and 15 years) related to the information flow of the officer on duty.

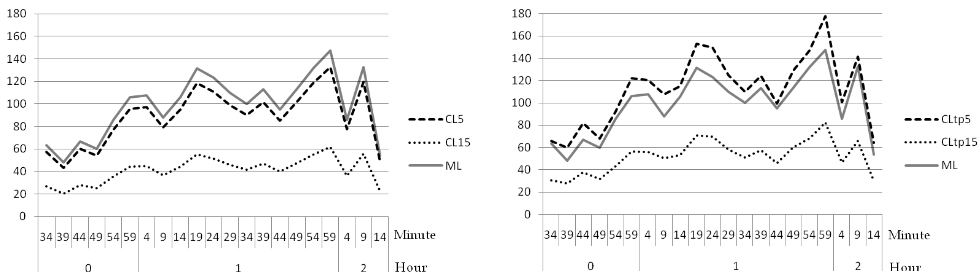
Table 7.1: *Hypothetical communication load for the officer on duty from the fire department in the COPI exercise with varying levels of role experience (5 and 15 years) under time pressured working conditions and under non time pressured working conditions.*

		Experience	
		5 Years	15 Years
Time Pressure	Present	Sum=2347,302	Sum=1090,516
		M=111,776	M=51,929
		SD=31.581	SD=14,672
	Not Present	Sum=1862,333	Sum=865,208
		M=88,683	M=41,200
		SD=25,289	SD=11,749

As is shown in Table 7.1, the *Sum*, *M* and *SD* decrease, due to an increase of experience and increase due the presence of time pressure. The differences in the *Sum*, *M* and *SD* between the two levels of experience are due to the ratio between the experience factors (exfmTL) that are associated with the numbers of years experience. The hypothetical officer on duty with 15 years of role experience has an experience factor of 0,4179, while the other hypothetical officer on duty (with 5 years of role experience), has an experience factor of 0,8995, which is 2,15 times the experience factor of the more experienced officer on duty. Consequently, *Sum*(5 years)=2,15(*Sum*(15 years)); *M*(5 years)=2,15(*M*(15 years)); and *SD*(5 years)=2.15(*SD*(15 years)).

The differences in *Sum*, *M* and *SD* determined by the presence of time pressure are due to the time pressure factor (tpf). As indicated in section 6.3.1.3, this is a random value between 1.144 and 1.376 that, when time pressure is present, is separately determined for each activity that is executed. In this example, due to the presence of time pressure, *Sum*(time pressure present)=1.26(*Sum*(time pressure not present)), *M*(time pressure present)=1.26(*M*(time pressure not present) and *SD*(time pressure present)=1.26(*SD*(time pressure not present)).

Figure 7.5 shows the message load (ML) and the average cumulative communication loads in a five minute interval over time in either time pressured working conditions (b) or non time pressured conditions (a). Since no time pressure is present in figure (a), the only factor influencing the relationship between message load and communication load is the experience factor. This factor is stable throughout an emergency response, therefore the average cumulative communication loads in a five minute intervals for the emergency responder with 5 (CL5) and 15 (CL15) years of experience "follow" the message load value.



(a) ML and CLtpe for 5 (CL5) and 15 (CL15) years of role experience without time pressure (tpf = 1).

(b) ML and CLtpe for 5 (CLtp5) and 15 (CLtp15) years of role experience with time pressure.

Figure 7.5: *Hypothetical communication load over time for the officer on duty from the fire department in the COPI exercise with vary levels of role experience (5 and 15 years) under time pressured working conditions (b) and under non time pressured working conditions (a).*

When messages have to be handled under time pressure, the random time pressure factor amplifies the experience factor. The semi random time pressure factor lets the communication load fluctuate between messages with a similar message load that are handled by emergency responders with the same level of experience, accounting for between and within emergency responder fluctuations. Furthermore, as was also demonstrated in Table 6.5 for workload, time pressure has a different absolute communication load increasing effect for experienced versus less experienced emergency responders; the absolute communication load increases more for less experienced responders handling information in time pressured conditions when compared with more experienced responders. This can be seen in the difference between the absolute increase of the height of the peaks corresponding with the CLtp5 and the CLtp15 lines in figure 7.5(b).

As a final remark, it should be noted that within task recovery, combined with the communication load associated with handling a task, in the end determines the impact of handling information on the processing capacity of the emergency responder. Given that during the COPI exercise the average number of information elements that were included in a message was 3 (with an associated message load of 2.14), handling information mostly concerned handling a large amount of easy to handle messages. As a consequence, within task recovery can take the edge out of the actual impact on the processing capacity of the communication load described above. Table 7.2 shows the hypothetical total within task recovery for the flow of messages to and from the officer on duty, varying the presence of time pressure (present, not present) and the level of experience (5 and 15 years).

Table 7.2: *Total within task recovery during communication activities for the officer on duty from the fire department in the COPI exercise with varying levels of role experience (5 and 15 years) under time pressured working conditions and under non time pressured working conditions.*

		Experience	
		5 Years	15 Years
Time Pressure	Present	61.20	66.37
	Not Present	63.18	69.58

As a consequence of the presence of time pressure, the total within task recovery decreases with 1.98 (from 63.18 to 61.20) for the person with 5 years of role experience and with 3.21 (from 69.58 to 66.37) for the person with 15 years of role experience. The total recovery for the person with 15 years of role experience, however, remains higher in both time pressure conditions, compared with the person with 5 years of experience. Although the presence of time pressure has more influence on within task recovery by the experienced responder through pushing more messages over the boundary beyond which task recovery is started (communication load is smaller than 8; see section 6.3.2), still a larger portion of messages has a smaller communication load for the experienced responder when compared with the less experienced responder. Between task recovery, however, remains to be determined by the time one is not engaged in activities or in handling information, and is not directly influenced by experience nor time pressure. Since receiving information often cannot be planned, the communication load associated with receiving information can both impact the processing capacity directly (due to the load associated with handling information) and indirectly, by disturbing the between task “refilling” process by occupying the receiver with the task of handling the information sent to him/her.

In the AWS, the communication load associated with handling a message is determined by the message load. Accounting for variations in communication load due to an increase of experience and the presence of time pressure, the relationship between message load and communication load is moderated by experience, whose effect, in turn, is damped by the presence of time pressure. Given the same message, more experienced emergency responders experience less communication load compared to less experience responders, and, when having to operate in time pressured conditions, experience a smaller absolute increase of the communication load. During the activity of handling a message, finally, the communication load depletes the processing capacity and within task recovery refills it. The processing capacity in turn influences the workflow.

The following section will present how the agents in the simulation model are equipped with the ability to handle messages (sending and receiving); how this is incorporated in the workflow planner; and how the agents are equipped with the functionality to assess the impact of message handling on the processing capacity.

7.3 Communication Planning

During communicative activities, one sender and one or more than one receiver are involved in handling the same message Communication load associated with handling messages therefore affects multiple emergency responders simultaneously. While the sender is occupied with sending the message, the receiver is occupied with receiving it.

Since communication is initiated by the sender, the activity of sending information is planned by the agent according to the rules defined in the workflow planner (as a normal task). Using the "to do list" analogy, the communication task is added to the list and is triggered based on its priority and the "first come first serve" principle. When the task of sending information has a higher priority than the current task of the emergency responder, the task of sending information can interrupt the current activity.

For the task of receiving information this is more of a rule than an exception. Where the sender can control the flow of messages originating from the sender, the receiver is not able to control the flow of messages directed towards him or her. Receiving information thus always interrupts the current activity of the receiver and cannot be "planned". Receiving communication so bypasses the workflow planner.

In the AWS, this is accounted for by defining an activity that falls in a higher priority range compared to "normal" activities that are performed. In the AWS, three priority ranges are defined. The first range concerns the activities that relate to workflow planning, workload, information load and processing capacity calculations that have to be executed before an activity can commence. These "base" activities are assigned a priority higher than 1000. The second range concerns the activities that are communication activities. While these activities (especially for the receiver of information) overrule the activities that are executed by the receiver, they do not overrule the "base" activities. The communication activities are assigned a priority between 800 and 900. The third range concerns the "normal" activities that are assigned a priority lower than 800. Base activities thus overrule communication activities, and communication activities overrule normal activities.

The "receiving information activity" is triggered whenever the belief `current.currentReceivingMessage(scurrentCommunication)` evaluates to true and the duration of the message is known. Figure 7.6 shows how this "triggering activity" is implemented in the AWS.

The first enabling condition is set by the sender of the message, where `current` refers to the receivers' name (one or more) and the `scurrentCommunication` refers to a unique string variable related to the message. When the `current` value matches the name of the receiver of this belief, first the communication load, within task recovery and the adjusted processing capacity is calculated that is the result of receiving the message (`calculateMessageImpact(scurrentCommunication)`) with a duration of 0. Secondly, an activity is initiated that interrupts the "normal" activities of the receiver(s) with the activity of receiving a message that has a similar duration as the sending duration (`currentMessageDuration(scurrentCommunication)`).

```
// Receiving information
workframe determineInformationImpact {
repeat:true;
  priority: 999;
  variables:
    foreach(string) scurrentCommunication;
    forone(int) dur;
  when (knownval(current.currentListening(scurrentCommunication) = true) and
      knownval(current.currentListenDuration(scurrentCommunication) = dur))
  do  {
      conclude ((current.currentListening(scurrentCommunication) = false), bc:100, fc:0);
      calculateMessageImpact(scurrentCommunication);
      pa_listening(dur);

    }
}
```
Figure 7.6: *Implementation of receiving information in the AWS.*

As a consequence, the receiver is engaged in handling the information that is sent by the sender. The receiver, furthermore, uses his subjective experience factor and situated time pressure factor to calculate the impact of the message on his/her processing capacity.

To summarize, communication from the sender's standpoint is planned in a similar fashion as a "normal" activity. However, to enable the receiver(s) of the information to also be engaged in handling information and have a subjective communication load associated with their level of experience and their current time pressure conditions, the receivers bypass the task planning module for "normal" tasks in the AWS. A separate activity was modelled to firstly calculate the impact of handling a message on the receiver's processing capacity and secondly to occupy the receiver with a "handling message" activity when handling messages.

7.4 Conclusion

This chapter presented how the load associated with handling messages influences the workflow of individual emergency responders during emergency response situations. The communication load model, that is a specific version of the workload model presented in Chapter 6, uses message load as the main determinant of communication load. The relationship between message load is moderated by the numbers of years of role experience of a responder, whose effect on this relationship is damped by time pressure.

Message load refers to the load of a message, independent of the person who is handling it or the conditions under which it is handled. The message load is determined by the number of information elements that the message contains and increases exponentially up to a point with each message added. The maximum message load is 20, relating to the maximum score of the NASA-TLX.

The communication load model was grounded by incorporating the experts' opinions in the development in the model and by testing the model's behaviour on real communication data based on the information handling activities of the officer on duty of the fire department during a COPI exercise. This provides an indication of the message load, communication load and within task recovery associated with the information flow to and from the officer on duty. The direct and indict beneficial effect of experience, furthermore, was illustrated by making the within task recovery explicit. Finally, the rationale behind the additional lines of code that had to be implemented in the AWS for it to able to let both sender and receiver(s) communicate simultaneously was presented.

The communication load model showed the desired behaviour, providing a plausible indication of the load associated with handling messages during emergency response situations. As was illustrated with Figure 7.5, adding time pressure and experience resulted in fluctuations of the base message load that remain between plausible boundaries, accounting for both situational and personal fluctuations within and between emergency responders. The differences in communication load, due to experience and time pressure, are small enough to remain between plausible boundaries, and are large enough to be able to distinguish between the load associated with combinations of different levels of experience and time pressure.

8. Proof of Concept Simulation

Leading to this chapter, the ingredients were presented that together answered the main research question that was posed in Chapter 1: How can we build and ground a generic model to simulate emergency response and adaptively provide information about the workflow, workload and communication load of emergency responders as a function of the information that is exchanged? With the use of Brahms (Chapter 3) a grounded (Chapter 2) template of emergency response (Chapter 4) was constructed that made an approximation of the workflow (Chapter 5), the workload (Chapter 6) and the communication load (Chapter 7) of the individual emergency responders involved in the emergency response.

Using the detailed real time communication data that was acquired during a COPI exercise (see section 2.3.3), this chapter will present a proof of concept simulation. A proof of concept here refers to a simplified realization of the AWS in which the general idea of the AWS is reflected to demonstrate its feasibility. The proof of concept simulation will be a reconstruction of a snippet of the COPI exercise in order to:

1. Demonstrate the applicability of the template model;
2. Demonstrate the behaviour of the adaptive elements within the AWS that serve the goal to approximate the workflow, workload, communication load and processing capacity of the emergency responders that are modelled;
3. Demonstrate how an agents' processing capacity influences the workflow;
4. Demonstrate the influence of having or not having information on the workflow.

While the first three goals demonstrate and test the AWS, the latter demonstrate the AWS' applicability to be used as a test bed for the information distributor (see section 1.1).

In section 8.1, the scenario snippet on which the proof of concept simulation will be based is presented, testing the applicability of the template (proof of concept goal 1). Starting with an initial run of the simulated snippet of COPI emergency response approximating the workflow, workload, communication load and processing capacity during the COPI, section 8.2 will demonstrate and test the adaptive elements in the AWS (proof of concept goal 2) and assess the influence of the processing capacity and the effect of having or not having information on the workflow (proof of concept goals 3 and 4).

8.1 Proof of Concept Scenario

The goal of the template model presented in Chapter 4 is to provide the modeller with a generic template of the reoccurring elements of emergency response. It supports the modeller by predefining the inheritance structure of the classes and instances in the model, allowing the modeller to focus on the elements of interest. This section will apply the template model to model the organizational assembly, the agents, objects and locations based on the information that is exchanged at the COPI level of the emergency response organization during the timeframe that is addressed by the proof of concept simulation. The focus of the proof of concept simulation will be on the emergency responders in the field.

