Research Report #358, ISBN 82-7368-316-8, ISSN 0806-3036
Department of Informatics, University of Oslo
June 2007

Developing Mobile Middleware – An Analysis of Rescue and Emergency Operations

Norun Christine Sanderson, Katrine Stemland Skjelsvik, Ovidiu Valentin Drugan, Matija Pužar, Vera Goebel, Ellen Munthe-Kaas, Thomas Plagemann

Department of Informatics, University of Oslo

{noruns, katrins, ovidiu, matija, goebel, ellenmk, plageman}@ifi.uio.no

Abstract 1
1 Introduction 2
2 Rescue Operation Characteristics and Scenario Examples 4
2.1 Characteristics of Rescue Operation Organisation and Structure 4
2.1.1 Organisations Involved 4
2.1.2 Rescue Operation Organisation 5
2.2 Earthquake Scenario 6
2.3 Railway Accident 8
2.4 Subway Station Accident 11
2.5 Scenario Commonalities and Differences 12
3 Requirements Analysis and Technical Considerations 15
3.1 Organisational Intra- and Inter-operability 15
3.2 Security Aspects 16
3.3 Network Characteristics 17
3.3.1 Devices and Network Topology 17
3.3.2 User Mobility 18
3.4 Communication Issues 19
3.5 Development Constraints 20
3.5.1 Synthetic Mobility Models 20
3.5.2 Synthetic Communication Models 22
3.5.3 Emulations and Simulations 22
4 A Framework Solution 23
4.1 Knowledge Manager 24
4.2 Distributed Event Notification Service 25
4.3 Resource Manager Component 26
4.4 Security and Privacy Manager 28
5 Conclusion 29
References/Sources 30

Abstract

The coordination and collaboration of the personnel from various organisations involved in a rescue operation is important for a successful operation. Data networks can facilitate effective collaboration by providing an efficient and continuous flow of information at the accident scene (i.e., among rescue personnel) and with the outside world (i.e., headquarters of the rescue operation). Unfortunately, this is not easy to achieve because in many cases the

infrastructure is not available or not working. If we can assume that all the rescue personnel is carrying a mobile computing device enabled for wireless communication, these devices can provide the necessary network infrastructure. The devices can create Sparse Mobile Ad Hoc Networks (Sparse MANETs), which can provide a best effort infrastructure for information flow, i.e., data sharing and dissemination. These types of networks are very dynamic in terms of network resource availability, e.g., bandwidth or connectivity between nodes. Additionally, the devices can be heterogeneous in terms of available hardware and software resources. Another source of heterogeneity is the way a device is used; this is determined by the affiliation of the end-user and its role. In such an environment, software applications need middleware support in order to best use the infrastructure provided by the Sparse MANET. In the Ad-Hoc InfoWare and MIDAS projects, we develop corresponding middleware services. In this paper, we present a detailed requirement analysis for middleware services imposed by the application domain. To understand better the impact of the application domain, we have analysed the structures of organisations and the intra-/inter-organisation interactions during a rescue operation. We present three different rescue scenarios, which helped us perform a detailed requirements analysis for the middleware components. We have also identified the technical challenges imposed by the highly dynamic environment. We propose a middleware framework composed of four services; knowledge manager, distributed event notification service with watchdogs' component, resource manager, and security manager. The requirements presented here shaped our design decisions for these building blocks.

1 Introduction

The purpose of this technical report is to present background information and requirements analysis for developing middleware that supports information sharing in rescue and emergency operations. This is done through looking at how rescue and emergency operations are structured and giving example scenarios as basis for requirements analysis from different perspectives, such as organisational, security, network and communication, as well as application development constraints.

In emergency and rescue operations, rescue personnel cooperate to save lives and to limit the damages on, e.g., nature, buildings, and infrastructure. The rescue personnel have assigned tasks that they perform such as carrying injured persons to a safe place, providing food, water and blankets, doing examinations, perform investigations etc. To organise such an effort, people involved in the operation need to communicate; they must inform each other of tasks done, discuss what decisions to make, inform about decisions made, what to prioritise, make organisational decisions about usage of work forces, distribute work tasks, as well as exchange information and provide information to the public. If they cannot communicate directly, they use, e.g., radios and phones. As computer devices have become smaller and thus portable, and have wireless networking interfaces, such devices can also be used for more efficient information exchange. Sometimes a phone or a radio is the best means of giving an order, while other pieces of information, e.g., a map and a photograph, cannot be communicated in the same way. In addition, the laptops, PDAs, and smart phones have storage capacities and processing power that enable running applications that provide functionalities that are not given by for instance a radio. A variety of different applications that can be useful exist; filter information according to the role of the person carrying the device, to store data, process data, make computations and do data mining to provide additional information. The applications can have user interfaces tailored for the device and

the role of the person carrying the device. Sensors may also be deployed in the area, providing additional information to help in the decision making process.

To make it easier to develop distributed emergency applications, middleware services are needed. The middleware must be able to manage the underlying network. The middleware can not assume a working network infrastructure, or otherwise stable conditions. Some parts of the devices may be in an infrastructure mode, others in ad-hoc mode. In infrastructure mode, they may be connected to base stations using WiFi (IEEE 802.11), or in ad-hoc mode using for instance WiFi or Bluetooth (IEEE 802.15.1), sensors can send data to a sink using Zigbee (IEEE 802.15.4). If the nodes are connected in ad-hoc mode, they form a Mobile Ad-hoc Network (MANET). There may be more than one MANET, and during the running of the operation, there may be network partitions and merging. Devices may be turned off due to battery drain or they may be located out of range of other nodes or base stations. Unstable conditions are rather the rule than the exception. Another main challenge is limited resources, especially bandwidth.

In the Ad-hoc InfoWare [28] and the MIDAS [11] projects, we are designing middleware services for emergency applications. The focus of the Ad-hoc InfoWare project is mainly concerned with sparse MANETs. In MIDAS there may be a mix of nodes running in infrastructure mode and ad-hoc mode, formed by devices possibly having different network interfaces. There may be user nodes in addition to service provider nodes, and user nodes deploy the MIDAS middleware on the spot. In Ad-hoc InfoWare we assume an a-priori-phase where the middleware is installed on all the nodes, together with certificates etc.

In Ad-hoc InfoWare, the importance of resource management and network prediction is higher due to the characteristics of sparse MANETs, and this implies a greater need for asynchronous communication that manages network partitions. In MIDAS, the main focus is to have a distributed database without high demands of data consistency, monitoring the network and data usage of the applications to allow smart data replication decisions. In MIDAS, we also aim for a highly context aware middleware, e.g., a middleware that adapts to the usage of network types, for instance by switching network interface to provide better quality of service (QoS) to the applications. Another example is to support context-addressable messages using context-based routing that performs better than flooding parts of the network. The nodes can receive messages based on their context and/or role. This is helpful, e.g., in the situation where a group leader wants to directly send a message to all of the group members, or the possibility to send an “evacuate message” to nodes in a particular area without having to flood the whole network.

In this technical report, we first describe general characteristics of rescue operations, and how rescue operations in Norway are organised. We present three possible scenarios that are different in magnitude, area, and number of people that may be involved. After presenting the scenarios, we look at the commonalities and differences of the scenarios. We focus on issues relevant for the Ad-Hoc InfoWare middleware system, such as communication, organisational structures, security aspects etc. We then deduce middleware requirements from the scenarios. We discuss some technical challenges and provide an overview of the Ad-hoc InfoWare services.

2 Rescue Operation Characteristics and Scenario Examples

In this section, we first discuss some general characteristics of rescue operations, and how rescue operations in Norway are organised. Then we describe three scenarios from a "non-technical-perspective". The first scenario is a rescue operation after an earthquake. Such a scenario covers a large area and involving a large number of people. The second is a railway accident that is more limited in both area and the potential number of people involved. The third scenario, an underground accident located in the Paris metro, is adopted from the MIDAS project. The scenarios are quite different in size and area of the accident, but there are also some similarities, which we discuss in section 2.5.

2.1 Characteristics of Rescue Operation Organisation and Structure

Rescue and emergency operations are characterised by very hectic and dynamic environments, where time is a critical factor. There is a lot of movement and activity on the site as personnel may arrive and leave the site at different times, e.g., in cases where personnel or other resources (ambulances, helicopters) are called out to other incidents in the area. A rescue operation is also of a specific type, e.g., a land, sea, or air rescue operation, national or international involvement, of a certain size, etc.