The timeframe that is addressed in the proof of concept simulation describes the situation in which the emergency response organization is faced with a potential threat when one of the fire fighters discovers a suspicious backpack, most likely containing a bomb. The discovery of the suspicious backpack results in a priority change for the emergency response organization that was well on the way handling the consequences of the collision between a packed passenger train and a military flatbed truck carrying a Leopard II tank. A more detailed description of the entire scenario can be found in section 2.3.1. During the unfolding of the proof of concept scenario, a total of 215 messages were exchanged in 9 minutes and 36 seconds incorporating 739 information elements.

Describing the initial state leading towards the proof of concept timeframe, the fire department deployed one fire fighting platoon including an officer on duty to handle the consequences of the collision between the passenger train and the military flatbed. During the proof of concept scenario, the platoon –that consists of 4 fire department units- is still active. Unit 110 is deployed in the construction pit attending the injured and trapped victims in the first two train compartments; unit 120 is deployed at the military flatbed truck attempting to free the driver and the co-driver; and, finally, units 130 and 140 are providing assistance in the pit area, and are inventorying the number and severity of the victims in the last two train compartments (compartments 3 and 4).

The medical services start off by sending four ambulance units (units 54 to 57) and the officer on duty from the medical services to the scene. Shortly after the officer on duty has reached the scene, 6 extra units are being deployed and are stationed at the ambulance collection point near the incident. Eventually, 21 more ambulances come to the scene coming from the neighbouring security regions and Germany. A helicopter of the mobile medical team (trauma team) is furthermore rushed to the scene, to provide medical care in the second compartment of the accident train. Unit 55 is providing medical assistance to the driver and the co-driver of the military flatbed truck. Units 54, 56 and 57 finally are deployed in the construction pit.

The police department deploys an officer on duty and 16 police units (each consisting of two police officers) that maintain the cordon around the incident location. The police department initially plays a limited role before the proof of concept scenario. In the time span covered by the proof of concept scenario this, however, changes when a police officer assesses the threat of the backpack.

Other parties that are involved in the emergency during the proof of concept scenario snapshot, are the dispatchers, other COPI members, regional operational team, regional policy team, victims, bystanders.

Although incorporating all the static elements that lead towards the initial state in the proof of concept model will make it an accurately matching representation of the elements that appeared during the emergency response exercise, it would also lead to an over specification of the model given the goals of the proof of concept simulation. Therefore, in the proof of concept model the number of agents, objects and locations represented was limited. The proof of concept simulation incorporates:

- Emergency responders that handle information during the time span addressed in the proof of concept;
- People, objects and locations that are the topic of communication during the time span addressed in the proof of concept;
- People, objects and locations that play a prominent role during the time span addressed in the proof of concept.

Figure 8.1 shows the agents that handled information during the proof of concept scenario. This is a subset of the total group of emergency responders that make up the information flow at the COPI level (see Figure 2.9). These include the four officers on duty; the officer on duty from the police department (OvD-P); the officer on duty from the fire department (OvD-B); the officer on duty from the medical services (OvD-G); the officer on duty from Prorail (company responsible for construction, maintenance and safety of the railway infrastructure in the Netherlands); the dispatchers from the police department (Twente) and the medical services (MK); the commanding officers from the units in the field from the fire department (units 110, 120, 130), the police department (units 1102, 4201) and from the medical services (unit 54); and finally the head officer on duty (H-OvD) and the operational leader (OL).

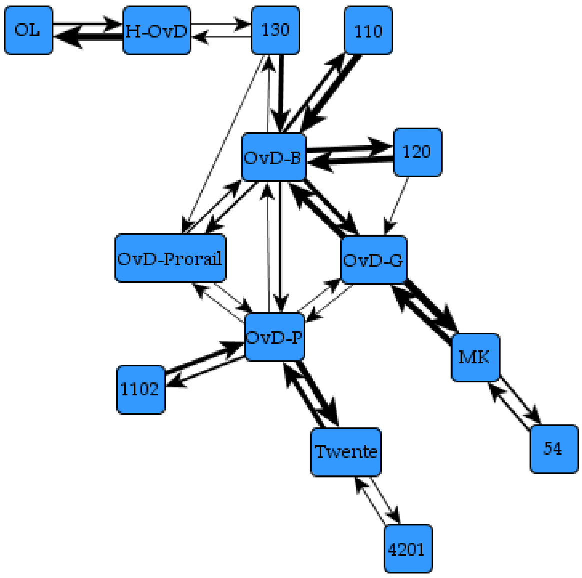


Figure 8.1: *Exchange of information elements between emergency responders during the backpack scenario.*

With the exception of the scenario specific agents, such as the officer on duty from Prorail and the units that were dispatched by the police department and the medical services, all emergency responders were pre specified (with the exception of their identifying characteristics) in the AWS template model at the agent level (Figure 4.4). The instances of these agents, therefore, were automatically included in the model. The police department units and the units from the medical services were present at the role "class level" in the

AWS template that groups similar roles. These scenario specific instances were added to the proof of concept model.

In addition to the senders and the receivers, the agents that were to become a topic of conversation also were included in the proof of concept simulation. Table 8.1 shows the abstract and concrete references to agents that were included in the proof of concept model.

Table 8.1: *Abstract and concrete agent references that appeared in the information flow during the proof of concept scenario apart from the senders and receivers.*

Abstract	Concrete
Medical services	Flatbed truck driver
Ambulance unit 55	Flatbed truck co-driver
Ambulance unit 56	
Victims	
Fire department	
Police officers	
Regional operational team	
NedTrain	
Bomb team	
Terrorists	
Bystanders	

In the AWS template, 6 out of the 11 abstract references were pre specified in the AWS template. These referred to the conceptual objects representing the abstract collections of people from the medical services; victims and bystanders (part of the `Civil_grp` in Figure 4.4a); fire department; police officers; and the regional operational team. The ambulance units were partly pre specified in the sense that the ambulance unit itself was pre specified in the AWS template, but the instantiation of units 55 and 56 was not. The bomb team, terrorists and the abstract reference to NedTrain (company responsible for the maintenance of trains) had to be added to the model. Furthermore, given the emergency specific character of the two concrete agents that were the subjects of conversation, the driver and co-driver also were not pre specified in the model and needed to be added.

Based on their location at the start of the proof of concept scenario, 8 other agents were included in the AWS that were not present in the communication and did not actively send or receive information from the officer on duty during the proof of concept. These included, fire department unit 140 (5 fire fighters and a commanding officer) and the mobile medical team (one doctor, a trauma nurse and a pilot). These agents were located either in the construction pit, or in one of the train sections. Their workflow would significantly be affected by the discovery of a bomb in the construction pit. With the exception of their identifying characteristics, these agents were all pre specified in the AWS model.

The objects that surfaced during the proof of concept scenario also concerned concrete and abstract references. During the communication, concrete objects turned up, such as the train, train segments, backpacks, military flatbed truck and walkie-talkie. These concrete objects mostly referred to scenario specific elements where the abstract objects concerned generic aspects of the emergency. Abstract objects, for example, referred to the emergency, the COPI meeting, the threat level and the previous message. The additional objects that were added to the proof of concept model concerned the pump-water-tenders that were used by units 110 to 140, the ambulances that were used by units 54 to 58 and the helicopter that was used by the mobile medical team. With the use of the hierarchal template presented in section 4.3, the AWS template model was able to incorporate into its hierarchal structure most objects that were the topic of communication in the proof of concept scenario, pre defining most of the attributes needed. Some objects however could not easily be placed, such as a noise barrier and provisions (food and drinks). These were added separately.

The agents and objects were situated in locations. The locations that were used during communication in the proof of concept scenario are shown in Table 8.2. In total 15 locations were communicated about.

Table 8.2: *Locations that surfaced in the communication during the proof of concept simulation.*

Location	Explanation	Extends / instance of (see Figure 4.18)
Construction Pit	Source area	Railway section (extends Railways)
Hospitals in the city of Enschede	Used to indicate where extracted victims went	Building
Almelo	City in which the accident occurred	Municipality
Egbert Gorterstraat	Location of bomb team (near the courthouse)	Roads and Street
Brouwerijstraat	Casualty collection point	Roads and Street
Ambstraart	Casualty collection point	Roads and Street
Louise van Haeftenplantsoen	Casualty collection point	Park
Train sections 1 to 4	Train sections of the accident train	Train section
Preston Palace	Casualty collection point	Building
Parking lot Polman soccer stadium	Ambulance collection point	Roads and Street
Railway station Almelo de Riet	Railway station	Building
Rembrandtlaan	Meeting point	Roads and Street

The locations concerned indoor and outdoor locations and addressed various levels of detail, ranging from representing the inside of the train sections to representing the entire municipality of Almelo. The location that appeared most frequently was the construction pit. The construction pit was used as a placeholder for the units and objects situated in that area and was communicated about with regard to the local working conditions, the number of victims located there and its general properties, such as its depth. Train section one and two (objects), were situated in the construction pit. The inside of the train sections also came up during communication, describing the number of victims, the working conditions and the tasks that were performed by the emergency responders. The other locations were used to describe the casualty collection and meeting points for victims; abstract agent locations (bomb team, ambulance units); general attributes of the city of Almelo or attributes of a hospital in Enschede.

In addition to these locations, three other locations were added to the proof of concept model: the COPI meeting location; the road on which the fire department units were parked; and the location where the flatbed and the rescue activities of the driver and the co driver of the flatbed were situated. As can be seen in Table 8.2, these locations could all be mapped onto the predefined location types shown in Figure 4.18.

To summarize, the AWS is able to support the modeller in constructing an agent, object and location model, providing a general framework of elements that appear during emergency response, leaving only emergency specific elements to be modelled additionally.

Regarding the agent model, the AWS provides the possibility to model multiple levels of details of agents. Firstly, concrete and abstract references to agents could be modelled, as well as the identifying characteristics of agents and groups. The identifying characteristics, however, mostly remained untouched and were not used in the proof of concept scenario. Secondly, while it was clear during the emergency response which roles the emergency responders had, their specific affiliation often remained unclear. For example, one knows that the unit commander of unit 140 has the role of unit commander, but one does not know of which fire department he is a member. By separating groups and organizations, this, however, was no problem in the AWS, since activities are role restricted rather than organization restricted. The flexibility in assigning agents to groups, in addition to the hierarchal structuring of these groups, furthermore, provided the opportunity to model an agent at the desired level of detail. In the example of the unit commander, although one does not know which fire department he is a member of, we do know that the unit commander works for the fire department. Incorporating agents with organizational specific knowledge thus only entails making them a member of a more specific organizational group.

The abstract and concrete objects and locations that turned up during the communication in the proof of concept scenario could be placed in the pre defined hierarchal structure. It can be concluded that the AWS template aids the modeller by providing a generic modelling architecture for emergency response modelling.

The following section will present the application of the proof of concept model to demonstrate the behaviour of the adaptive elements within the AWS that serve the goal to approximate the workflow, workload, communication load and processing capacity of the emergency responders that are modelled. This demonstrates how an agent's processing capacity influences the workflow and the assessment of the effect of having or not having information on the workflow.

8.2 Adaptive Emergency Response Simulation

For the AWS to operate as a function of the information that is exchanged, it first has to represent the information flow. In the AWS, the flow of messages is regulated by a separate agent: the scenario agent. The scenario agent distributes the messages and beliefs incorporated in the messages in the same order as they were exchanged during the COPI emergency response exercise, to the persons that were involved in the communication. It, furthermore, provides the senders and receiver with beliefs that trigger the activities that are involved when handling information (see section 7.3). Figure 8.2 shows the activities involved with handling information over time, for the three officers on duty whose communication was monitored and the scenario agent. Each message handling activity is shown as a block that is placed on the horizontal time axis. The width of a block thus represents the time occupied by handling messages by the agent.

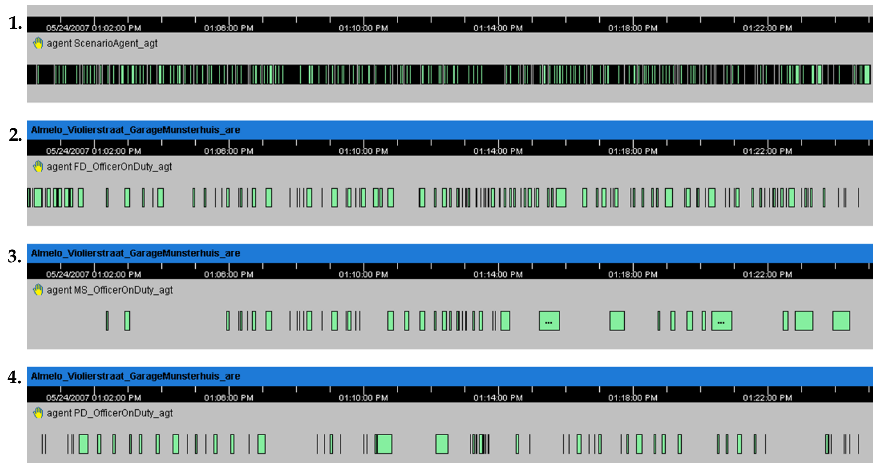


Figure 8.2: *AWS Information handling activities*

Bar 1 in Figure 8.2 shows the activities of the scenario agent (distributing information) over time. As can be seen by the density of the blocks representing the activities, the scenario agent is constantly engaged in distributing information to all agents incorporated in the model. In the second bar, the information handling activities of the officer on duty from the fire department is shown; the third bar represents the message handling activities of the officer on duty from the medical services; and bar four represents the message handling activities of the officer on duty from the police department. The communication activities of all emergency responders that handle information in the proof of concept scenario is represented in this way in the AWS.

With the use of the communication load model, this load over time, associated with handling the messages, can be approximated for each emergency responder incorporated in the model. The scenario agent thus provides the AWS with the ability to firstly model the information flow, secondly model the activities associated with handling information and thirdly assess the load associated with handling information.

When adding “between task recovery” to the model, the impact of message handling on the processing capacity can be approximated for the hypothetical situation that no other tasks are being executed besides handling information. Figure 8.3 shows the results from a simulated run concerning the fluctuations of the available processing capacity (APC) for the officer on duty from the fire department and the officer on duty from the medical services during the proof of concept scenario, while varying their experience levels (5 years of role experience and 15 years of role experience). Figure 8.3, furthermore, illustrates how these fluctuations are influenced by message density by syncing both time lines from the information handling activities of the officer on duty of the fire department and medical services (Figure 8.1) and the APC fluctuations.

When no messages are handled, between task recovery “refills” the processing capacity. As a result the impact of handling a message on the processing capacity can be undone. This can be seen in the initial 365 seconds of handling messages for the officer on duty from the medical services. During this time interval he receives two messages, but fully refills the processing capacity.

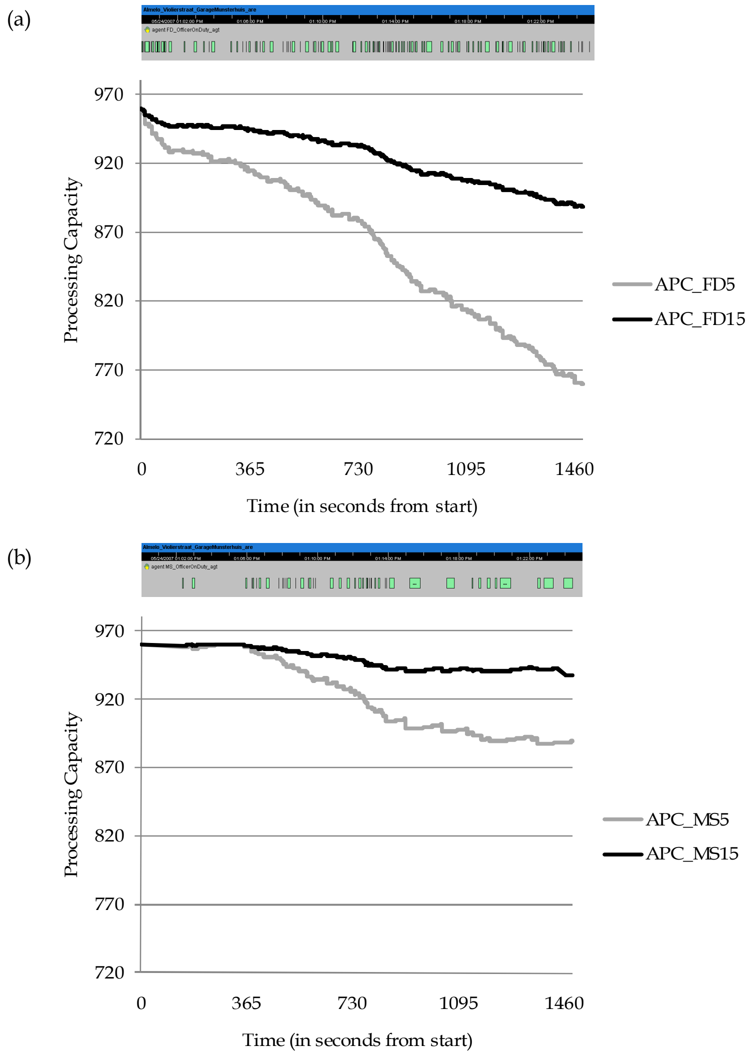


Figure 8.3: *Fluctuations of the available processing capacity (APC) due to message handling over time (in seconds from the start of the simulation) for the officer on duty from the fire department (a) with 5 years of role experience (APC_FD5) and with 15 years of role experience (APC_FD15); and for the officer on duty from the medical services (b) with 5 years of role experience (APC_MS5) and with 15 years of role experience (APC_MS15).*

In the proof of concept simulation, the officers on duty thus are either handling information or are "refilling" the processing capacity. The impact on the processing capacity associated with handling messages for the officer on duty from the fire department is higher when compared with the impact on the processing capacity associated with handling messages for the officer on duty from the medical services. The processing capacity of the experienced (15 years of role experience) emergency officer on duty from the fire department is depleted with 71 (M = 0.61, N = 117 messages), while the processing capacity of the officer on duty from the medical services is depleted with 17 (M = 0.39, N = 45 messages). For less experienced emergency responders (with 5 years of role experience), the processing capacity of the officer on duty for the fire department is depleted with 200 (M = 1.71, N = 117 messages), while the processing capacity of the officer on duty for the medical services is depleted with 68 (M = 1.51, N = 45 messages). Given the message handling pattern in the proof of concept scenario, the amount of processing capacity depletion thus is amplified 2.8 times when comparing the less experienced officer on duty of the fire department with the more experienced officer on duty from the fire department; and is amplified 3.9 times when comparing the less experienced officer on duty from the medical services with the more experienced officer on duty from the fire department. The number of years of role experience thus has a different amplifying influence on different message flows.

Up to this point, the situation was simulated where the officers on duty were not performing any tasks and therefore were able to recover. Figure 8.4 shows the simulated workflow of the officer on duty from the fire department during the proof of concept simulation in which tasks are included.

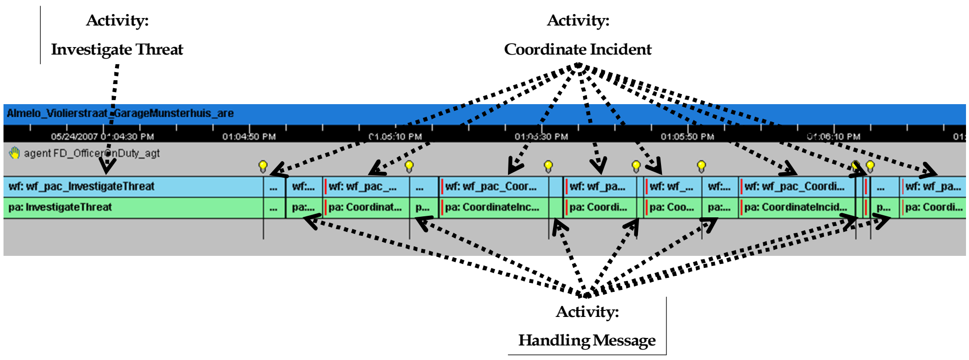


Figure 8.4: *Workflow of the officer on duty during a part of the proof of concept scenario*

At the start of the snippet presented in figure 8.4, the officer on duty is investigating the threat leading to the conclusion that there was no direct threat for the units in the field. After having concluded that there is no threat, the officer on duty starts the activity of coordinating the incident. During this activity, the officer on duty is interrupted several times to either send or receive a message. The officer on duty thus is constantly engaged in activities, leaving no room for between task recovery. As a consequence, the processing capacity is not refilled by between task recovery. It should however be noted that within task recovery still is active. The officer on duty thus can recover by performing relatively easy activities.

Figure 8.5 shows the depletion of the processing capacity of the officer on duty from the fire department while performing activities and receiving information while varying the number of years of role experience (5 and 15 years). The tasks that are performed have an initial task load of 14.78 for the activity of investigating a threat and 12.78 for coordinating the incident. Both tasks consisted for 98% of mental activities and 2% of physical activities. The activity of investigating a threat had a duration of 200 seconds, and the activity of coordinating the incident had a duration of 300 seconds. It should be noted that these values do not represent the actual values associated for these activities, but are assigned for illustrative purposes. While the task of investigating the threat is triggered by a belief, the coordinating incident task is “looped”. When the coordinating incident task is finished it is put on the “to do” list again, and is again ready for execution, making it a background task that is continuously active unless it is interrupted or overruled by higher priority activities.

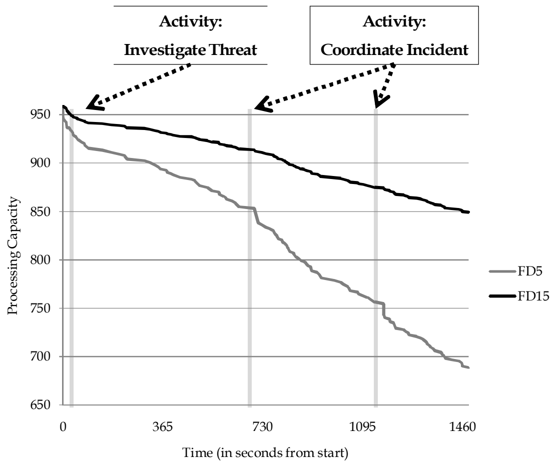


Figure 8.5: *Depletion of the processing capacity of the officer on duty from the fire department while performing activities and handling information while varying the numbers of years of role experience.*

As can be seen in Figure 8.5, the lines do not “spike” anymore (indicating no between task recovery), but gradually decrease. Whenever a task is executed (vertical lines), this task impacts the processing capacity. The total impact of the information and the tasks on the processing capacity, without the ability to recover between tasks, is 110.8 for the officer on duty with 15 years of role experience, and 271.3 for the emergency responder with 5 years of role experience. When looking at the impact of the tasks, it can be seen that the impact of the task execution is minor for the officer on duty with 15 years of role experience; the experience in combination with within task recovery reduces the impact of the task to nearly zero.

This remains so, even when the tasks are performed under time pressure. Figure 8.6 shows the decrease of the processing due to task performance and message handling for the officer on duty from the fire department varying the number of years of role experience, while performing these tasks in time pressured conditions. For the officer on duty with 5 years of role experience, the total impact on the processing capacity in the proof of concept simulation is 145.1. For the officer on duty with 15 years of role experience, the processing capacity is depleted by 342.6. Compared with the previous simulation, the degree of depletion increased with 71.3 for the officer on duty with 5 years of role experience, and increased with 34.3 for the officer on duty with 15 years of role experience, which equals the amplification value due to time pressure on the damping effect of role experience on the relationship between task and workload, and message load and communication load.

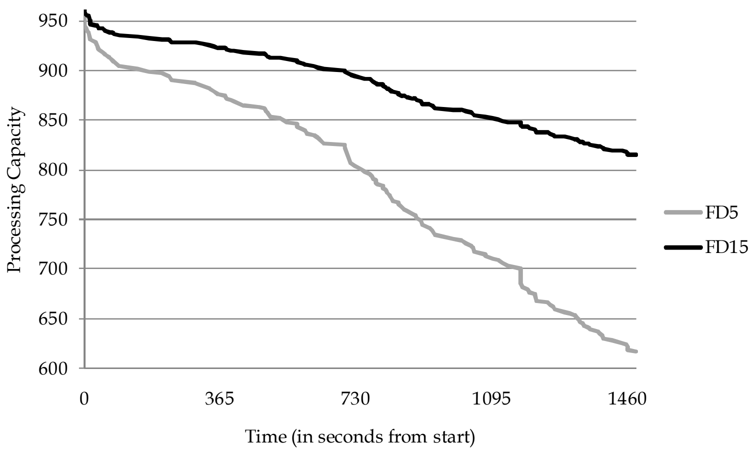


Figure 8.6: *Depletion of the processing capacity of the officer on duty from the fire department while performing activities and handling information while varying the numbers of years of role experience (5 years of role experience (FD5) and 15 years of role experience (FD15)).*

The simulations of the proof of concept scenario demonstrate the fluctuations of the available processing capacity due to different information flows, different levels of emergency responder experience and the presence of time pressure. When provided with the opportunity to recover between activities, emergency responders in the AWS are able to recover from their activities (by taking a break from work). As was demonstrated with the information flow of the officer on duty from the medical services, recovery was able to offset the impact of message handling on the processing capacity. However, as was also demonstrated, an increase of the number of years of role experience in combination with the duration of an activity, could also lead to a decreased depletion of the processing capacity, or even a "refill" of the processing capacity. When no between task recovery is possible, one thus is still able to recover. Finally, the amplifying effect of time pressure on the damping effect of role experience on the relationship between task load and workload and between message load and communication load was demonstrated.

As was indicated in Chapter 6, overstepping the depletion level of the processing capacity changes the workflow, as it forces the emergency responder to first recover before commencing with another task. The time to complete the activities so is lengthened with the time needed for recovery. Figure 8.7 demonstrates this for the workflow of fire fighter 112, who is engaged in the activity of rescuing victims that are located in the construction pit. The workflow of unit 112 (`FD_112_agt`) is compared with the workflow of unit 111 (`FD_111_agt`), which also is engaged in rescue activities. Within this model the activity of "rescuing" is looped in a similar fashion as the "coordinate incident" task was looped for the officer on duty. In addition, the rescuing activity also was synchronized between agent 111 and agent 112 for comparison reasons.

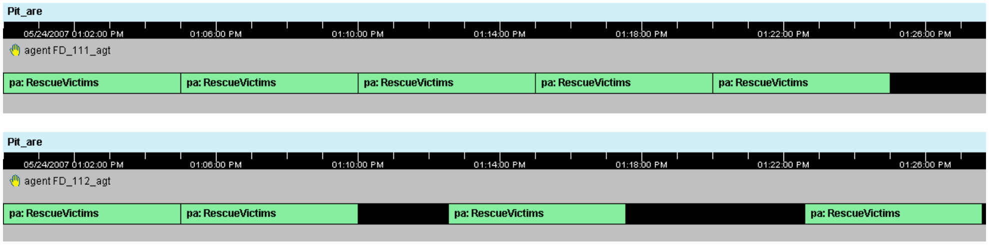


Figure 8.7: *Influence of processing capacity depletion on the workflow of fire fighter 112.*

As can be seen in Figure 8.7, both agents start with their rescue activities. However, when the task is executed for the second time, agent 112 oversteps his processing capacity and needs to recover (black blocks in the time line) before he can commence with the next task. As a consequence of this task, fire fighter 112 oversteps the processing capacity again, again needing recovery after the task is executed (see Figure 6.15). As a consequence of overstepping the processing capacity, more time is needed to execute the four tasks compared to the time agent 111 needs, and task execution may run out of sync (see Figure 8.7).

A secondary effect of overstepping one's processing capacity in the AWS is that it also influences the task of sending information, since this is planned as a work task. Information, furthermore, influences the workflow since it triggers activities of other emergency responders. Depletion of the processing capacity so can lead to a snowballing effect, affecting several emergency responder incorporated in the model.