2.1.1 Organisations Involved

Typically, several organisations are involved in the operation, e.g., paramedics, fire fighters and police, in addition to a number of other organisations, some of which are voluntary. In Table 1, we can see a list of the organisations typically participating in a rescue operation organised by the Norwegian Rescue and Search Service (SAR) [26]. The list is taken from the report from a serious train accident in Norway in January 2000 [29].

Table 1. Overview of participating personnel involved in the Åsta accident.

Participating personnel	**Number**
Police and bailiffs	73
Fire department/brigade	70
Medical and paramedics personnel	150
Air ambulance and helicopter personnel	13
The Norwegian Civil Defence	5
The Norwegian Armed Forces	203
The Norwegian Red Cross	29
Crises and care teams	43
The Church of Norway	12
Norwegian Rescue Dogs	2
External	4
Total	**ca. 600**

Source: Police/Kripos – adopted from [29]

The police has the main responsibility for rescue site coordination and management, and calling in other governmental departments/services as necessary. They are also responsible for ensuring communication with the county governor for establishing and maintaining mutual understanding and cooperation. Norwegian military forces are required to assist the police. Fire brigades have special equipment for rescue work in tunnels. The person responsible for handling the accident also has responsibility for information given to relatives, media, and central authorities. The (Joint) Rescue Coordination (Command) Central coordinates rescue operations, and a Rescue Sub-Centre is set up to handle regional rescue operation management. Helicopter resources and pre-hospital services are expected to be requested

directly by the rescue operation management, and the county is expected to support with several supplementing special services.

Each organisation has its own set of rescue operation procedures and guidelines that it must follow. The cooperation incentive in this situation is very strong since all participants share the same overall goal; rescue people and limit the impact of the disaster. Additionally, cross-organisational interaction procedures involving governmental organisation and other authorities have been defined. These procedures can be expected to include rules to establish a command and coordination structure for the scene of the accident. The coordination structure to some extent shapes the flow of information in the scene. It is however important to note that some deviations in the shape of information flow will always be present, due to cooperation among team members and on the spot decision making during the operation. Thus, spontaneous communication and information flow between rescue personnel during the operation should not be limited by a predefined structure, as personnel are likely to perceive this as a hindrance and a source of frustration. It has been found that feelings of not being able to access (in time) existing and possibly helpful information reduces the state of cognitive absorption and creates a feeling there is little effective control. The threat rigidity syndrome – additional stress caused by loss of control of the situation or reduced understanding of reality – has been observed in the emergency field [5].

2.1.2 Rescue Operation Organisation

Governmental authorities give cross-organisational guidelines regarding the operational structure and organisation of rescue and emergency operations, and the resulting procedures and policies imply rules regarding responsibility and the reporting chains in the organisations. As an example of an operational structure of rescue operations, we give a very brief description of the guidelines given by the Norwegian authorities for the Norwegian SAR [26] for land rescue operations and present an example model. Both the description and the model are restricted to include only what is relevant for our example.

Land operations are usually handled by the Rescue Sub-Centre (RSC), which has regional responsibility and appoints the on-scene coordinator/commander (OSC) at operation initiation. For larger operations there is usually an on-scene coordinating team consisting of a police officer in charge of public order, a fire officer in charge of fire control, and a medical officer in charge of medical treatment, all reporting directly to the OSC acting as the leader of on-scene coordination team (OSC-team). These officers all report directly to the OSC. There may also be a level of team leaders in each organisation (e.g., medical) reporting to the officers that are members of the OSC-team. Operational command is organised on three levels: the OSC – operational direction and coordination on the scene; the RSC – local resource coordination; and the top-level coordination of the entire operation by a (Joint) Rescue Coordination Centre (RCC). In most rescue operations, only the first two levels will be needed.

The main role and responsibility of the OSC is to coordinate resources and support at the site, and to accommodate the efforts of personnel from the participating organisations. It is of vital importance that the OSC has full overview of all available resources contributed from all participating teams and organisations at all times. The main role of the RSC is that of assisting and relieving the OSC-team, as well as coordination of resources, e.g., when there is more than one rescue operation in the area. The role of the RCC is mainly to monitor the operation and give advice.

In Figure 1, we give an example model illustrating the organisational structure of rescue operations. The model is based on descriptions taken from the Norwegian SAR. The figure shows role hierarchy and lines of reporting.

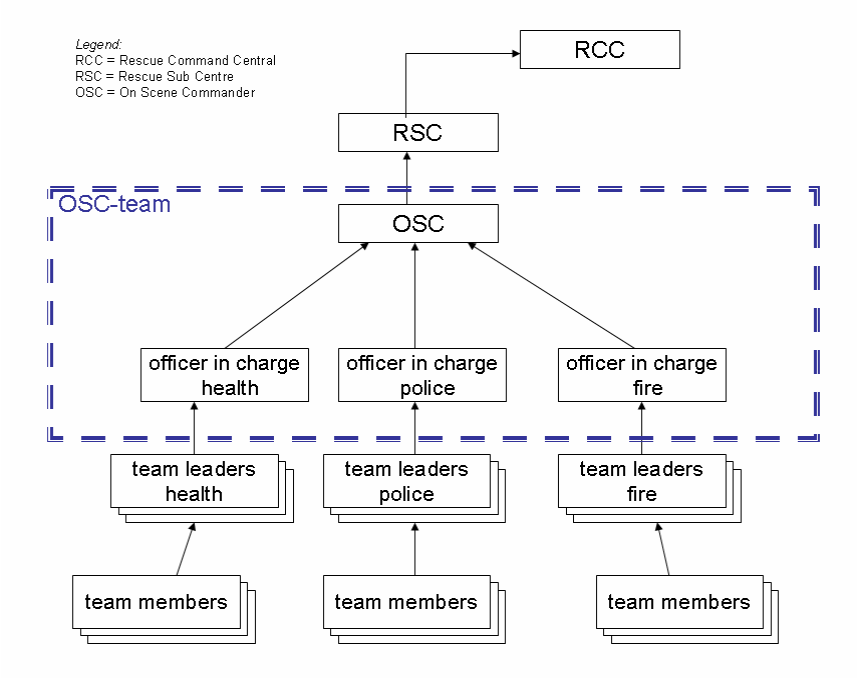

Figure 1: Organisation and structure in rescue operations

Norwegian health personnel follow directives for evaluating patients/casualties and appropriate action as given by the Norwegian Index for Medical Aid [25]. The emergency codes are Acute (Red), Urgent (Yellow), Regular (Green); it is common to use coloured tags during rescue operation/ on site. The Naca scale [34] – an eight level Severity of Injury of Illness Index for grading injuries or diseases - is another way of categorising casualties, ranging from 0 (no injury or disease) to 7 (lethal injuries or diseases).

2.2 Earthquake Scenario

A strong earthquake may cause a high number of casualties and injured people and severe damages. The rescue operation after a powerful earthquake can last for days and weeks and cover a large area. In the following, we describe in more detail possible damages caused by the earthquake, and how these damages might influence the intervention of the rescue personnel, and why information sharing is vital during the rescue operation. We have used as a source of information an earthquake scenario [9] that has a high magnitude on the Richter scale (above 7.0), and is located in a dense populated area.

During an earthquake the ground is shaking, this may cause falling rocks; there may be snow avalanches, rock avalanches, fires, falling trees, and landslides. The effects may be tremendous on buildings, roads, and infrastructure, for example buildings can collapse or get damaged by earthquake related fires. However, even though not all buildings and houses fall down, many people are injured by windows breaking, furniture falling, book shelves, wardrobes, lamps etc, also called non-structural damage. Bridges and tunnels may break down, and objects in the way, e.g., fallen trees, might block many roads.

The utilities in the area, such as electricity, gas and water lines, and sewer, may also be affected. For instance, the electric power may be gone in parts of the disaster area if the distribution system is damaged. It may take days to restore the electrical power supply, and the work of repairing the system is difficult due to the conditions. Likewise, water pipelines may be damaged, causing leaks etc. Power loss at groundwater pumping stations impacts the available water supply, rock falls damage transmission lines, and surface faulting and liquefaction damage to distribution lines. Rescue vehicles need refuelling, but distributing petrol may be difficult because of tanks at gas stations and pipelines may be damaged.

Effects on rescue operation

One important consequence of catastrophes affecting large areas, such as powerful earthquakes, tsunamis, etc., is that the rescue operation equipment, police stations, fire stations, and hospitals may also be damaged. Blocked and damaged roads delay the work of getting police, ambulance, fire department vehicles around. The roads that are not blocked may also be jammed by people who are frightened and fear aftershocks, trying to get away from the area.

Since a large number of people is affected, there are many simultaneous emergency calls. In addition to the high number of calls, telephone wires may be damaged or cut off, which makes communication difficult. GSM network may be overloaded. People are also likely to want to call relatives/friends/family or call authorities for help. Because of the high number of phone calls, many will not get through, try again, and further increase the pressure of the network.