In the AWS activities are related to the roles that are performed by the agents. As we saw in Figure 8.4, which described a part of the workflow of the officer on duty from the fire department, the officer on duty was involved in "investigating the threat" and "coordinating the incident". While the second task was a looped activity that is active during the entire proof of concept simulation, the first task is triggered by a belief about a possible bomb at the incident location. Whenever the beliefs are communicated that `CurrentThreat.present("bomb") = unknown` and `CurrentThreat.location("bomb") = [incident location]`, the officer on duty will engage in the activity of investigating the threat, making the activities associated with investigating the threat active.

In the proof of concept scenario no bomb initially was found. However, to assess the influence of having or not having information on the activities of the officer on duty, the belief that was send by the officer on duty from the police department, indicating that there was no bomb present (`CurrentThreat.present("bomb") = false`) was replaced with `CurrentThreat.present("bomb") = true`. Figure 8.8(a) and (b) show the last part of the original workflow of the officer on duty during which he is provided with information about the possible presence of the bomb. In Figure 8.8(a), the original workflow is shown, and in Figure 8.8(b) the adjusted workflow is shown, caused by the incorporation of the belief that a bomb is present at the incident location.

As can be seen in Figure 8.8(a), the workflow of the officer on duty remains unchanged (the task of coordinating the incident remains active) since no actions have to be taken based on the information that is provided to the officer on duty.

(a)
(b)

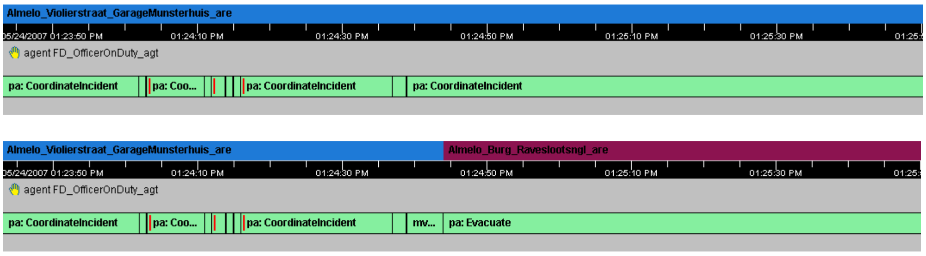


Figure 8.8: *Influence of information on the workflow of the officer on duty from the fire department*

However, when the belief is changed due to a message from the officer on duty from the police department to the officer on duty from the fire department the workflow of the latter changes, making the evacuation procedure active, and causing the officer on duty to meet with the commanding officers in the field. In the AWS this translates to the officer on duty activating a series of activities so they can be planned using the AWS workflow planner. As can be seen in Figure 8.8(b), first a move activity is activated (`mv..`) that moves the officer on duty from his initial location `Almelo_Violierstraat_GarageMunsterhuis_are` where the COPI meeting unit is located to the location where the unit commanders that are assigned to the construction pit are located (`Almelo_Burg_Raveslootsingl_are`). Once the officer on duty is at that location he starts the evacuation procedure. The fact that the simulation is able to adaptively construct the workflow based on the information that is exchanged demonstrates that the AWS is capable to function as a test bed to assess the effect of information on a workflow.

8.3 Conclusion

This chapter demonstrated subsequently the applicability of the template model, the behaviour of the adaptive elements, the influence of an agent's processing capacity on the workflow, and the influence of obtaining certain information on the workflow. In this way, the general idea and the feasibility of the AWS, that was developed in this dissertation, to simulate emergency response as a function of the information that is exchanged providing information about the workload and communication load of the emergency responders was evidenced.

Is also demonstrated that the elements included in the template model simplified the modelling process by pre specifying most of the elements that appeared within the proof of concept simulation (organizational assembly, objects and location).

Finally, the behaviour of the adaptive elements which are incorporated in the AWS that serve the goal to approximate the workload, communication load and processing capacity of the emergency responders, provided a plausible indication for the load associated with handling information and performing tasks leading to changes in how the workflow unfolds over time.

It should, however, be noted that the task scenarios that were used to assess these effects were fairly basic, not entirely doing justice to the complexity of the tasks that are performed in practice. Given the limited number of tasks, the proportion of the decrease of processing capacity due to handling messages and executing activities therefore could not be assessed. Since messages handling activities and normal tasks differ both in frequency and duration the total decrease of the processing capacity could spin out of balance. As was indicated by the experts in the field of emergency response, both activities determine about the same proportion of the experienced load during the emergency response. The proportion of the total experienced load during an emergency response due to the execution of activities as was indicated by the experts was 21% (N=23, SD=9.26) where the load associated with handling information was 26% (N=23, SD=6.84).

Implementation of the proof of concept scenario did indicate the potential of the AWS to provide information regarding the load associated with handling messages and the load associated with task execution, and to on the fly adapt the workflow during an emergency response based on the information flow.

9. Summary and Conclusions

The adhocratic emergency response organisation has to operate in a fast changing environment changing its assembly to the demands posed by the emergency situation. A relatively large proportion of time and effort is put into communication to coordinate the multidisciplinary activities, possess up to date information about the emergency situation and possess up to date information about the present assembly of the emergency response organisation. Information sharing plays an important role, influencing the course of the emergency response and subsequently the workflow, workload and the load put on the emergency responders related to handling information. Recent incidents and major training exercises in and outside the Netherlands have persistently shown that not having or not sharing information during emergency response are major sources of emergency response inefficiency and error, and affect incident mitigation outcomes through workflow planning that is based on this information.

In this dissertation the grounded development and initial testing of the Adaptive Workflow Simulator (AWS) was presented. The goal of the AWS is to simulate and on the fly provide information about the current state of the workflow, workload and communication load of emergency responders during emergency response situations as a function of the information that is exchanged. In the light of the broader TAID project of which this dissertation is a sub project, the AWS supplies this contextual information to an Information Distributor (the other sub project) that, based on machine learning techniques, aims at optimizing information dissemination in emergency response situations. As a function of the information flow, the AWS is used to approximate the workflow; approximate the strain put on the emergency responders as a consequence of task execution and handling information, and finally provide a test bed for the ID to assess the impact of improved information distribution on the workflow.

The general research question that was addressed in this dissertation was "How can we build and ground a generic model to on the fly simulate emergency response and adaptively provide information about the workflow, workload and communication load of emergency responders as a function of the information that is exchanged?" To answer this question, the general research question was broken down in 6 sub questions.

1. What additional developmental and methodological considerations have to be taken into account within the development of the AWS to simulate emergency response?
2. Which modelling and simulation environment is best suited to model the AWS?
3. How can the static elements of emergency response be represented in the AWS?
4. How can we model emergency response workflow?
5. How can we model agent specific workload in the AWS and let it be able to impact the workflow?
6. How can we model the agent specific load associated with handling messages?

In short, the abovementioned general research question can be answered by:

With the use of the Brahms simulation environment (Chapter 3) a grounded (Chapter 2) template of emergency response can be constructed (Chapter 4) that is able to make an approximation of the emergency response workflow using the workflow model (Chapter 5), the workload using a workload model that is grounded in emergency response (Chapter 6) and the communication load using a grounded communication load model (Chapter 7) of the individual emergency responders involved in the emergency response.

To test the models included in the AWS, three emergency response exercises were analysed, one past emergency response was modelled, a questionnaire was administered to a group of experts in the field of emergency response and finally, to test the feasibility of the AWS, a proof of concept simulation was conducted that, by using real time communication data, demonstrated that the AWS is able to provide an approximation of the workflow, and workload and communication load put on the emergency responder as a consequence of handling messages.

In this concluding section of the dissertation we will summarize and discuss each the sub questions posed.

9.1 What developmental and research considerations have be taken into account within the development of the AWS?

Systems developed for the field of emergency response have to make a trade off between the breadth of their application and the level of detail within the system. It was concluded that in the AWS, communication and information exchange during emergency response determine the level of detail used. The level of detail varies, depending on the communication during the response. This viewpoint reduces the number of attributes and states of a concept to be incorporated, to the number of attributes used during communication.

By having this limitation, it is made possible to create a flexible, adaptive simulation of emergency response that focuses on the aspects that significantly differentiate emergency response from day to day incident responses: workflow complexity, communication and information. By describing these reoccurring aspects in detail, and simplifying others, the simulation's applicability to other emergency types significantly increases. A general emergency response simulation template did emerge. The AWS uses a generic template based system, applicable to many situations with the possibility to increase or decrease detail. The level of detail is determined by the level of detail present in the communication during emergency responses and the goal of a particular simulation.

Secondly, the TAID project applies a grounded system centred development approach. A prominent role in the development of the AWS is reserved for grounding. Grounding refers to the degree in which assumptions and relations in a simulation are based on empirical findings, instead of being hypothetical. Grounding the development will minimize the gap between reality and the simulation, will lead to a better founded simulation and will increase the soundness of the results. The use of the adaptive workflow simulator of emergency response, which approximates and provides information about the workflow and workload of the responders to the information distributor, only has added value when it is able to provide valid information about these elements. For the AWS it therefore is crucial that it is grounded thoroughly in practice and achieves a sufficient level of functional fidelity. The AWS uses a system development focus, using data from practice to ground the elements incorporated in the system.

Valid data that can be used for grounding is difficult to acquire. This is due to the adaptive character of the response itself; the volatile and multimodal content of communication and information; lack of experimental control within and between emergencies; post emergency recall biases; the limited prescriptive value of small emergency responses for large emergency responses; and the limited generalizability of emergency response exercises to actual emergencies.

The template model, incorporating the semi static reoccurring elements of emergency response, was grounded by the use of topic analyses of the communication during two mock emergency response exercises. The mock emergency response exercises provided information about which topics generally emerge in emergency response situations. The first exercise concerned an operational emergency response exercise that dealt with a monodisciplinary emergency in which the fire department had to handle a severe traffic accident. The second emergency response exercise concerned a mock emergency response to a complex multidisciplinary emergency at the COPI level of the emergency response organisation. Based on these mock emergency responses, it was furthermore concluded that, given the central position of the OvD in the emergency response organisation concerning the genesis of information and the coordination of the workflow of the units in the field, in combination with the multidisciplinary complex open ended character of the COPI in which the OvD participates, the OvDs activities at the COPI level are the best suited level for the proof of concept simulation and for the TAID system as a whole.

To ground the workload and communication load models, a questionnaire regarding workload, communication load, recovery speed, model integration and experience was administered to a group of 26 experts in the field of emergency response. The results from the questionnaire provided us with a sufficient basis for grounding and further specifying the AWS workload and communication load model.

The data acquisition activities performed to acquire detailed (complete) real time data concerning the information exchanged at the OvD level that is used for the proof of concept simulation, entailed monitoring all information exchanged to and from the OvDs (fire department, police department and medical services) during another COPI exercise. This was achieved by monitoring the walkie-talkie channels that were used by the officers on duty and by equipping the officers on duty with microphones that recorded all face to face communication during a COPI exercise. A complete information flow resulting from communication could be created for the officers on duty that were involved in the mock emergency response, which provides a good test bed for the TAID system and the AWS.

9.2 Which modelling and simulation environment is best suited to model the AWS?

Chapter 3 investigated available simulation methods that can be used to model the Adaptive Workflow Simulator. Based on the requirements that should be met by the AWS, it turned out that a multi agent based simulation would best fit its purposes, since it allows using multiple levels to facilitate emerging behaviour, allows communication between agents (objects, environment, and actors), allows the modeller to generate complex agent models, and is able to handle a sufficient number of agents.

After the Brahms multi agent system was chosen to model and simulate work practice, a test simulation of an emergency response to a plane crash from the past was developed. Based on the result from this test simulation, it was concluded that the Brahms modelling and simulation environment made an efficient and valid representation of the emergency response possible. The object oriented approach that is incorporated in Brahms, provided us with the opportunity to model emergency response at a generic level as well as at a more detailed level, depending on the level of detail derived from the focus of the simulation. As a consequence, the Brahms modelling and simulation environment was used to develop the AWS and served as the testing environment in this research.

9.3 How can the static elements of emergency response be represented in the AWS?

In Chapter 4, the reoccurring, semi statistic elements in emergency situations were defined in the AWS. These elements are similar between emergency responses and are related to the emergency itself, the emergency response organisation, the roles and activities that are performed by these organisations, the agents that are involved in the emergency response, the concrete objects that are used during the emergency response and the geography in which the emergency unfolds. With these elements, a grounded template of emergency response was developed: the main modelling architecture for the AWS simulation in which emergency response scenarios can be simulated.

By using both empirical grounded data and literature on emergency response practice, the actual elements and the level of detail of these elements was determined. The level of detail is important for determining the reusability of the template. *Under specifying* leads to additional effort when building the actual model, while *over specifying* will increase the "payload" of the model, shifting effort towards the development phase with the danger of incorporating redundant elements. A balance has to be found concerning the level of detail needed in the template model for it to be complete, but not overly complete, without compromising the goals and the focus of the simulation.

Since the focus of the AWS is to simulate emergency response as a function of the information that is exchanged, the level of detail within the template model thus can be reduced to the level on which is communicated, decreasing the number of attributes and states of an object or agent to general attributes that surface in the literature and more detailed attributes used in communication.

The empirical grounding of the model was again based on communication during emergency response exercises at the operational level (during a traffic accident exercise) and the COPI level (during a major fire incident exercise) of emergency response.

Firstly, an initial set of attributes of the emergency was selected for implementation in the AWS. The selection was based on communication data gathered during the two emergency response exercises, in combination with the analysis of 18 emergency types that could be identified in the literature.

Secondly, the emergency response organisational structure, as it is seen in practice, was rebuild in the AWS, incorporating multiple group membership (role, organisation, team) to structure the inheritance of attributes that propagate down from these groups.

Thirdly, the level of detail of the identifying and physical attributes of the emergency responders, was determined by applying the attributes that are incorporated in the "standard" casualty card that is used by emergency personnel to describe the relevant medical and identifying attributes of a person.

Fourthly a general model of the objects that are used, used up, encountered or communicated about in emergency response situations and abstract objects which are not situated, do not undergo state changes and are solely used to model communication about non tangible abstract concepts, were included in the AWS template. Concrete objects were classified into 4 main categories: objects that are used for communication (Communication objects class); that are used as equipment by the emergency services (Equipment class), that are used as a the basic means of transport (Means of transport class), and that can be used as add-ons to the basic means of transport containing a collection of specialist equipment (Trailers class). Abstract objects are used to model abstract agent group references (such as "*the* police"), references to collections of objects (such as " ambulances"), abstract concepts (such as "the emergency") and attributes of meetings and gatherings (such as starting time, location).

Finally, the geography enables the AWS to situate the emergency response and enable the geographic instances to interact with agents and objects. The geography representation that is used in the AWS serves as a placeholder (locating agents and objects), as an information provider (providing location information to its inhabitants), and as the framework that is used to model movement of agents and objects between locations.

The selection of elements and the level of detail of these elements and attributes of the elements that were included in the AWS template model were grounded in emergency response practice. The implementation of the proof of concept scenario furthermore showed that the AWS is able to support the modeller in constructing an agent, object and location model, providing a general framework of elements that appear during emergency response, leaving only emergency specific elements to be modelled additionally.

9.4 How can we model emergency response workflow?

The workflow in emergency response practice is characterized by the combination of adaptive execution of elements of textbook workflows and the execution of emergent work. Four basic workflow types can be defined in this operational practice, where the AWS workflow model must be able to simultaneously engage in: the textbook workflow, the precondition workflow, the priority workflow and the interrupt workflow, describing the different relations between tasks within the total workflow.

Simulation results show that the AWS workflow model can sufficiently represent all these workflow types that are needed to make an enhanced workflow model for tasks that are performed during emergency response. It is shown that the AWS workflow model is able to deal with the adaptive and agile work structuring that is commonly seen in emergency response practice. The AWS workflow model can structure agent activities based on agent specific variations in the enabling conditions (precondition workflow), incorporate rigid elements in the simulation by including the possibility to use predefined task execution order (textbook workflow), and can use the subjective priority of tasks to structure activities (priority workflow) or interrupt to interrupt the workflow (interrupt workflow). These workflow types furthermore exist simultaneously within the AWS, making a fusion of or combinations between these workflow types possible on the fly.

Based on the formulated requirements for a workflow model describing emergency response, it can be concluded that the general workflow model used within the AWS workflow model meets the requirements for an enriched workflow representation. The generic character of the AWS workflow model provides us with the opportunity to use the AWS workflow model in an emergency independent manner.

9.5 How can we model agent specific workload in the AWS and let it be able to impact the workflow?

To define how agent specific workload can be modelled in the AWS, the relationship between workload and the processing capacity was explored. Using the analogy of a processing capacity reservoir, it was concluded that the present available processing capacity is the result of the starting level of the reservoir minus the cumulative historic workloads posed on the task executer plus the cumulative amount of processing capacity due to recovery between and during the execution of tasks.

The reservoir thus is depleted by the workload associated with executing tasks and is refilled by recovering between and during tasks. The individual responder workflow is influenced by using the depletion level as a precondition for task execution. Tasks can only be executed when the processing capacity is not below the depletion level. This results in a time delay in the workflow due to the added duration of recovery between the tasks.

When no moderating factors of the relationship between task load and workload would be present, it could be concluded that task load = workload. It, however, was argued that the relationship between task load and workload is moderated by several moderators residing in the task environment and the task performer. The main environmental moderator is time pressure and the main performer related moderator is experience.

In the resulting workload model, firstly, workload is determined by the task load. Secondly, the relationship between task load and workload is negatively moderated by the number of years of role experience of the emergency responder. Thirdly, time pressure amplifies the damping effect of experience on the relationship between task load and workload. And, finally, the number of years of role experience has a positive influence on the task recovery speed.

This workload model was grounded in emergency response by administering a questionnaire addressing the relationships in the model to experts in the field of emergency response. The three main conclusions based on the experts' opinions, expressed in the questionnaire, were that firstly, the number of years of role experience only negatively moderated the mental elements in a task and not moderate the physical ones. Secondly, the number of years of role experience had no direct influence on the recovery speed, and thirdly, when present, time pressure accounts for 26 percent of the workload.

These findings resulted in an adjustment of the AWS workload model. According to the experts in the field of emergency response, the factors of workload, recovery and consequently the processing capacity, do not fully support the initial model. Besides grounding the theoretical model, the experts in the field of emergency also provided information about the type of relations between the elements in the model, which are used to quantify and calculate the processing capacity, workload and recovery.

Using the AWS workload model, it is possible to on the fly approximate the workload, processing capacity and the recovery of individual emergency responders by only specifying 6 variables, which all can be defined in advance. Four variables are linked to the task in order to determine task load; one variable is a value representing the number of years of role experience of the emergency responder and one value provides an indication of the presence of time pressure. Since the workframes and thoughtframes responsible for the workload and processing capacity calculations, the starting processing capacity, the presence of time pressure and the task attributes are defined at a high level in the inheritance tree, they are automatically available to all emergency responders. By modifying these values, they thus automatically are modified for all emergency responders in the model. They, however, can be overruled by the children groups or individual agents, adjusting the values locally, bringing flexibility to the simulation. A generic model emerged which can be adjusted easily at a the agent level.

To summarize, using the workload model implemented in the AWS, we have build a generic and grounded model of the workload, processing capacity and the influence of workload and processing capacity on the workflow, which lets us on the fly approximate the workload and processing capacity of individual emergency responders in the AWS. In this way the global behaviour of emergency responders, as far as the workload factors influence the execution of tasks is concerned, is taken into account in the simulation, increasing the functional fidelity of the simulation.

9.6 How can we model the agent specific load associated with handling messages?

The communication load model, that is a specific version of the workload model presented in Chapter 6, uses message load as the main determinant of communication load. The relationship between message load is moderated by the number of years of role experience of a responder, whose effect on this relationship is damped by time pressure.

Message load refers to the load of a message, independent of the person who is handling it or the conditions under which it is handled. The message load is determined by the number of information elements that the message contains and increases exponentially up to a point with each message added. The maximum message load is 20, relating to the maximum score of the NASA-TLX used in the workload model.

The communication load model was grounded by incorporating the experts' opinions in the development in the model and by testing the model's behaviour on real communication data based on the information handling activities of the officer on duty of the fire department during a COPI exercise. This provides an indication of the message load, communication load and within task recovery associated with the information flow to and from the officer on duty. The direct and indict beneficial effect of experience, furthermore, was illustrated by making the within task recovery explicit. Finally, the rationale behind the additional lines of code that had to be implemented in the AWS for it to able to let both sender and receiver(s) communicate simultaneously was presented.

The communication load model showed the desired behaviour, providing a plausible indication of the load associated with handling messages during emergency response situations. As was illustrated, adding time pressure and experience resulted in fluctuations of the base message load that remain between plausible boundaries, accounting for both situational and personal fluctuations within and between emergency responders. The differences in communication load, due to experience and time pressure, are small enough to remain between plausible boundaries, and are large enough to be able to distinguish between the load associated with combinations of different levels of experience and time pressure.

General Conclusions and Application of the AWS

The AWS provides a generic simulation architecture to model the workflow and approximate the workload and communication load during emergency responses. In contrast to most research into the physical and mental load in the domain of emergency response, the AWS is the first approach that separates load associated with handling messages and the load associated with the execution of tasks, acknowledging the additional load associated with handling messages on the emergency responders besides the load already present and associated with task execution, and secondly focuses on the multidisciplinairy emergency response as a whole, instead of on a subset of tasks for a subset of emergency responders. This enables the AWS to make explicit both interdisciplinary and intra disciplinary differences regarding the workload and communication load.

By approximating the workflow based on the information flow and the enhanced workflow planner, the AWS is able to structure the simulated workflow of the emergency responders in an adaptive manner that is different from commonly used approaches since the resulting workflow of the AWS is information dependent rather than process dependent, capturing the dynamics of the workflow of emergency responders in this way.

Two evident practical application of the AWS in the field of emergency response, for example, are the evaluation and development of multidisciplinary protocols and the evaluation of the workflow workload and communication load of past emergencies.

For the first application area, the AWS can make explicit the consequences that timing, information sharing and workload can have in emergency response procedures. Secondly, the workflow resulting from combinations of active protocols can be explored with the AWS thus creating lessons learned concerning protocol combinations without having to conduct costly multidisciplinary exercises.

For the second application area (evaluation of past emergencies), the AWS is able to provide additional information about the overall workload and communication load of the evaluated emergency response. In a similar manner as was demonstrated with the modelling of the Hercules plane crash (Chapter 3) and the proof of concept scenario (Chapter 8) the tasks and or the information flow of an emergency response can be "rebuild" in the AWS. Based on the activities and information flow, the AWS can approximate the workload and communication load of the emergency responders. Furthermore, by altering the original information flow or adjusting other values, such as the location of certain emergency responders, an alternative scenario can emerge that is able to give insight into what would have happened under different circumstances, giving simulated insight into the "What if.. " question.

Finally, the application of the AWS in the TAID system is to provide information about the workflow, workload, communication load and location of the emergency responders as a function of the information that is exchanged, and, secondly, provide a test bed for the ID to assess the impact of enhanced information distribution on the emergency response. Since the proof of concept simulation is able to structure the workflow based on the information that is exchanged and the processing capacity of the emergency responders, it was concluded in Chapter 8 that the AWS is able to function as a test bed for the ID to assess the impact of enhanced information distribution on the workflow, workload and communication load of the emergency responders. A detailed description of the integration of the ID in the AWS to form the TAID system will be presented in the forthcoming dissertation of Niels Netten (University of Amsterdam).

Future research and development

In this dissertation, the development and initial testing of the AWS was presented. Using a proof of concept approach, the feasibility of the AWS to provide information about the current workflow, workload and communication was demonstrated.

However, as was indicated in Chapter 8, it should be noted that the task scenarios that were used were fairly basic, not entirely doing justice to the complexity of the tasks that are performed in practice. Given the limited number of tasks, the proportion between the decrease of processing capacity due to handling messages and the decrease of the processing capacity due to execution of activities could not be assessed. As was indicated by the experts in the field of emergency response, both messages handling and executing activities should determine about the same proportion of the experienced load during the emergency response. However, since messages handling activities and normal tasks differ both in frequency and duration, the total decrease of the processing capacity could get out of balance, overvaluing the one over the other. More research is needed to find out if the decrease of the processing capacity gets out of balance, and if this holds true, how this can be adjusted in the model. This can be achieved by modelling an entire emergency response with the AWS.

Related to this problem is that, due to the small number of tasks that were included in the proof of concept, it could not be concluded how the AWS will behave when provided with more complex models. In the AWS the starting value of the processing capacity of all agents is set at 960, more complex models however have to show if this is a plausible starting value.

Throughout this dissertation the Brahms modelling and simulation environment has proven to be well suited to be used to model and simulate adaptive emergency response using the

AWS. During the development process of the AWS, three limitations of the Brahms modelling and simulation environment surfaced that changed the initially planned implementation. The first Brahms limitation regarded modelling of movement. When a person is moving from area A to area B *directly,* in the simulation the person first is located in area A and, depending on how fast the person is moving, is next located in area B. Unfortunately, when a person moves from area A to area B *through* area X, the simulation shows the same behaviour; first locating the person in area A, then after the duration of the move activity placing the person in area B. In emergency response, moving entail getting information about the locations one is passing through and moving to. In the AWS, this was resolved by pre specifying the movement paths that are used in the simulation, breaking down the original movement to moving from area A to area X and moving from area X to area B. Specifying this for the nine locations in the proof of concept simulation meant adding 11513 lines of code making 537 movements that were embedded in 72 composite move activities. To bring things into perspective, this dissertation has about 8500 lines of text. The movement activities thus need a considerable amount of additional code to model movement for a fairly simple scenario. Additional research is needed whether this will not negatively influence the AWS when more complex scenario's are modelled.

The second Brahms limitation resides in the information providing function of the geography. In the AWS, each area was equipped with an object that deals with the adaptive attributes of the geography, that has the same attributes of the area it resides in, but, in contrast to the area concept, it can engage in state changes. A parallel structure thus is implemented in the form of the area objects to compensate for the inability of the area concepts to engage in state changes. While this works well for smaller models, larger models are most likely to experience a high payload from this parallel structure and, secondly, having parallel structures complicates the modelling process, making it more error prone.

The final Brahms limitation concerns another parallel structure that was built to handle abstract agent references, such as "the police", when referring to someone in the police department. At the time the AWS was developed, the Brahms modelling and simulation environment can only refer to concrete agents. Much communication during emergency response, however, does not refer to concrete agents, since this high level of detail is not needed or is not functional; for example, if it only is necessary to know that someone belongs to a certain organisation or has a certain role.

To Summarize

To summarize, the research presented in this dissertation, developing and testing an adaptive workflow simulator to simulate emergency response workflow as a function of the information that is exchanged and from this derive information concerning the workload and communication load of the emergency responders, takes a promising step forward in making explicit the multidisciplinary information flow, multidisciplinary workflow, emergency responder workload and communication load during the entire emergency response.

The development of the AWS that is described in this dissertation, contributes to different fields of research as a consequence of the application of theoretical models to determine workload and communication load; the development of an enriched workflow representation; the construction of the emergency response template; and the methodological and system development considerations.

The development of the AWS integrated knowledge and theoretical models from several fields of research and applied them in a practical context. Application in the AWS meant operationalization and quantification of these models to the level of detail at which one can use them in a formula or express them in a Brahms model. By operationalizing theoretical models, the development of the AWS contributed to the fields of workload research (by operationalizing task load workload, and processing capacity), information load research (by operationalizing message load and communication load), communication and interaction research (by operationalizing the information element building on the Bales interaction theory).

Given that the AWS is a simulation in which initial values can be adjusted while others remain constant, it also can be used for theory building and conducting experiments that cannot be done in practice. This was demonstrated on several occasions in this dissertation by varying the level of role experience, while the information flow remains the same. When this is extrapolated one can assess the influence of role experience on emergency response outcome. Similar experiments can be imagined for the other factors in the AWS, such as the available processing capacity, workload, communication load and workflow.