Given the possible damage to existing infrastructure, contacting extra rescue personnel may be difficult in the first hours after the earthquake. Hospitals may also be damaged because of the shaking of the ground, mobile equipment falling down, the power may be gone, and there may be problems with portable generator services. Some casualties requiring hospitalisation may need to be transported to other areas. Hazardous materials/incidents are particularly problematic during earthquakes due to the potential for numerous simultaneous and widespread incidents. With resources stretched thin, the potential exists for casualties to go untreated for some time and for spills not to be cleaned up for weeks or even months. If there are factories or other industrial sites in the area, there may also be a risk of dangerous gas leaks that may explode.

Rescue operation communication and tasks

In such a large scale rescue operation, having efficient communication services and means for information dissemination can speed up the work and consequently may save lives. The devices of rescue personnel can retrieve information like patient files, drawings of buildings and maps, from the Internet or via a gateway to the Internet, and distribute this information to others. Pure ad-hoc communication mode can be used if base stations are damaged and the infrastructured network/communication does not work.

The first task of the rescue personnel arriving at the scene is to get an overview of the situation and the extent of the disaster. A Command Centre is set up. The next step is to get buried people out of the buildings. First aid is given at the place, and severely injured persons are transported to a hospital either by ambulance of by helicopter. Others are taken out from dangerous places, such as buildings that may collapse. Tags to indicate the severity of the injury are placed on the patients, and possibly also sensors for monitoring body functions. Schools or other official buildings not having collapsed nor in danger of collapsing can be

used as emergency shelter. Fire fighters are busy trying to put out fires caused by electrical wires, gas explosions, and buildings fallen down.

Applications running on nodes that belong to rescue personnel having coordinating roles, may subscribe to information such as the number of people living in the buildings, number of rescue personnel, who is involved, status, who has a leader role etc. So when another of these coordinating nodes receives such information, a notification about this information or a replica of the information is sent to the other nodes.

Some nodes may have a gateway to the Internet and get information such as drawings of buildings, patient files and maps. This may be very valuable information; for instance having access to building plans could help getting people trapped inside a collapsed building out, by knowing where there are concrete walls, lining wall and information about, e.g., abandoned ventilation systems, abandoned electrical chases or concrete shafts, that can be used to get to people locked inside.

There are many flows of communication:

- People working within a group, this may be a group with people from the same organisation or from different organisations.
- Information flow from rescue operation leader to the next level in the hierarchy, about what to do next, instructions, etc.
- Information from organisational leaders to rescue leader about the work, available personnel, and so on.
- Information from coordinators to groups working on the scene, possibly information received from "outside", task lists etc.
- Information from "ground people", working on the scene, giving first aid, stopping fires, repairing electrical wires, leaks, securing dangerous goods, etc – about how their work is going, what is done, equipment needed.
- Information from sensors – to people on the scene – health monitoring sensors, sensors measuring temperatures or pressure in tanks that may explode.
- Information from other experts, not involved in the rescue operation.

The resources on the scene can for instance be electronic maps of the disaster area or electronic charts of injured people. For example, electronic maps of the area can be annotated to show obstacles and indicate available access paths in the area. This requires for all maps on devices at the scene to be annotated and publish a change request for all later arrivals of devices carrying maps. In case the access path is cleared, an update needs to be propagated in the network to all the changed maps. These actions require fast localisation of resources on the scene, in this case the electronic maps of the area. Electronic medical charts of injured people represent a very important resource since medical personnel having access to it can save lives. The chart of a patient should follow the patient, meaning the chart should be replicated ahead to the next treatment points. In cases where there is no end-to-end communication path between two consecutive treatment points, the chart needs to be physically carried together with the patient. These actions require localising the devices where the charts should be replicated, such that it insures against loss or late access to the medical data of the patient.

2.3 Railway Accident

A serious railway accident in inaccessible terrain can be caused by, e.g., landslides/rockslides, technical failure, sabotage or collisions. If the accident is at a mountain pass, it will normally

be in areas with limited infrastructure and scattered buildings. As every train can carry close to 500 passengers, such an incident at an inaccessible mountain pass will cause acute, extraordinary need for transportation, and an immediate need for transporting injured passengers from the accident site to a collection site for pre-hospital treatment, or directly to a hospital. The rescue service has established routines for this, and there are helicopters for airlifting patients out of inaccessible areas. In addition, there will be a need for transporting other (not injured) passengers to an appropriate collection place, and alternative transport bypassing the accident site has to be established for a period. There may also be requirements for assistance in delivery of equipment to the accident site after the evacuation phase. Relevant organisations and governmental departments have certain responsibilities and roles in connection with limiting damage of accidents as was described in 2.1 above. The consequences of such an accident will have effects on both health and environment; there may be a large number of casualties and local pollution, e.g., oil. In cases of trains carrying both goods and passengers through (densely) populated areas, the consequences may be catastrophic, with a large number of casualties both injured and dead.

The railway accident in this scenario is located at the Bergensbanen railway in Hordaland county, in a tunnel at a mountain pass, e.g., in or close to Raundalen. The area has weak infrastructure and a need for special services. Jernbaneverket (the Norwegian National Rail Administration) is responsible for the national railway network, which is 4087 km with a total of 704 tunnels (of which the longest is Romeriksporten (14 580 m)) [15] [36].The Bergensbanen (Oslo – Bergen) is 482 km long and has 155 tunnels between Hønefoss and Bergen (372 km). Yearly, about 600 000 passengers are transported across the Hardangervidda. The possibility for an accident is thus high.

The accident is caused by a rockslide just outside of a tunnel; large blocks of stone are lying on the tracks. The rockslide has weakened the ground beneath the tracks, and the tracks are partly broken. The temperature is minus 10 degrees Celsius, with deep snow in the area. A nearby mountain lodge can be used for collecting evacuated train passengers. The lodge can be accessed by a mountain road which can be opened up for accessing traffic. The railway tracks can not be used due to the damage to its structure and danger of repeated rockslides, and the fact that it is blocked. A train between Oslo and Bergen, carrying about 400 passengers, runs into rocks when coming out of a small tunnel, it goes off the tracks and the train engine and some of the carriages are lying on their side. The train is partly inside the tunnel. The locomotive has gone off the tracks and is lying on its side, smashed. There are a number of casualties in the train. Some people in shock have walked out of the tunnel themselves. Others are still inside the carriages, some trapped under luggage and train parts due to the crash. One of the train carriages is completely crashed. The train has a diesel locomotive. Diesel cannot take fire unless it is already preheated to a temperature above a certain point (depends on the diesel, say about 60-70 degrees Celsius). Unless exposed to open flame or spark, it will not take fire until a temperature of 235-245 degrees Celsius, at which point it will light immediately without any spark or open flame [29]. This means that the diesel can ignite in this scenario either if nearby fire heats it up to its critical temperature, or preheated diesel catches fire from an open flame or sparks from a nearby fire or explosion.

After the incident, the train driver follows the appropriate procedure, and reports the incident and location to the train control centre. The person on duty at the control centre contacts ambulance and fire department through an emergency call, starts the internal emergency procedures for Jernbaneverket [15], and organises for other trains on the same track to be kept on hold (waiting), stopped, or redirected if possible. The emergency central is alerted, and a

rescue operation is initiated. The emergency central and participating organisations start gathering information about the area and available resources, e.g., maps of the area, weather condition information, available personnel and equipment, etc. All personnel get relevant parts of this information to their devices before/on leaving for the accident (briefing personnel). The incident is reported to the rescue service. RCC together with RSC launches/starts a rescue operation to evacuate those in need of acute medical treatment. Remaining and not injured, or slightly injured, persons are, due to weather conditions and terrain, also in acute danger and needs taking care of by the rescue services.

The tunnel, rocks, and train carriages hinder communication. The mountain pass is a difficult accessible area, which puts extra demands and limitations on rescue operation, personnel, and equipment. The low temperature and the deep snow in the accident area creates extreme conditions, which in addition to implications for the rescue operation also may have an impact on how well the devices will function. There is also a danger of repeated rockslides in the area. The lack of accessibility by road means that special vehicles – snow scooters or snow mobiles and helicopters – are needed for evacuation and other transport in/out of the actual accident area (both to the collection place and directly to hospital). As noted above, there is a high risk of fire, especially as the crash may have caused diesel from the locomotive to spread outside of the tank and this may have created a kind of "diesel fog", consisting of very fine diesel drops and oxygen, which may ignite very easily. Also mentioned, diesel itself needs to reach a certain temperature before it will take fire. Thus, avoiding fire to spread close to the locomotive is very important. Sensors can be placed around the diesel tank and locomotive to monitor the temperature so personnel can be alerted in case of increased danger of fire or explosion.