The AWS, secondly, implemented an enriched workflow representation that uses information as its trigger. By making the workflow a function of the information that is exchanged the AWS takes a different approach when compared with traditional workflow and process management systems. The discourse taken by the AWS provides it with the flexibility needed to represent adaptive complex (improvised) workflows. The development of the AWS presented in this dissertation thus contributes to the field of workflow research.

The construction of a hierarchal structure of the elements that make up emergency response finally contributes to the field of emergency response research. Using empirical data, this template was constructed, identifying the components that are generic to most emergency responses.

Also benefiting the field of emergency response research, the methodological and system development considerations presented in this dissertation provide an overview of the elements that have to be taken into account when developing a system for emergency response.

Thus, besides having its value in the TAID project as a test bed for the ID and its potential practical value for evaluation of protocols and past emergencies the development of the adaptive workflow simulator presented in this dissertation contributed to several fields of research, and has the potential to function as an experimental environment for theory building and theoretical experiments that cannot be easily conducted in practice

References

Abramson, M., Chao, W., Macker, J., & Mittu, R. (2008). *Coordination in Disaster Management and Response: A Unified Approach.* USA, Washington DC, Naval Research Laboratory.

ACIR (Adviescommissie Coördinatie ICT Rampenbestrijding) (2005). *De vrijblijvendheid voorbij: Op naar een effectieve multidisciplinaire informatievoorziening bij grootschalig gezamenlijk optreden in onze gedecentraliseerde eenheidsstaat.* Den Haag: Author.

Acquisti, A., Sierhuis, M., Clancey, W.J., Bradshaw, J.M. (2002). Agent based modelling of collaboration and work practices onboard the international space station. In *11th Computer generated forces and behaviour representation conference*, Orlando FL, USA, pp. 181-188.

Alavizadeh, F., Moshiri, B., & Lucas, C. (2008). *Application of Naturalistic Decision Making to Emergency Evacuation Simulations.* In Darzentas, J., Vouros, G.A., Vosinikis, S., Arnellos, A. (Eds). Proceedings of the 5th Hellenic conference on Artificial Intelligence: Theories, Models and Applications. Syros, Greece.

Auf der Heide, E. (1989). *Disaster Response: Principles of Preparation and Coordination.* St. Louis: Mosby.

Auf der Heide, E. (2006). The Importance of Evidence-Based Disaster Planning. *Annals of Emergency Medicine*, 47(1), 34-49.

Bales, R. F. (1970). *Personality and interpersonal behavior.* New York: Holt, Rinehart, and Winston.

Becker, A.B., Warm, J.S., Dember, W.N., & Hancock, P.A. (1995). Effects of jet engine noise and performance feedback on perceived workload in a monitoring task. *International Journal of Aviation Psychology ,* Vol. 5, Issue 1, Pages 49-62.

Berfield, A., Chrysanthis, P. K., & Labrinidis, A. (2004). Automated service integration for crisis management. *Paper presented at the Databases in Virtual Organizations (DIVO 2004)*, Paris.

Boin, R.A., Kofman-Bos, C.J., & Overdijk, W.I.E. (2004). Crisis simulations: Exploring tomorrow's vulnerabilities and threats. *Simulation and gaming*, 35(3), 378-393.

Brenner, T., & Werker, C. (2007). A taxonomy of inference in simulation models. *Computational Economics*, 30, 227-244.

Bruinsma, G. (2005). *De rol van communicatie, informatie en workflow in het hulpverleningsproces tijdens (tunnel)rampen*. Internal publication: Enschede: University of Twente.

Calderwood, B., Klein, G.A. & Crandall, B.W. (1988). Time pressure, skill, and move quality in chess. *American Journal of Psychology, 101*, 481-493.

Campbell, D. J. (1988). Task complexity: A review and analysis. *Academy of Management Review*. Vol 13(1), 40-52.

Capgemini, TNO, Berenschot (2008). *Evaluatie oefening Voyager: Een systeemevaluatie van kritische processen bij crisisbeheersing*. Utrecht: Capgemini.

Carver, L., & Turoff, M. (2007). Human-Computer Interaction: The Human and Computer as a Team in Emergency Management Information Systems, *Communications of the ACM*, 50, 3, 33-38.

Chen, X., Meaker, J.W., Zhan, F.B. (2006). Agent-Based Modeling and Analysis of Hurricane Evacuation Procedures for the Florida Keys. *Natural Hazards 38*(3): 321-338.

Clancey, W.J., P. Sachs, P., Sierhuis, M., & Hoof, R. Van. (1998). Brahms: Simulating practice for work systems design, *International Journal on Human-Computer Studies*, vol. 49, pp. 831-865.

Commissie onderzoek vuurwerkramp (2001). *De vuurwerkramp: eindrapport*. Enschede/Den Haag, The Netherlands: Author.

Corbacioglu, S., & Kapucu, N. (2006). Organisational learning and self adaptation in dynamic disaster environments. *Disasters*, 30(2), 212-233.

Crystal, A. & Ellington, B. (2004). Task analysis and human-computer interaction: approaches, techniques, and levels of analysis. *Paper presented at the 2004 Americas Conference on Information Systems (AMCIS '04)*. New York, NY, August 5 - 8.

de Groeve, T., Kugler, Z., & Brakenridge, G.R. (2007). Near real time flood alerting for the global disaster alert and coordination system. In Van de Walle, B., Burghardt, P. & Nieuwenhuis, C. (Eds), *Proceedings of the 4th International ISCRAM Conference*, Delft, The Netherlands.

de Muralt, R. (2007). *Procesmanagement bij de brandweer* [Process management within the fire department]. Den Haag, The Netherlands: SDU.

de Zwart, B.C., Frings-Dresen, M.H., & van Dijk, F.J. (1995). Physical workload and the aging worker: a review of the literature. *International Archives of Occupational and Environmental Health,* 68(1):1–12.

Divitini, M., & Simone, C. (2000). Supporting different dimensions of adaptability in workflow modelling. *Computer Supported Cooperative Work*, 9, 365–397.

Dominguez, S. (2003). Visual Anomaly Detection via Soft Computing: A Prototype Application at NASA. *Mathware & Soft Computing*, 10, 131-139.

Dutch ministry of the interior and kingdom relations (2003). Handboek voorbereiding rampenbestrijding: Overzicht docmentatie rampenbestrijding. The Hague, the Netherlands: Author.

French Minister of the Interior Equipment Transportation and Housing (1999). *Task Force for Technical Investigation of The 24 March 1999 Fire in The Mont Blanc Vehicular Tunnel - Report of 30 June 1999* (English translation).

Georgiadou, P.S., Papazoglou, I.A., Kiranoudis, C.T., & Markatos, N.C. (2007). Modeling emergency evacuation for major hazard industrial sites. *Reliability Engineering & Systems Safety*, Volume 92, Issue 10, Pages 1388-1402.

Gilbert, N., & Troitzsch, K.G. (2005). *Simulation for the Social Scientist*. Milton Keynes: Open University Press.

Griefahn, B., Künemund, C., & Bröde, P. (2003). Evaluation of performance and load in simulated rescue tasks for a novel design SCBA: effect of weight, volume and weight distribution. *Applied Ergonomics* 34, 157-165.

Grootjen, M., Neerincx, M.A., Weert, J.C.M. van (2006). Task-based interpretation of operator state information for adaptive support, *Foundations of Augmented Cognition, 2nd Edition*, ISBN 10: 0-9789812-0-0, no. Strategic Analysis, pp. 236-242, Arlington, Virginia, 2006.

Guidotti, T.L.(1992). Human factors in fire fighting: ergonomic-, cardiopulmonary-, and psychogenic stress-related issues. *International* Archives of *Occupational* and *Environmental Health* (1992), 64:1-12

Hart, S.G., & Staveland, L.E. (1988). Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In Hancock, P.A., & Meshkati, N. (eds) *Human Mental Workload*, pp. 239-250, Amsterdam: North Holland Press.

Hatum, A., & Pettigrew, A.M. (2005). Determinants of organizational flexibility: a study in an emerging economy. *British journal of management*, 16.

Hays, R.T., & Singer, M.J. (1989). *Simulation fidelity in training systems design: bridging the gap between reality and training.* New York: Springer-Verslag.

Henein, C.M., & White, T. (2005). Agent-Based Modelling of Forces in Crowds. MAMABS05. *Lecture Notes in Computer Science v. 3415. Springer,* 173-184.

Hilburn, B., & Jorna, P. (2001). Workload and air traffic control. In: P.A. Hancock and P.A. Desmond, Editors, *Stress, Workload, and Fatigue. Human Factors in Transportation*, Lawrence Erlbaum Associates, Mahwah, NJ (2001), pp. 384–394.

Iannoni, A.P. & Morabito, R. (2006). A discrete simulation analysis of a logistics supply system, *Transportation Research E42 (3)* (2006), pp. 191-210.

Institute for Safety Security and Crisis Management (COT) (2005). *Bordering on reality: Report of findings on Bonfire crisis management simulation*. Den Haag, The Netherlands: Author.

International Federation of Red Cross and Red Crescent Societies (2005). *World disasters report: Focus on information in disasters*. London: Eurospan.

Joorse, R., Slofstra, P., & van Dijke, A. (1995). Voorschrift bepakking (tank)autospuit [Packaging prescriptions pumper / pump water tender]. Arnhem, The Netherlands: Nederlandse brandweer federatie [Dutch federation of fire services].

Kahneman, D. (1973). *Attention and Effort*. Englewood Cliffs, NJ: Prentice Hall.

Kang, K. (2007). Application of Code Approach for Emergency Evacuation in a Rail Station. *Fire Technology*, *Springer Netherlands*, 43, 4, 331-346.

Keramitsoglou, I., Kiranoudis, C., Sarimveis, H., & N. Sifakis, N. (2004). "A Multidisciplinary decision support system for forest fire crisis management", *Environmental Management*, 33 ,2 , 212-225.

Killian, L. M. (2002). An Introduction To Methodological Problems Of Field Studies In Disasters. In R. A. Stallings (Ed.), Methods Of Disaster Research (pp. 49-93). *Internationals Research Committee on Disasters.* California, USA.

Klein, M., Dellarocas, C., & Bernstein, A. (2000). Introduction to the special issue on adaptive workflow systems. *Computer Supported Cooperative Work*, 9, 265–267.

Kramer, P.J., Bolcer, G.A., Taylor, R.N., Hitomi, A.S., & Bergman, M. (2000). *Techniques for supporting Dynamic and adaptive workflow. Computer supported cooperative work*, 9, 269-292.

Lemon, P.W.R., & Hermiston, R.T. (1977). The human energy cost of fire fighting. *Journal of* Occupational *and Environmental Medicine, 1*9, 558–562.

Levin, S., France, D.J., Hemphill, R., Jones, I., Chen, K.Y., Rickard, D., Makowski, R. & Aronsky, D. (2006). Tracking workload in the emergency department. *Human Factors.* Fall, 48(3), 526-39.

Li, H., Tang, W., & Simpson, D. (2004). Behaviour Based Motion Simulation for Fire Evacuation Procedures, *Theory and Practice of Computer Graphics 2004* , pp.112-118.

Llavador, M., Letelier, P., Penadés, M. C., Canós, J.H., Borges, M., & Solís, C. (2006). Precise yet flexible specification of emergency resolution procedures. In Van de Walle, B., & Turoff, M. (Eds), *Proceedings of the 3rd International ISCRAM Conference*, 110-120. Newark, NJ (USA).

Lloyd, A., & Somerville, M. (2006). Working information. *Journal of Workplace Learning*, 18(3), 186-198.

Loft, S., Sanderson, P., Neal, A. & Mooij, M. (2007). Modeling and predicting mental workload in en route air traffic control: critical review and broader implications. *Human Factors*. Jun 2007, Vol. 49, Issue 3, Pages 376-99.

Losh, E. (2007). The Birth of the Virtual Clinic: Game Spaces in "The Virtual Practicum" and the Virtual Terrorism Response Academy. In Van de Walle, B., Burghardt, P. & Nieuwenhuis, C. (Eds), *Proceedings of the 4th International ISCRAM Conference*, Delft, The Netherlands.

Luczak, H. (1971). The Use of Simulators For Testing Individual Mental Working Capacity. *Ergonomics*, 1971, vol. 14, no. 5, 651-660.

Manning, C.A., Mills, S.H., Fox, C.M., Pfeiderer, E.M., & Mogilka, H.J. (2002). *Using air traffic control taskload measures and communication events to predict subjective workload*. (Report No. DOT/FAA/AM-02/4). Washington DC: Office of Aerospace Medicine Federal Aviation Administration.

Manoj, B.S., & Baker, H.A. (2007). Communication challenges in emergency response. *Communications of the ACM*, Volume 50, Number 3, 51-53.

Mendonça, D., Jefferson, T. & Harrald, J.R. (2007). Collaborative adhocracies and mix-and-match technologies in emergency management. *Communications of the ACM* 50(3): 44-49 (2007).

Mintzberg, H. (1983). Structure in fives: *Designing effective organizations. Englewood Cliffs.* N.J.: Patience hall.

National commission on terrorist attacks upon the United States (2004). *The 9/11 commission report*. Washington, DC: U.S. Government Printing Office.

Netten, N., Bruinsma, G., van Someren, M., & de Hoog, R. (2006). Task-adaptive information distribution for dynamic collaborative emergency response. *International Journal of Intelligent Control and Systems*, 11(4), 238-247.

Newell, G.S., & Mansfield, N.J. (2007). *Influence of twisted posture and multi-axis vibration on seat-to-head transmissibility*. In Bust, P.B. (Ed.) Contemporary Ergonomics 2007: Proceedings of the International Conference on Contemporary Ergonomics. Notingham: Taylor and Francis Group.

Nibra (2005). *Manschappen 1: Oriëntatie*. Arnhem, The Netherlands: Author.

O'Donnell, R. D., & Eggemeier, F. T. (1986). Workload assessment methodology. In K. R. Boff, L. Kaufman, and J. Thomas (Eds.), *Handbook of Perception and Human Performance: Volume II, Cognitive Processes and Performance.* New York: John Wiley.

Open Geospatial Consortium (2006). *OpenGIS Implementation specification for geographic information – simple feature access – Part 1 common architecture*: www.opengeospatial.org.

Otto, P., & Belardo, S. (2006). A Theoretical Evaluation of Information Processing Resources during Organizational Crisis, *Proceedings of the 3rd International Conference on Information Systems for Crisis Management, ISCRAM 2006*, Newark, New Jersey (USA).

Parasuraman, R., & Hancock, P.A. (2001). Adaptive control of mental workload, in P.A. Honcock and P.A. Desmond (eds), *Stress, workload, and fatique* (Mahwah, NJ: Erlbaum), 305-320.

Patten, C.J.D., Kircher, A., Östlund, J., Lena Nilsson, L., & Svenson, O. (2006). Driver experience and cognitive workload in different traffic environments. *Accident Analysis & Prevention*, Volume 38, Issue 5, 887-894.

Projectgroep Modificatie Bepakkingslijsten. (2008). *Branchevoorschrift Standaardbepakking Tankautospuit en Hulpverleningsvoertuig 2008*. The Hague, The Netherlands: Landelijke Faciliteit Rampenbestrijding.

Raad van RGF'en (2006). De Nederlandse slachtofferregistratiekaart. Utrecht, the Netherlands: Raad van RGF'en.

Rasmussen, J. (1983). Skills, rules, and knowledge: Signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(3), 257-266.

Reid, G. B., & Nygren, T. E. (1988). The subjective workload assessment technique: a scaling procedure for measuring mental workload. In P. A. Hancock & N. Meshkati (Eds.), *Human Mental Workload*: 185-218. North-Holland: Elsevier Science Publishers.

Rich, E. (2006). Modeling Risk Dynamics in E-Operations Transitions, *Proceedings of the 3rd International Conference on Information Systems for Crisis Management, ISCRAM 2006*, Newark, New Jersey (USA).

Sabastiani, F. (2002). Machine learning in automated text categorization. *ACM Computer Surveys*, 34(1), 1-47.

SAVE & Adviesbureau van Dijke (2000). *Leidraad maatramp*. Den Haag: The Dutch ministry of the interior and kingdom relations.

Shah, S.A.A., Kim, H., Baek, S., Chang, H., & Ahn, B.H. (2008). System architecture of a decision support system for freeway incident management in Republic of Korea. *Transportation Research Part A*, 42, 799-810.

Sharma, S., Singh, H., & Prakash, A. (2008). Multi-agent modeling and simulation of human behavior in aircraft evacuations, *IEEE Transactions on Aerospace and Electronic Systems*, Vol.44, No.4, October 2008.

Shea, G.F. (1981). *The new employee: Developing a productive human resource*. London: Addison-Wesley.

Shiffrin, R. M. & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual Learning, automatic attending, and a general theory. *Psychological Review*, 84, 127-190.

Sierhuis, M. (2000). *Modeling and Simulation of Work Practices on the Moon*, Paper presented at the Computational Analysis of Social and Organizational Systems 2000, Carnegie Mellon University, Pittsburgh, PA.

Sierhuis, M., Diegelman, T.E., Seah, C., Shalin, V., Clancey, W.J., Selvin, A.M. (2007). Agent based simulation of shuttle mission operations. In *Agent directed simulation 2007 part of the 2007 spring simulation multiconference*, pp 53-60. Norfolk VA, USA, The society for modeling and and simulation international (ACM/SIGSIM).

Siliunas, M. (1991). Oxygen consumption during fire suppression: error of heart rate estimation. *Ergonomics*, 34(12), 1469-1474.

Sothmann, M., Saupe, K., Raven, P., Davis, P., Dotson, C., & Landy, F. (1991). Oxigen consumption during fire suppression: error of heart rate estimation. *Ergonomics*, 34, 1469-1474.

Spilker, B.C. (1995). The Effects of Time Pressure and Knowledge on Key Word Selection Behavior in Tax Research. *The Accounting Review*, 70, 1, 49-70.

Staal, M.A. (2004). *Stress, Cognition, and Human Performance: A Literature Review and Conceptual Framework.* NASA/TM-2004-212824. Ames Research Center, Moffett Field, California.

Storms, P.P.A. (2004). Combined Systems, A System of Systems Architecture. *Proceedings of ISCRAM 2004 (Eds. B. Carle and B. Van de Walle)*, 139-144

Sulistio, A., Yeo, C.S., & Buyya, R. (2004). A taxonomy of computer-based simulations and its mapping to parallel and distributed system simulation tools. *Software Practice and Experience*, 34, 653-673.

Turoff, M., Chummer, M., van de Walle, B., & Yoa, X. (2004). The design of a dynamic emergency response management information systems (DERMIS). *Annual Review of Communications*, 58, 637-661.

van Aart, C.J., Wielinga, B., & Schreiber, G. (2004). Organizational building blocks for design of distributed intelligent system. *International Journal of Human-Computer Studies*, 61, 567-599.

van den Eede, G., Muhren W., Smals R., & van de Walle B. (2006). Incident Response Information Systems Capability: the role of the DERMIS design principles. In *Proceedings of the 3rd International Conference on Information Systems for Crisis Response and Management (ISCRAM 2006)*, Newark NJ, USA, pp. 251–261.

van der Aalst, W. M. P., ter Hofstede, A. H. M., Kiepuszewski, B., & Barros, A. P. (2003). Workflow patterns. *Distributed and Parallel Databases*, 14(1), 5-51.

Wang, J., Rosca, D., Tepfenhart, W., & Milewski, A. (2006). Incident command system workflow modeling and analysis: A case study. In Van de Walle, B. & Turoff, M. (Eds), *Proceedings of the 3rd International ISCRAM Conference,* 127-136. Newark, NJ (USA).

Wickens, C. D. & Hollands, J. G. (1999). *Engineering Psychology and Human Performance (3rd ed.)*. Upper Saddle River, New Jersey: Prentice Hall.

Yamamoto, K., Kokubo, S., Yamashita, H., & Nishinari, K. (2008). *Simulation of Fire Evacuation by Real-Coded Cellular Automata (RCA)*. Dep. Mechanical Science and Engineering, Nagoya University, Japan.

Samenvatting

Ten tijde van incidenten en rampen opereert de hulpverleningsorganisatie in een snel veranderende omgeving en is genoodzaakt haar samenstelling aan te passen aan de eisen die het incident of de ramp aan haar stelt. Een adhocratische organisatie zoals de hulpverleningsorganisatie besteedt als gevolg relatief veel tijd en moeite aan communicatie om onder andere de multidisciplinaire activiteiten te stroomlijnen en een actueel en correct beeld te verkrijgen van haar samenstelling en de huidige staat van het incident. Het delen van informatie speelt hierbij een belangrijke rol; het beïnvloedt het verloop van de repressie, de volgorde waarin werkzaamheden worden uitgevoerd (workflow) en de mate waarin hulpverleners worden belast door het uitvoeren van werkzaamheden en de stroom van informatie.

Uit evaluaties van rampen en grootschalige incidenten is herhaaldelijk naar voren gekomen dat het ontbreken van informatie als gevolg van gebrekkige communicatie het verloop, en zeer aannemelijk ook de uitkomst, van het hulpverleningsproces negatief beïnvloedt. Vanuit de veronderstelling dat informatie een sturende rol heeft in het uitvoeren van activiteiten van hulpverleners, beschrijft het proefschrift de ontwikkeling en een initiële test van een simulatiemodel voor het nabootsen en evalueren van de gevolgen die het al dan niet hebben van informatie kan hebben op de werkzaamheden van personen die betrokken zijn bij de hulpverlening tijdens rampen. De simulatie kan zo een belangrijke rol spelen in het beantwoorden van "Wat als…. " vragen met de mogelijkheid om zonder uitgebreide oefeningen "*lessons learned*" te creëren. Echter, naast het expliciteren van de relatie tussen informatie en de werkprocessen van de individuele hulpverleners wordt in het proefschrift op basis van de gemodelleerde werkzaamheden en informatiestroom eveneens een schatting gemaakt van de (fysieke en mentale) belasting en de additionele belasting veroorzaakt door de hoeveelheid informatie die de individuele hulpverleners gedurende de (gesimuleerde) hulpverlening moet verwerken.

Als onderdeel van het bredere TAID project, waar de ontwikkeling van het in dit proefschrift beschreven simulatiemodel een deelproject is, verschaft het simulatiemodel extra contextuele informatie met betrekking tot de werkzaamheden, werkbelasting en belasting veroorzaakt door communicatie van de individuele hulpverleners aan een geautomatiseerde informatieverdeler (ID). De ID gebruikt deze extra contextuele informatie om de verspreiding van informatie te verbeteren voor hulpverleningssituaties. De in dit proefschrift ontwikkelde simulatie zal voor de ID tevens dienen als testomgeving om de gevolgen van de verbeterde informatieverdeling op de werkprocessen, werkbelasting en belasting veroorzaakt door de hoeveelheid gedeelde informatie expliciet te maken.

De onderzoeksvraag die in dit proefschrift aan de orde komt is "Hoe kunnen we als functie van de informatie die wordt uitgewisseld tussen hulpverleners een generiek simulatiemodel ontwikkelen en verankeren in de hulpverleningspraktijk, dat op adaptieve wijze de activiteiten van hulpverleners kan simuleren en informatie kan verschaffen over de volgorde waarin de activiteiten plaatsvinden, als mede over de werkbelasting en communicatiebelasting van individuele hulpverleners?"

Voor de ontwikkeling van het simulatiemodel is gekozen voor het gebruik van een bestaande simulatieomgeving ontwikkeld door NASA (Brahms). Deze omgeving biedt de mogelijkheid om op een functioneel niveau adaptieve werkmodellen te creëren en te simuleren zonder dat dit model de precieze fysieke werkelijkheid hoeft te representeren, zoals bijvoorbeeld wel het geval is bij grafische (3d) simulatie pakketen. Hierdoor is de mogelijkheid gecreëerd om het simulatiemodel relatief eenvoudig en goedkoop toe te passen om de informatiestroom en werkprocessen te simuleren en evalueren.

De toepassingsgerichtheid wordt verder benadrukt door het model te baseren op een *algemeen toepasbare structuur* van elementen die tijdens elke hulpverleningssituatie terugkomen: de organisaties (zoals brandweer, politie, GHOR), rollen (zoals OvD's ROT en GBT leden), taken (zoals blussen, vervoer slachtoffers, afzetten brongebied), objecten (tankautospuit, gewondennest, CTPI haakarmbak) en locatietypen (zoals steden, straten, gebouwen). Deze algemene structuur vormt een herbruikbare basis voor het modelleren van verschillende incidentscenario's.

Een belangrijke rol in het proefschrift is weggelegd voor het verankeren van de simulatie in de hulpverleningpraktijk. Geregistreerde communicatie van de hulpverleners tijdens twee multidisciplinaire oefeningen in combinatie met cursusmateriaal afkomstig van het NIFV en kerndocumenten op het gebied van hulpverleningsprocessen, hebben geleid tot een solide verankering van de algemeen toepasbare structuur in de praktijk. Aanvullend en ter toetsing van de achterliggende veronderstellingen over de werkbelasting- en communicatiebelasting modellen, is een groep van 26 experts op het gebied van crisismanagement gevraagd hun mening te geven over relaties tussen onder andere ervaring en belasting door informatie en (fysieke en mentale) arbeid; oorzaken van werkbelasting in alle lagen van de hulpverleningsorganisatie (van operationeel tot bestuurlijk); en de snelheid van herstel van fysieke en mentale werkzaamheden.

De resulterende algemeen toepasbare structuur en de modellen met betrekking tot werkstructurering op basis van informatie, werkbelasting en communicatiebelasting zijn vervolgens getest door het nabootsen van een deel van een multidisciplinaire oefening. Deze maakt gebruik van een gedetailleerde registratie van de communicatie van een andere multidisciplinaire oefening.

Uit de resultaten van deze nabootsing bleek dat ondanks de complexiteit en dynamiek van de rampen en incidenten, relatief veel aspecten constant blijven. De algemeen toepasbare structuur bleek voor het gemodelleerde testscenario voldoende detail te bezitten, waardoor de uiteindelijke nabootsing efficiënt verliep. Ten tweede bleek dat de werkstructurering op basis van informatie de potentie heeft om ook complexere situaties aan te kunnen om daadwerkelijk "Wat als .. " vragen te beantwoorden, bijvoorbeeld voor de evaluatie van hulpverlening tijdens incidenten. Als laatste bleken de modellen van werk- en informatie belasting te leiden tot geloofwaardige effecten op het vermogen van hulpverleners om verschillende taken uit te voeren. Meer onderzoek en toepassingen zouden echter moeten uitwijzen in hoevere de validiteit en generaliseerbaarheid van deze modellen alsmede het werkstructurering model is gegarandeerd.

Annex: Expert Questionnaire

Achtergrondinformatie

1.	Bij welke organisatie bent u werkzaam? (omcirkel het juiste antwoord)		
	a.	Brandweer	
	b.	Politie	
	c.	Geneeskundig	
	d.	Anders, namelijk:	
		1. …………………………………………….	
2.	Wat is/zijn de functie(s) die u binnen deze organisatie uitoefent?		
		1. …………………………………………….	
		2. …………………………………………….	
3.	Wat is uw werkervaring in jaren op:		
			Jaren
		Operationeel niveau	……….Jaar
		Tactisch niveau	……….Jaar
		Strategisch niveau	……….Jaar
		Bestuurlijk niveau	……….Jaar

Oorzaken werkbelasting

		Vragen 4 t/m 7 hebben betrekking op de factoren die de voornaamste oorzaken zijn van werkbelasting op de verschillende organisatieniveaus die actief zijn tijdens incidentbestrijding. Het is belangrijk dat u voor alle niveaus een schatting geeft van de mate waarin de genoemde factoren bepalend zijn voor de werkbelasting.	