Rescue operation communication and tasks

Sharing information in this scenario may be very useful for personnel, and a MANET can provide the needed infrastructure so devices can communicate. If a MANET is to be used, nodes and routing daemons will have to be started up at the site to set up the MANET. The devices connect on arrival and become nodes in the network. Nodes inside the tunnel cannot communicate with nodes outside the tunnel, and because of hindrances, there is limited communication range inside of the tunnel. In addition, given the activity on site and possibly poor infrastructure in the mountain area, means there may be frequent network partitions and a node may be disconnected from the network for periods.

The leader of the team arriving first has the role of temporary rescue operation leader or on-site-commander – OSC. Once the police arrive, the higher ranked police officer will take this role. The OSC sets up a place of command, and tries to get an overview of the situation based on information from the train control centre, and the train driver, e.g., the number of wagons, passengers, any goods etc. The OSC coordinates equipment and personnel as they arrive. As the fire brigade arrives, fire fighters are going inside the tunnel - they are wearing sensors that monitor their heart rate and temperature.

All personnel are involved in evacuating people from the carriages, and they are moved away in safe distance from the wreck. Medical personnel start evaluation of medical state, register all persons, and mark each with a colour tag showing the degree of injury and need for acute treatment (red: acute/immediate, yellow: not immediate, green: no injury). They are then transported to the mountain lodge for further treatment. Sensors are placed on patients in stable conditions to monitor their state (e.g., heart rate, oxygen flow, blood pressure, temperature, etc). Team leaders receive information about the location of their subordinates.

They may have received a task list and can check out the tasks as they are completed. This information is then sent to the work leader. The police start gathering evidence to investigate causes of the accident.

There are a number of possible communication flows in this scenario. We list three examples of such communication flows here: The first is among *team members from the same organisation*, e.g., between doctors sharing registration and medical information about patients they share responsibility for, or fire fighters (devices) sharing temperature information from sensors in monitored area. The second example is among *members of task oriented ad-hoc teams* (can be cross-organisational or not), e.g., in railway accident a team targeted for going through a certain train carriage to report situation etc, consisting of paramedic, police, fire fighter, possibly rescue dogs (to find people trapped or hidden in the wreck). The third example is communication between *different levels in the rescue operation organisational hierarchy*, e.g., RSC and OSC, team members and team leaders, and team leaders and OSC.

The landscape where this accident happened has a big impact on the way the topology of the communication network is shaped; specifically it can create partitioned networks such as:

- Inside the tunnel, formed by devices carried by the rescue personnel and sensors placed in the tunnel.
- At the un-blocked end of the tunnel where most of the teams may be situated and active. For example, first aid might be given to the injured here, the on-site commander centre may be situated here, or resources needed during the rescue operation may be stored here.
- At the blocked end of the tunnel where personnel try to clear the tunnel opening.
- At the lodge where there is established a transportation hub for taking the injured to the hospital and distributing resources for the accident scene.

In such a network, data and services must be made available to all the possible network partitions. This requires finding the best way of using the available resources on the nodes to run services that help the rescue intervention. In addition, the mobility of the nodes must be used in order to transmit data between the network partitions.

2.4 Subway Station Accident

The subway station accident described in this scenario is located in the Jussieu underground station of the urban metro system of Paris, France. It is based on a scenario description provided by the Regie Autonome des Transports Parisiens [19]. The area can be considered to have a good and well-maintained infrastructure, compared with the one of the railway accident scenario presented in the previous section. Certain communication services might be available in the metro station but may be missing in the tunnels between the stations.

There are two trains in the scenario: Train 1010 going east that is about to stop at Jussieu station, and Train 1020 going west that stops on the opposite platform. The driver of Train 1010 sees smoke coming out of car number 2. He immediately reports the fire to the operation control Centre (PCC) and tries to give supplementary information about the fire, such as where the fire is located. He requests the passengers to step out of the train. The PCC turns on the fire alarm and shuts down the “traction power” for the trains in the area of the Jussieu metro station. The fire alarm is automatically reported to other devices in the area, such as the mobile device of the driver of Train 1020. The PCC asks the passengers to evacuate the Jussieu metro station and provide traffic information for passengers on nearby stations. The

PCC passes the information about the accident higher up in the hierarchy to the operation duty inspector (IPEX) . The drivers of trains 1010 and 1020 take pictures with their mobile devices; these are automatically annotated with correct metadata, e.g., date, time, incident X, Jussieu metro station. The pictures are sent to PCC, which can use this additional information to decide which intervention procedure is most appropriate in this situation. The IPEX calls for external support from different actors, e.g., fire department, ambulance, police, etc; and inform leaders, such as the Metro director. During this time, the passengers are evacuated from the metro station and the station is secured.

Due to the existing security risks, the metro has a set of different emergency procedures, e.g., as with small accidents – secure the place, call for help, help the injured, etc. It also has a very precise hierarchy for the information flow, i.e., whom to inform, and what phone number to call (home/office/mobile), depends on who is on duty. All this information is context dependent and immediately available, which shortens the briefing phase. When an operation supervisor arrives at the scene, he becomes the incident manager. The incident manager receives information from the mobile agents at the site and sends reports to the control centre. The IPEX can then inform agents at close by stations about the status of the accident, and ask them to evacuate the station. They may also send personlised list about what to do, e.g., station evacuation, smoke removal system command, public information, help passengers, unblock validation line, etc. When a task is completed it can be checked, and this information is then automatically sent to the control unit.

Different groups of people are involved; train driver, people working at the station, emergency teams, e.g., paramedics, police, fire fighters, etc., and accident leaders from the metro department. In this scenario, the participants have mobile devices for communication and information sharing. They use ad-hoc and relayed communication in tunnels/underground stations when GPRS and WiFi are not available.

2.5 Scenario Commonalities and Differences

In this section, we examine the presented scenarios to find commonalities and differences useful in a requirements analysis for designing middleware support for information sharing in such scenarios.

The scenarios described are quite different, e.g., with respect to the landscape, the size of the area of the accident, number of people involved, available resources, time span etc. The earthquake scenario covers a large area and involves many people. There may be huge infrastructure damages and many casualties, and may last several days. In contradiction, the train and metro scenarios are more limited in time and space, involving less people, fewer casualties and rescue personnel, and the number of bystanders and non-organisational people involved will likely be much lower.

It is however important to emphasise the similarities as these provide the basis for a requirement analysis for building a middleware. The differences may represent what needs to be configurable in such a middleware. The degree of heterogeneity, and to a certain extent the complexity, will in general increase with the size of the area, the time span, and the severity of the accident. In the following, we look into different aspects such as people involved, organisations, and ways of communicating, in addition to similarities and differences in the three scenarios.

Many of the same organisations are involved in the scenarios, there are medical personnel, fire brigade, and police at all three scenarios, and they have similar tasks:

- **Rescue site leader (OSC):** the main tasks are to set up a place of command, get overview of the situation, coordinate equipment and personnel, assign tasks, and report to control centre. In the railway scenario, examples of gathered information about the situation from the train control centre and train driver, e.g., number of wagons, passengers, any goods etc.
- **Fire brigade:** The main tasks are to control fire, as well as to monitor areas in danger of fire or explosion. Other typical tasks are to cut loose trapped people, help where needed, and place sensors to aid the monitoring of dangerous areas.
- **Medical personnel:** The main task is medical care, including registration of patients and evaluation of medical state. If sensors are used in aiding patient monitoring, medical personnel will place these on patients.
- **Police:** tasks include gathering evidence and securing the area.
- **All personnel** are involved in evacuating people and in general cooperating and supporting the rescue operation as needed.

Differences concerning organisations and personnel are related to the number of people involved; in general, the larger incident, the more people, more volunteers, and different kinds of experts involved. For instance, in cases of gas leaks or oil spills, people with the appropriate knowledge are required. In the train accident scenario, people from Jernbaneverket and the Norwegian State Railways (NSB) are involved, and in the earthquake scenario people to repair damaged infrastructure are needed, e.g., roads and telephone system, as well as teams for searching for missing people.

Possible sources for information include both mobile devices carried by personnel, stationary devices, PCs in ambulances and rescue helicopters, and sensors. Another possible information source is an Internet gateway. The information from these sources can be shared, but there are cases when sharing of sensitive information is not desired. As sensitive information, we can classify medical records of injured persons, environmental data, layout of buildings and installations, information about dangerous goods, collected evidence, available resources, and status reports.