4.	In welke mate wordt de werkbelasting tijdens incidentbestrijding, binnen de eigen organisatie op ***operationeel*** niveau, bepaald door: (verdeel 100% over de categorieën)		
			Percentage
	a.	De mentale en fysieke aspecten van de werkzaamheden zelf (taakbelasting)	……….%
	b.	De hoeveelheid uitgewisselde informatie (informatiebelasting)	……….%
	c.	De Tijdsdruk	……….%
	d.	Het ontbreken van de mogelijkheid te herstellen van werkzaamheden	……….%
	e.	De hoeveelheid werkzaamheden die tegelijkertijd moeten worden uitgevoerd	……….%
	f.	Anders, namelijk:	
		1. …………………………………………….	……….%
		2. …………………………………………….	……….%
			+ --------
			100 %

5.	In welke mate wordt de werkbelasting tijdens incidentbestrijding, binnen de eigen organisatie op ***strategisch*** niveau, bepaald door: (verdeel 100% over de categorieën)		
			Percentage
	a.	De mentale en fysieke aspecten van de werkzaamheden zelf (taakbelasting)	……….%
	b.	De hoeveelheid uitgewisselde informatie (informatiebelasting)	……….%
	c.	De tijdsdruk.	……….%
	d.	Het ontbreken van de mogelijkheid te herstellen van werkzaamheden	……….%
	e.	De hoeveelheid werkzaamheden die tegelijkertijd moeten worden uitgevoerd	……….%
	f.	Anders, namelijk:	
		1. …………………………………………….	……….%
		2. …………………………………………….	……….%
			+ --------
			100 %

6.	In welke mate wordt de werkbelasting tijdens incidentbestrijding, binnen de eigen organisatie op ***tactisch*** niveau, bepaald door: (verdeel 100% over de categorieën)		
			Percentage
	a.	De mentale en fysieke aspecten van de werkzaamheden zelf (taakbelasting)	……….%
	b.	De hoeveelheid uitgewisselde informatie (informatiebelasting)	……….%
	c.	De tijdsdruk.	……….%
	d.	Het ontbreken van de mogelijkheid te herstellen van werkzaamheden	……….%
	e.	De hoeveelheid werkzaamheden die tegelijkertijd moeten worden uitgevoerd	……….%
	f.	Anders, namelijk:	
		1. …………………………………………….	……….%
		2. …………………………………………….	……….%
			+ --------
			100 %

7.	In welke mate wordt de werkbelasting tijdens incidentbestrijding op ***bestuurlijk*** niveau, bepaald door: (verdeel 100% over de categorieën)		
			Percentage
	a.	De mentale en fysieke aspecten van de werkzaamheden zelf (taakbelasting)	……….%
	b.	De hoeveelheid uitgewisselde informatie (informatiebelasting)	……….%
	c.	De tijdsdruk.	……….%
	d.	Het ontbreken van de mogelijkheid te herstellen van werkzaamheden	……….%
	e.	De hoeveelheid werkzaamheden die tegelijkertijd moeten worden uitgevoerd	……….%
	f.	Anders, namelijk:	
		1. …………………………………………….	……….%
		2. …………………………………………….	……….%
			+ --------
			100 %

Ervaring en informatieverwerking

Vragen 8 t/m 10 hebben betrekking op de rol die een toename van het aantal jaren werkervaring heeft op de verwerking van en belasting door informatie tijdens de incidentbestrijding.

8. Geef voor de volgende stellingen over de rol van ervaring bij informatieverwerking aan in hoeverre u het er mee eens of oneens bent

a. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, is deze beter in staat de relevantie van de informatie te beoordelen.

helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○

b. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, is deze in staat *sneller* de relevantie van informatie te beoordelen.

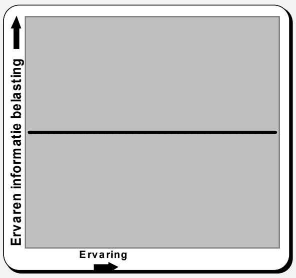
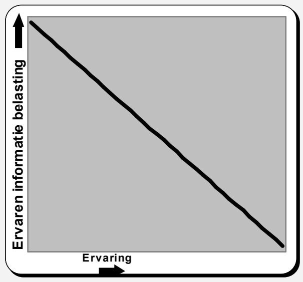
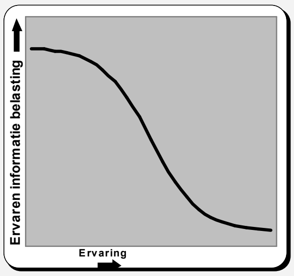
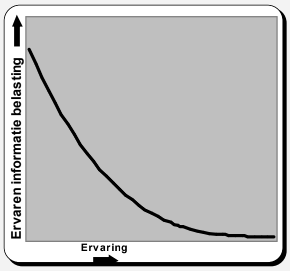
helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○

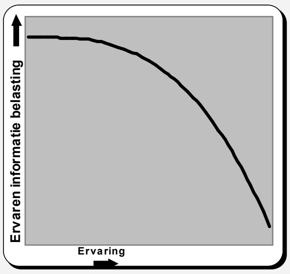
c. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, zal dezelfde hoeveelheid informatie minder belastend zijn voor hem/haar dan voor iemand met minder ervaring (informatiebelasting neemt af met het aantal jaren ervaring).

helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○
Ga verder naar vraag 10		**Ga verder naar vraag 9**		

9. Geef aan welke van de onderstaande vijf grafieken volgens u het best de relatie weergeeft tussen ervaring (horizontale as) en de ervaren informatiebelasting (verticale as) bij een gelijkblijvende hoeveelheid informatie. De ervaren informatiebelasting is de belasting die een persoon ervaart als gevolg van een bepaalde hoeveelheid informatie die hij/zij krijgt.

"Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleed tijdens de incidentbestrijding, zal dezelfde hoeveelheid informatie minder belastend zijn voor hem/haar dan voor iemand met minder ervaring (informatiebelasting neemt af met het aantal jaren ervaring)."(vraag 8c)

	
○	○
Er is geen invloed van ervaring op de ervaren informatiebelasting. Deze is constant over personen met een verschillende hoeveelheid ervaring.	*Er is een rechtlijnig dalend verband tussen ervaring en ervaren informatiebelasting. Naarmate een persoon meer ervaring heeft neemt de informatiebelasting met een constante hoeveelheid per jaarervaring af.*
	
○	○
Er is een Z-vormig verband tussen ervaring en ervaren informatiebelasting. In de beginjaren van de carrière neemt de ervaren informatiebelasting relatief weinig af. In de periode die volgt neemt deze snel af. Vervolgens stabiliseert de afname, waardoor een extra jaar ervaring aan het eind van deze laatste periode zorgt voor een relatief kleine afname van de ervaren informatiebelasting.	*In de beginjaren van de carrière neemt de ervaren informatiebelasting relatief snel af Vervolgens stabiliseert de afname, waardoor een extra jaar ervaring aan het eind van deze laatste periode zorgt voor een relatief kleine afname van de ervaren informatiebelasting.*

					
			○		
			In de beginjaren van de carrière neemt de ervaren informatiebelasting relatief weinig af. Vervolgens neemt de ervaren informatiebelasting elk jaar met grotere hoeveelheden af.		

10.	Geef voor de volgende stellingen over de rol van ervaring aan in hoeverre u het er mee eens of oneens bent						
	a.	De informatiebelasting op ***tactisch*** niveau neemt toe naarmate het incident een hogere GRIP status heeft.					
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	
	b.	De informatiebelasting op ***strategisch*** niveau neemt toe naarmate het incident een hogere GRIP status heeft.					
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	
	c.	De informatiebelasting op ***tactisch*** niveau neemt toe naarmate er meer hulpverleners bij het incident betrokken zijn.					
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	
	d.	De informatiebelasting op ***strategisch*** niveau neemt toe naarmate er meer hulpverleners bij het incident betrokken zijn.					
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	
	e.	De hulpverleners van de verschillende organisaties die bij de incidentbestrijding betrokken zijn weten welke informatie die zij hebben nuttig is voor de hulpverleners van de andere organisaties.					
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	

Ervaring en taakbelasting

Vraag 11 heeft betrekking op de rol die een toename van het aantal jaren werkervaring heeft op de belasting door deze werkzaamheden tijdens de incidentbestrijding.

Voorbeeldstelling:
Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleed tijdens de incidentbestrijding, zal dezelfde taak minder belastend zijn voor hem/haar dan voor iemand met minder ervaring (taakbelasting neemt af met een toename van het aantal jaren ervaring).

11. Geef voor de volgende stellingen over de rol van ervaring aan in hoeverre u het er mee eens of oneens bent

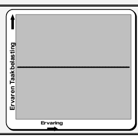
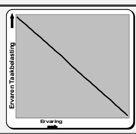
a. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, is de belasting als gevolg van de werkzaamheden (taakbelasting), los van de belasting door communicatie, voor hem/haar minder groot dan bij minder ervaren personen.

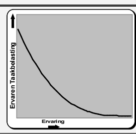
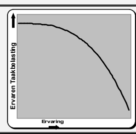
helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○

b. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, is de belasting als gevolg van de mentale werkzaamheden (denken, beslissen, onthouden, zoeken), voor hem/haar minder groot dan bij minder ervaren personen.

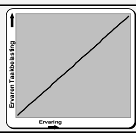
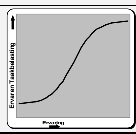
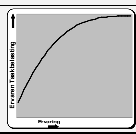
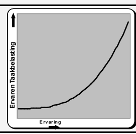
helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○

c. Geef aan welke van de onderstaande negen grafieken volgens u het best de relatie weergeeft tussen ervaring (horizontale as) en de ervaren belasting van de ***mentale*** werkzaamheden (verticale as) bij dezelfde taak.

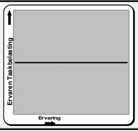
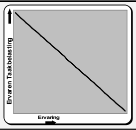
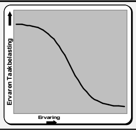
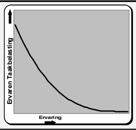
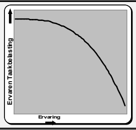
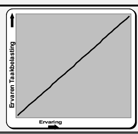
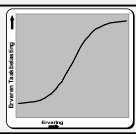
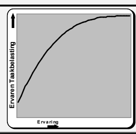
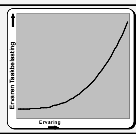
○ ○ ○ ○ ○

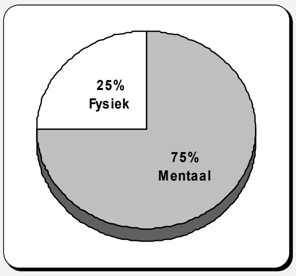





○ ○ ○ ○

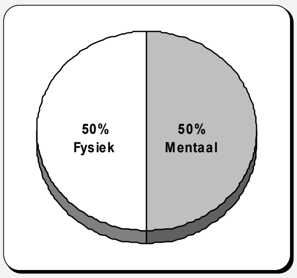
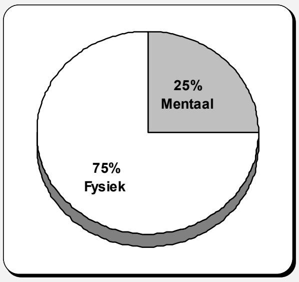
d. Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, is de belasting als gevolg van de fysieke werkzaamheden (bewegen, trekken, duwen) bij dezelfde taak, voor hem/haar minder groot dan bij minder ervaren personen.

helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
○	○	○	○	○

	e.	Geef aan welke van de onderstaande negen grafieken volgens u het best de relatie weergeeft tussen ervaring (horizontale as) en de ervaren belasting van de ***fysieke*** werkzaamheden (verticale as) bij dezelfde taak.					
							
		○	○	○	○	○	
							
			○	○	○	○	
	f.	Bij werkzaamheden die voor ***75% uit mentale aspecten*** bestaan en voor ***25% uit fysieke aspecten*** is, naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, de belasting voor hem/haar minder dan bij minder ervaren personen.					



		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	

	g.	Bij werkzaamheden die voor ***50% uit mentale aspecten*** bestaan en voor ***50% uit fysieke aspecten*** is, naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, de belasting voor hem/haar minder dan bij minder ervaren personen.					
							
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	
	h.	Bij werkzaamheden die voor ***25% uit mentale aspecten*** bestaan en voor ***75% uit fysieke aspecten*** is, naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, de belasting voor hem/haar minder dan bij minder ervaren personen.					
							
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens	
		○	○	○	○	○	

Ervaring en herstel			
		Vraag 12 heeft betrekking op de rol die een toename van het aantal jaren werkervaring heeft op de snelheid waarmee iemand herstelt van werkzaamheden tijdens de incidentbestrijding. Voorbeeldstelling: Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, zal deze persoon sneller herstellen (geen negatieve gevolgen (zoals vermoeidheid) ervaren van de voorgaande taak op de huidige taak) dan een persoon met minder ervaring; de ervaren persoon is sneller weer inzetbaar voor een volgende taak (snelheid van herstel neemt toe met een toename van het aantal jaren ervaring).	

12.	Geef voor de volgende stellingen over de rol van ervaring aan in hoeverre u het er mee eens of oneens bent.					
	a.	Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, verloopt het herstel van de belasting die het gevolg is van ***fysieke werkzaamheden*** (bewegen, trekken, duwen), voor hem/haar sneller dan bij minder ervaren personen.				
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
		○	○	○	○	○
	b.	Naarmate een persoon meer ervaring heeft (in jaren) in de rol die hij/zij bekleedt tijdens de incidentbestrijding, verloopt het herstel van de belasting die het gevolg is van ***mentale werkzaamheden*** (denken, beslissen, onthouden, zoeken), voor hem/haar sneller dan bij minder ervaren personen.				
		helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
		○	○	○	○	○

13.	Door ervaring (in jaren) ontwikkel je handigheden in het werk waardoor informatieverwerking makkelijker, sneller en beter verloopt; taken als minder belastend worden ervaren en herstel van belastende taken sneller verloopt.				
	helemaal mee oneens	oneens	neutraal	eens	helemaal mee eens
	○	○	○	○	○

Einde vragenlijst

Bedankt voor uw medewerking