Communication issues with respect to technical equipment may also differ. The train scenario is located at a deserted place, there is a lack of infrastructure, they need to put up their own MANET, and there are problems/challenges related to communication between people inside and outside of the tunnel. Weather conditions may also have an impact; devices may not work properly in extreme weather, e.g., temperatures well below zero in the train accident. In the earthquake scenario, the infrastructure is likely only half-functioning, and there may be different means for people to communicate, either ad-hoc using wireless devices, like laptops and PDAs or other pre-setup devices, or if some infrastructure is available, using mobile phones. In addition, radio and TV stations may broadcast public announcements. Depending on the condition of the GSM-network - mobile phones may sometimes work, sometimes not.

Regarding the communication lines, there are many similarities, as the hierarchy in an organisation is the same regardless of what kind of rescue operation. Possible communication lines include:

- among team members from same organisation
- among members of task oriented ad-hoc teams (can be cross-organisational or not)
- between different levels in the rescue operation organisational hierarchy

However, the bigger operation, and the longer time span, the greater the need for improvising. Thus, there may be differences concerning cooperation procedures and hierarchical levels depending on the size and kind of operation. For instance, the hierarchy in the metro scenario has few levels, creating a shorter way from top to ground personnel. It is also strict, in the sense that procedures are very detailed and people involved are drilled in the same manner. The size and type of operation will also have an impact regarding getting an overview and briefing personnel, as it takes longer time to get an overview of the earthquake scenario, than in the train or metro-scenario.

In large scenarios, it is important to get relevant information to the public, in order to save lives. In the train scenario, such information distribution is not so critical, except for redirecting people travelling, although informing families of the victims will still be important. In the earthquake scenario there may be difficulties concerning supplies of clean water, gas and electricity, while in the other two scenarios, these problems are less critical because the area is more limited, and supplies can be transported from outside the rescue area. The main transportation needs in the train accident are to evacuate people and get supplies in to the area, while in the earthquake scenario, people will also have to be evacuated out of dangerous areas and taken to shelters and centres for medical care. Here, greater difficulties finding non-damaged vehicles and open roads may be a case for concern.

Some of the commonalities can be extracted into a set of general rescue scenario phases tailored for using MANETs in rescue operations. In the Ad-Hoc InfoWare project, we have identified six rescue scenario phases, as described in the following. The phases were previously published in [22].

Phase 1 – A priori: This phase is before any accident takes place, when the relevant organisations - in cooperation with the authorities - exchange information on data formats and shared vocabularies, and make agreements on procedures and working methods. Required certificates would be installed in this phase, and applications can be installed and run so as to allow completion of an initial self-configuration phase. A communication and knowledge environment tailored to relevant applications can be prepared by the middleware, and data replication strategies chosen. In a context aware system, contexts reflecting different scenarios can be prepared, group memberships based on user profiles set up.
Phase 2 – Briefing: This phase starts once the incident has been reported. The briefing involves gathering of information about the accident, e.g., weather, location, number of people involved, and facilities in the area. Some preliminary decisions about rescue procedures and working methods are also made at this stage. Based on information gathered during this phase, applications can be configured further, security levels chosen, and, if applicable, relevant rescue contexts and profiles put in force.
Phase 3 – Bootstrapping the network: This phase takes place at the rescue site, and involves devices joining and registering as nodes in the network on arrival. In addition, the appointing of rescue leaders takes place in this phase. By preparing communication and taking care of security restrictions in force, the middleware can improve the working environment of the applications.
Phase 4 – Running of the network: This is the main phase during the rescue operation. Events that may affect the middleware services include nodes joining and leaving the network and network partitions and merges. Information is collected, exchanged and distributed. There may be changes in the roles different personnel have in the rescue operation, e.g., change of rescue site leader. New organisations and personnel may arrive and leave the rescue site, new

groups of an ad-hoc, task-oriented kind may form, possibly involving people from different organisations. Applications communicate about available resources and capabilities of the nodes in the network, using whatever knowledge is provided by the middleware. It can update to changes in available resources as the network is evolving, query for more data or information as it becomes available, and adjust its configuration and behaviour accordingly. Computing resources, processing environments and applications situated at neighbours can be utilised, using resource information provided by the middleware and obeying accepted policies for resource sharing. Replicas and proxies can be placed at strategic nodes in the network, and nodes can receive event notifications based on relevance and priority. As nodes join and leave the network the middleware can keep track of available resources and adjust its communication and knowledge environment accordingly.
Phase 5 – Closing of the network: At the end of the rescue operation all services must be terminated. Applications can adapt to the closing of the network by acting on received information about degradation of the capabilities and resources of the network.
Phase 6 – Post processing: After the rescue operation, operation specific data, e.g., resource use, user movements, and how and what type of information was shared, may be analysed to gain knowledge for future situations. Depending on the nature of the application, it may have gathered statistical or other information for post scenario analysis or for future use.

As we have seen from the described scenario, fixed networks cannot be relied on during the rescue operation itself. In the opening phases (phases 1-2), there are however no such restrictions, which give possibilities for preparations that to some degree can compensate for a lack of resources during the rescue operation.

3 Requirements Analysis and Technical Considerations

In this section, we discuss various issues from the application scenarios that influence the development of the middleware services.

3.1 Organisational Intra- and Inter-operability

From the rescue scenarios described earlier, we can see that rescue operations are characterised by cooperating personnel from a number of participating organisations, and a hectic and dynamic environment. Each organisation handles different aspects of the operation, and needs different information or possibly disparate views of the same information. We have also seen that a rescue and emergency operation is an organised venture, with a certain structure of responsibility and reporting, and that specialised procedures and policies are followed. For the coordination and organisation of the operation, it would be beneficial if applications can access available information shared automatically among the participants and across organisations. Examples of such applications mentioned in the scenarios are for dispatching of personnel and equipment, on-site identification of passengers, and registering casualties for medical treatment.

To accommodate the heterogeneity of organisations involved, the information should be presented in a way that all organisations can understand. This implies supporting functionality similar to high-level distributed database system functionality, querying available information and keeping track of what information is available in the network. The middleware must account for different domain ontologies and standards that might be used by the organisations. A major challenge is to support information sharing across organisations such that they understand each other’s structure and data descriptions. There are cross-organisational

procedures involving governmental and other authorities as well as internal procedures that each organisation has to follow in a rescue operation. Thus, there should be some support for both intra- and inter-organisational structure in the middleware. Contextual support may be beneficial as contexts can be used for reflecting specific rescue procedures in force, for supporting user role profiling and personalisation, device profiling, providing temporal and spatial information, movement patterns, etc.

The organisations involved, each covering different areas of expertise, may use different data models, structuring and syntax, as well as appropriate domain ontologies for their respective areas. Thus, to achieve cross-organisational information sharing and interoperability we need support for handling structural, syntactic, and semantic heterogeneity. The domain of distributed databases has solutions for structural and syntactic heterogeneity. For semantic heterogeneity, two issues need addressing; the first relates to understanding, and concerns mapping and translation between different vocabularies (and data models), for instance through using a common base minimum model; the second concerns the serialisation of the vocabularies in agreed-upon standard languages and syntax, e.g., XML and RDF.

Organisations working towards common terminology and standards in the health sector include CEN/TC251 [6](Europe) focusing on standards in areas including information models, terminology, security, and interoperability; HL7 [13] (USA) focusing on standards for the exchange, management and integration of electronic healthcare information; and KITH [24] (Norway) working on standards in areas including information security, message exchange, and terminology.

Work within the area of knowledge management has found that sharing knowledge in the form of metadata about where knowledge resides, often can be as important as the original knowledge itself [1], for instance as organisational knowledge maps, i.e., sharing the “what and where/who” of information available in an organisation. The fact that a rescue and emergency operation is organised with designated roles, lines of reporting, procedures and a certain structure, and as such can be seen as an ad-hoc organisation created for the purpose of the operation, means that it may be beneficial for information sharing in rescue operations to share this kind of metadata.

3.2 Security Aspects

Use of wireless networking presents the system with a series of security related challenges. The term *security* comprises four main categories: authenticity, integrity, confidentiality and non-repudiation. Starting from the medium itself, one of its main characteristics is that anyone with a tuned wireless device can *hear* the traffic in the network, as well as induce traffic into it. Since neither of the two issues can be efficiently prevented in a rescue operation, they introduce problems into at least three of the abovementioned security categories:

- authenticity: mechanisms need to be present to ensure that only authorised devices can be part of the network
- integrity: it must be ensured that no one can manipulate with the network traffic by, e.g., changing its contents before retransmitting it
- confidentiality: some data must be kept confidential, i.e., not available to unauthorised recipients

Another issue not directly related to data security but still directly related to the efficiency of the network (and as such to the efficiency of the rescue operation itself) is signal jamming. Unlike wired networks where a device needs physical access to the medium to perform any

action, wireless networks are highly prone to such attacks and there is no real solution other than locating the perpetrator and disabling the source of jamming.
All the abovementioned issues are considered *external* attacks [16], i.e., attacks coming from nodes that are not (or not supposed to be) part of the network.

A much different, and in some cases more dangerous type of attacks are so called *internal* attacks [30].These attacks come from nodes that have already been authenticated and as such considered a legitimate member of the network, but at some point has become compromised. This can happen if devices are stolen, lost and then found by a malicious person, or even worse, if their legitimate owner becomes compromised. Detecting and excluding such nodes is a much more difficult process and requires specialised ideas and solutions.

Other than attacks, the system must take care of confidentiality within the network. Inter-organisational collaboration is one of the key functionalities in a rescue operation. Nevertheless, different organisations may have different security requirements and policies and, in addition, not all data within the network should be seen by every member. Examples of such data may be medical records, police records, other personal or confidential information, etc. Additional challenges are imposed by different organisational structures and levels of confidentiality within the organisations themselves, which could also change dynamically.

3.3 Network Characteristics

Deployment of communication networks in the affected area must be quick, and must not make any assumptions about the available infrastructure in the area. Therefore, utilising MANETs that are formed by the wireless devices brought into the area and carried by the rescue personnel is the most promising approach for this application domain. However, the layout of the affected area, physical obstacles in the area, and the mobility of the network nodes might lead to frequent and/or long term network partitions. We call this kind of networks Sparse MANETs, which can be seen as a combination of MANETs and Delay Tolerant Networks. These are typically highly dynamic networks in terms of available communication partners, available network resources, connectivity, etc.

3.3.1 Devices and Network Topology

In the train accident described in Section 2.3., injured passengers may be leaving the tunnel through both ends, in which case rescue personnel have to be at both ends to take care of them. Additionally, some rescue personnel might be in the middle of the tunnel to search for passengers and to extinguish the fire in the train. In this situation, the network might be split into three partitions, i.e., one at each end of the tunnel and one in the tunnel. Obviously, these network partitions must function independently as good as possible and provide the rescue personnel access to all mission critical services and information.

The end-user devices are very heterogeneous, ranging from high-end laptops to low-end PDAs and mobile phones. CPU storage space, bandwidth, and battery power represent important resources. Finally, many application scenarios, like coordination of rescue teams, have also quite hard non-functional requirements such as availability, efficient resource utilisation, security, and privacy. Both the heterogeneity of devices and the broad range of functional and non-functional requirements, impose the need for resource management mechanisms.

In Sparse MANETs, the middleware for resource sharing based on traditional resource reservation will not work in a proper manner, and guarantees for resource availability cannot be given. Instead, best effort resource reservation might be treated as a soft-state which is only valid for a specified time, either for the time period the resources are needed exclusively by a process, or for the time period the resources are (with high probability) accessible. Resource management can benefit from predicting future availability of resources, not only to establish meaningful time-outs for soft-state reservations, but also to increase the availability of information and services through replication and graceful degradation. There are several approaches to address prediction of future connectivity and by this, future access to resources and services. One approach is to analyse location and movement of nodes with GPS information. However, GPS devices might not always work, e.g., in buildings and tunnels. The most common approach is to assume that at least a fraction of the devices are location-aware, e.g., with GPS, in order to improve information accessibility in MANETs. However, a GPS based solution is not reliable enough, especially for critical services like rescue services. The GPS receivers do not work in many locations, for example inside buildings, tunnels, close to large buildings, and even dense forests. Therefore, it is important to be able to predict network connectivity in the absence of GPS support. Another solution for location-awareness is to combine sensor data from sensors such as accelerometers, compasses or acoustic sensors. This requires that each device has access to such detailed sensor data, which is highly unlikely due to the heterogeneity of the devices.

The network infrastructure and device limitations impose the following set of requirements onto a resource management solution:

- *Communication non-intrusiveness*: The resource prediction algorithm should not rely on frequent updating of information about available resources on remote nodes.
- *Position independence*: The resource prediction algorithm should not require the exact geographical location of the device or of other devices.
- *Routing protocol diversity*: The resource prediction algorithm should work with any routing protocol.

3.3.2 User Mobility

In a rescue situation, the movement of the nodes can be classified using different criterias, we consider physical, social and network mobility. The **physical mobility** of the node distinguishes *mobile nodes* that change their physical position with respect to the other nodes, and *stationary nodes* that move insignificantly after their deployment. The **social mobility** of the nodes considers the relationships of a node to other nodes, meaning a node can be a *singular node* or part of a *group of nodes*. Additionally a node might have stabile, variable, random, or periodic group membership relations with other nodes. The **network mobility** of a node describes the way a node is available for communication, in other words if it is continuously connected to a network or not. The *continuous connected nodes* are available for communication all the time and they can participate in synchronous communications. The *intermittently connected nodes* participate to communications only occasionally due to the device's physical constraints, for example sensor nodes, or due to physical mobility, for example cars. In many cases, the only way nodes can communicate is in an asynchronous manner.

We can assume that the nodes have a very strong incentive to collaborate. The nodes have various resources and services they are willing to share and they provide services to each other. We also assume that it is a negligible percentage of selfish nodes, i.e., not willing to collaborate and share resources and services. This means that in the worst case, the nodes

collaborate only by sharing bandwidth and providing routing. In the ideal case, nodes share resources in such a manner that storage of information and computation becomes pervasive. Unfortunately, the way the groups are formed might not be fully known to the middleware because there might be “off-line” agreements between the rescue personnel. This might lead to teams and actions unforeseeable for the middleware, i.e., not according to knowledge available on the nodes.

In conclusion, the middleware needs to reduce its communication overhead in the network. It also needs to reduce the requirements for specific functionalities from the devices. However, it can take advantage as good as possible of the information extracted from the routing protocol. This information is “up-to-date”, it is available on all nodes and it comes at no additional communication cost from the middleware.

3.4 Communication Issues

The user devices can connect and exchange information using, e.g., wireless network technology. If the MANET is fully connected, any two nodes can communicate synchronously, using some other nodes as routers, or directly, if they are one-hop neighbours. Information can also be flooded into the network. The task of finding paths between two nodes is performed by a routing protocol. There are two main groups of routing protocols: reactive and proactive. The reactive protocols find a path between a source and a receiver when the source wants to send a message, the proactive protocols keep routing tables up-do-date at all times. We omit in this report a description of different routing protocols. As described in the scenarios, we may not always be sure that the network is fully connected at all times. Tunnels, mountains, or buildings may block the signals, or the area may be too large. In addition, devices may be switched off for a time period to save battery. Thus, we need a communication service that is able to handle sparse MANETs where any two nodes may not always be able to communicate directly. In publish/subscribe systems a subscriber subscribes to events of interest and a publisher publishes notifications about such events. The subscribers and publishers usually connect to a node or nodes running the service, often called broker or mediator, and they are thus de-coupled in time and space [10]. If a subscriber is disconnected, it may receive notifications when it reconnects if the service provides storage of undelivered notifications. However, there may not always be the case that network partitions merge; the communication service should therefore also support the possibility of *routing in time.* This has been described as the *store-carry-forward* paradigm, or *epidemic routing* [35]. Epidemic routing may however result in too much information being replicated. In Message Ferrying [37] the randomness is less as some dedicated nodes are used and following pre-defined paths for doing the store-carry-forward-operations. This lowers the amount of data carried and replicated. Thus, the publish/subscribe paradigm provides a means for asynchronous communication. For delivery of notifications the service can use unicast or multicast as provided by an underlying routing protocol, (or flooding), or it can use store-carry-forward using all the nodes, or a subset of the nodes, as in message ferrying.

The subscriber expresses its interest according to the supported subscription language. In a rescue operation, all kinds of events can be of interest such as sensor data, new resources, new nodes, availability of a map, updated information on patients, and health sensor data. Different applications have different needs for subscription languages. In a topic-based language, the subscriber subscribes to one or more predefined subjects and each event has a subject. The content-based language is more expressive as the subscription is a Boolean function used on the content (“sensor = temperature, temp_value > 35C”). In a hectic environment, there may not always be time to give crisp values or value ranges for a

subscription predicate. Selecting terms such as “warm”, “high” etc may be more valuable and may have different meanings in different contexts. The most common way to bypass this uncertainty is to replace approximate values by crisp values so the subscription does not necessarily reflect the real interest of the subscriber. There may also be cases where a subscriber subscribes for a composite event containing different simple events from various sources. Some subscribers may also be interested in trends over time that could be expressed as continuous queries. In some cases, a simple topic based language is sufficient, in other cases a richer language specifying content, location, time constraints and perhaps semantic uncertainties are needed. The more complex subscription language the more specific and to-the-point-subscriptions are possible for the application to make, the disadvantage is that the filtering algorithm will be more complex as well, and doing routing optimisations is more difficult.

In a rescue operation, individuals from different organisations cooperate by sharing information. With respect to the current state-of-the-art, it is reasonable to assume that each organisation is using its own standardised vocabulary and data model. Therefore, it is also desirable to support subscriptions that are targeting different vocabularies. This support must also include the translation or mapping between different vocabularies since information in rescue operations is shared also across different organisations. For example, the coordinators of fire fighters and police officers might be interested in the temperature in a certain room. They may use different terms for temperature even if the sensor supports only the fire fighters’ vocabulary.

In [21] challenges related to communication are discussed; technological, sociological and organisational challenges. First, they deal with some technological issues related to usage of radios because different organisations may use different radio frequencies, and how existing infrastructure may provide interference and create more problems. They also mention the need for devices having different network capabilities to make them more fault tolerant. The second communication challenge is related to sociological issues; understanding how humans communicate and behave under stressful conditions, and the need for periodic news updates. As previously mentioned, part of the challenge in developing useful communication services is to overcome the difficulties that organisations use different ontologies and vocabularies. Also mentioned in this article is that emergency operations can benefit from using a system where people can cooperate easily across organisations and not always follow pre-defined hierarchies if that is not efficient.

3.5 Development Constraints

Development of applications and middleware for rescue and emergency operations faces a set of reality-oriented constraints such as real movement of personnel, i.e., nodes’ movement on the scene, real network traffic and load, i.e., communication needs and patterns. Additionally, the realistic software development and test environment constitutes a problem. We discuss in the following, how these influence the development of rescue and emergency applications.

3.5.1 Synthetic Mobility Models

As mentioned, to have realistic mobility models is important to test if the developed solution can overcome the dynamicity of the system. To test during the development phase one can use mobility traces obtained with synthetic mobility models, computer simulated rescue operations, and real rescue exercises.

One can use the synthetic mobility generators that provide random movement, such as the single mobility models *Random Walk Mobility* [8], *Steady-State Random Waypoint Mobility* [23] and *Random Direction Mobility* [32], and many other. As group mobility model we use *Random Point Group Mobility Model* [14] and its variations. Each synthetic mobility model has different characteristics that generate some unrealistic mobility patterns. For example, in the *Random Walk Mobility Model* a node chooses at random a speed and a direction from predefined sets. In case a node reaches the simulation boundary, it will bounce and continue in the new direction. The node moves for a constant period of time or for a constant distance, at the end of which a new direction and speed will be calculated. This model has been reported to produce regions with high density of nodes and that nodes roam around their initial position. In the *Steady-State Random Waypoint Mobility Model* a node chooses at random a destination and a speed which is drawn from a uniform distribution between a maximum and minimum speed. At the destination, it will pause for a random period of time and then start moving again. In this mechanism, the destination of a node has a higher probability of falling close to the centre than to the edge of the simulation area. This leads to a higher number of nodes being found at the centre of the simulation area than at the edges. The *Random Direction Mobility Model* was designed to create an even distribution of nodes in the scene during the lifetime of the simulation. A node chooses a random destination similarly as in the *Random Waypoint Mobility Model* so it will travel towards the boundary of the simulation area. When this boundary is encountered, the node will stop for a period of time before starting again. This might not be valid in a real rescue situation since the scene can contain obstacles that need to be avoided. The *Random Point Group Mobility Model* associates each node to a group; each group has associated a logical centre, and this is used to calculate the movement of the group. The motion of the group characterises the motion of each node that is part of that group. Additionally, each node has its own reference point in the group around which it roams. Therefore, a node movement is a combination of the movement of the logical centre of the group and of the node's individual movement with respect to its reference point in the group. This model creates transient partitioned networks, mainly because nodes roam around their group reference points. This is not too realistic since team members might separate and join other groups.

In a real rescue situation, nodes have different movement patterns depending on groups of nodes and individuals. However, to the best of our knowledge, there are no publicly available movement traces of personnel in the disaster area during a real rescue operation or exercise. To overcome this situation one can use the mobility traces instrumented from an agent-based simulation environment such as *RoboCupRescue Simulation Project* [31]. It provides a generic urban disaster simulation environment. In this environment, intelligent software agents representing rescue teams of ambulances, police forces, and fire brigades coordinate and collaborate to save as many civilians as possible and to extinguish fires. We consider these traces to be as close as possible to traces from a real rescue operation. A limitation of the RoboCupRescue resides in the granularity of the teams, since one software agent represents a team. This implies that a member of a team cannot have independent movements or change teams.

Each of the aforementioned types of mobility traces has a set of drawbacks and cannot provide a complete test environment. The best solution for testing would be to use traces from the real exercises or operations. The mobility traces from real exercises have the advantage of realism, and the middleware can be tuned to adapt very well to that scenario. One needs also to consider a certain degree of randomness, since the layout of the scene and obstacles present may be very different at each accident site. Unfortunately, this is not possible now and the

only viable solution is to perform tests using both synthetic mobility models and simulated rescue operations.

3.5.2 Synthetic Communication Models

Another problem for middleware development and testing is data traffic. For development, this means what type of traffic, nodes will generate during a rescue operation. For testing, it implies finding the parameter limits of the possible data traffic that nodes will be able to generate during a rescue operation. Aschenbruck et al. [3] have analyzed the characteristics of the voice traffic during a rescue exercise involving the German public authorities. They have observed that most of the conversations (channel holding time) were shorter than 20 seconds and the time between communication was in average shorter than 10 seconds. They have concluded that the traffic in a disaster area has different properties than traffic in cellular networks.

Communication can be modelled as voice traffic, as suggested by Aschenbruck et al. [3]. Another choice is to use periodic communication with multiple transmission sources. The communication is periodic and it should produce a complete discovery of the network and keep the nodes connected. Another issue is the organisation of the communication flows, meaning flat or structured. In the *flat communication* models, one chooses a subset of the nodes, for example at random, to take the role of transmission sources; they transmit periodically data to all the nodes in the simulation. In *structured communication* models, one can use the inter-organisational hierarchies and intra-organisational interactions to shape the communication flows. Most communication takes place between the nodes in a group. One node plays the role of communication gateway and group leader. Additionally, group leaders of different groups communicate periodically with each other. This is similar to real life communication in rescue scenarios, i.e., team members report to the team/group leader, and group leaders coordinate their actions.

The parameters of the data traffic, i.e., upper and lower bound, are quite hard to determine since a Sparse MANET is by itself a best effort infrastructure with no guaranties. Therefore, to test the upper limit of the data traffic, one can inject as much data as possible in the network and let the underlining infrastructure limit the amount of data traffic possible. To find the lower limit one needs to limit the traffic created by the application, which can give us a good idea of the overhead created by the middleware.

3.5.3 Emulations and Simulations

In our simulations, we use two routing protocols: Ad Hoc On-Demand Distance Vector (AODV) [27]and Optimized Link State Routing (OLSR) [7].We have picked these protocols since they are two of the most developed and widespread MANET routing protocols. AODV is a reactive routing protocol, i.e., it discovers a route on a when-needed basis and maintains it for as long as it is used. It can also use hello messages to maintain local updated information. OLSR is a proactive routing protocol, i.e., it exchanges topology information with other nodes of the network regularly. The protocol relies on selected nodes to forward topology information during the flooding process. One of the advantages of both AODV and OLSR is that they support the concept of neighbourhood, by keeping track of nodes in one hop communication range.

4 A Framework Solution

The challenges presented are reflected in a number of requirements to the middleware. In the following, which is partly adopted from [22], we summarise the resulting requirements and present a framework for Sparse MANETs for rescue and emergency scenarios.

We need support for intra- and inter-organisational information flow and knowledge exchange, as well as means to announce and discover information sources. Contextual support enables applications to adapt better to particular scenarios and allow them to fine-tune according to spatial and temporal data. Profiling and personalisation can assist in filtering and presenting information in accordance with the needs of users and devices, as well as displaying their capabilities. The middleware should provide support for organisational structure and for creating groups on-the-fly. Security must be dynamic, enabling privileged users to grant group memberships at the rescue site, as well as to influence changes to the security regime when circumstances demand it. Communication must be available, reliable and efficient even in the presence of frequent network partitioning. There should be extensive support for resource sharing between devices, including ways to register and discover available resources of different types. To allow graceful degradation the middleware must do monitoring and be prepared for it. This leaves us with nine articulated requirements and goals for the middleware, which we list in the order of appearance: **Intra- and inter-organisational information flow**, **Service availability**, **Context management**, **Profiling and personalisation**, **Group- and organisational support**, **Dynamic security**, **Communication**, **Resource sharing**, and **Graceful degradation**. In addition, there is an ever-present need for **Data sharing and storage**. These requirements are addressed by six middleware *concerns*, which in the middleware architecture correspond to five components. The concerns constitute a foundation for a middleware framework covering the required services for Sparse MANETs in rescue scenarios:

- **Knowledge Management** – to handle ontologies, support metadata integration and interpretation;
- **Context Management** – to manage context models, context sharing, profiling and personalisation;
- **Data Management** – to cater for capabilities similar to those of distributed databases;
- **Communication Infrastructure** – for supporting distributed event notification, publish and subscribe services, and message mediation;
- **Resource Management** – to register and discover information sources and web services as well as resources available, to handle neighbour awareness, computation and application sharing, mobile agents, proxy and replica placement, and movement prediction;
- **Security Management** – for access control, message signing and encryption, supporting group- and organisational structure, group key assignment, and dynamic security services.

For an overview of the *concerns*, see Figure 2.

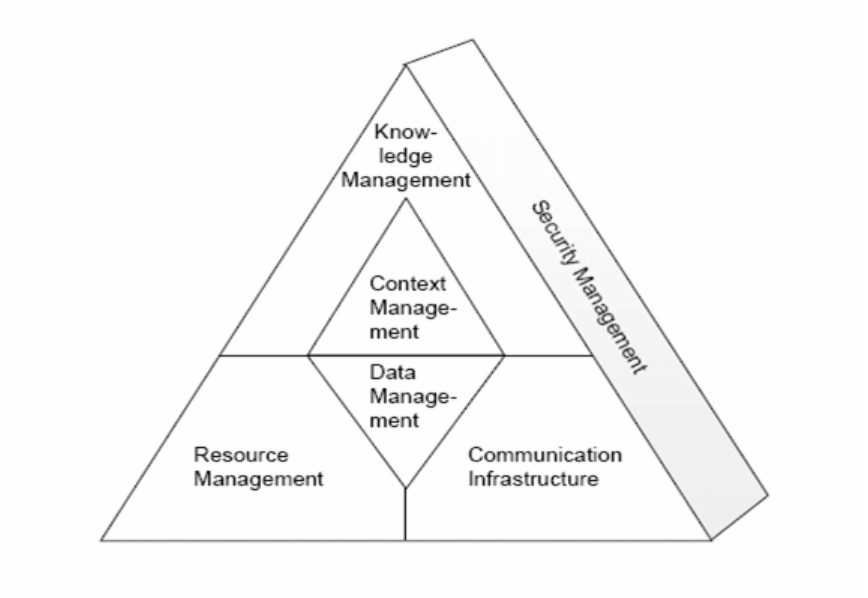

Figure 2: Middleware Concerns

The concerns correspond to architecture components in the following manner: Communication Infrastructure to the **Distributed Event Notification Service** and **Watchdogs** components; Resource Management to the **Resource Manager**; Security Management to the **Security and Privacy Manager**, and the concerns Knowledge Management and Context Management are handled by the **Knowledge Manager**. The middleware components all rely heavily on each other's services. Each component will be briefly presented in the following.

4.1 Knowledge Manager

This component corresponds to the concerns Knowledge Management and Context Management. The purpose of this component is to provide flexible services that allow relating metadata descriptions of information items to a semantic context and support management of knowledge sharing and integration in a rescue operation scenario. The Knowledge Manager offers support for the dissemination, sharing and interpretation of ontologies, and browsing and querying of ontologies and ontology contents. Thus, there is a need of distributed knowledge base functionality and a global view of what knowledge is available in the network. Issues that need addressing include *understanding* across domains and organisations through use of knowledge management techniques, avoiding *information overflow* through content filtering and personalisation, managing *availability* of information, metadata and ontologies, offering information *query and retrieval* services, and supporting *information exchange.*

Overall, we can differentiate required functionality into those handling the structure, content and meaning of information, and those that have a supportive role. The first group of services all have do with handling metadata at some level, ranging from data structure definitions/descriptions to conceptual definitions of some domain. This group of components we have termed *metadata handling components*, and include a Semantic Metadata and Ontology framework to deal with sharing and interpretation of ontologies, Data Dictionary Management for management of metadata in local and global data dictionaries, and Profile and Context Management supporting filtering and personalisation. The components handling the group of supportive services/functions are termed *tool components*. These are the Query

Management for querying ontology and metadata as well as retrieval of relevant information, and XML Parsing for information exchange in a standardised format.

We use the term global (distributed) dictionary in a more extended sense than the traditional, as our approach does not have a global conceptual schema. This is possible through 1) making the assumption that organisations have agreed upon a basic common ontology or vocabulary, as well as exchanged information about standards and data models in use in the organisation, a priori to the rescue operation; and 2) using a standardised format for information exchange. Such agreements and exchanges regarding vocabularies, data models and standards would take place in the *a priori phase* of the general rescue scenario phases described in 2.5.

The Knowledge Manager adopts the hierarchical view of knowledge, i.e., data, information, and knowledge seen as levels of increasing semantics. Our main concern is management and sharing of explicit knowledge, i.e., factual knowledge found for instance in documents, databases, files, and models. Tacit knowledge, e.g., in mental models, procedures and skills, is not handled, although sharing this kind of knowledge may be valuable in rescue operations. The knowledge management processes in focus are knowledge storage/retrieval and knowledge transfer/sharing. We do not address aspects of knowledge learning (creation) and application, as these are not directly relevant in the application scenario. An overview of different kinds of knowledge and knowledge management phases is found in [1].

In our approach for metadata management, metadata descriptions of information items for sharing are enriched with concepts from ontologies and vocabularies. Enhancing metadata with concepts from domain ontologies is an approach also used in Kashyap and Sheth [17] and in solutions for the Semantic Web [33]. As device resources, e.g., memory, processor, energy, and bandwidth, may be limited, simply propagating all information/data to all participants would be too expensive resource wise. By sharing extracts from the metadata descriptions, rather than full metadata descriptions, knowledge of available information resources can be spread through the network while saving resources like bandwidth and energy, thus contributing to a more effective use of available resources.

The creation and maintenance of ontologies will happen outside of the rescue operation itself. Thus, our use for ontologies is of already existing ontologies utilised in the following way: domain ontologies and/or vocabularies from relevant domains, e.g., medical domain; as support in sharing vocabularies, e.g., as a bridge or an upper level ontology; and for enhancing metadata descriptions with terms/concepts from the ontologies. The reasoning tasks needed during the operation can be limited to instance reasoning. Given the resource limitations of our scenario and depending on the context in an ongoing rescue operation, e.g., resources and capabilities on the currently available devices, it is possible that only a limited set of instance reasoning services can be offered. Powerful devices/laptops will be the only nodes able to offer such services, and smaller devices will have to request these services from these, as resource weak devices will not have the capabilities to perform such inference tasks.

4.2 Distributed Event Notification Service

The Distributed Event Notification Service (DENS) provides a publish/subscribe service and delay-tolerant delivery of notifications in case of, e.g., network partitioning. In the publish/subscribe service a subscriber subscribes for information, and the publishers publish information, independently. If DENS cannot deliver a notification to a subscriber, the service will store the notification and try to deliver the notification using the store-carry-forward paradigm. The main design goals for DENS are to support a flexible subscription model,

ensure high delivery ratio, as close to at-least-once semantics as possible, and use a-priori information and information collected at run-time to best configure the system.

Providing an asynchronous communication service makes sense because of the nature of sparse MANETs. Only providing synchronous communication would be too limiting because of communication disruptions, like when devices are suddenly out of reach or turned off. In addition, having such a service makes it easier for the application to receive information by just sending subscriptions to the service and getting notified when the event takes place, and therefore not having to find the information and communicating directly with the node.

In the described scenarios, there is a variety of kinds of new information or changes in information that could be of interest, and that could be regarded as events, both to rescue operation applications and other middleware service. These changes can happen at all kinds of nodes, meaning that any node in principle could be an information source. Some information is known a priori to be needed, for example to maintain the communication services it is necessary to gather information about size of network in terms of nodes and area, of available resources, etc, other subscriptions are made during the operation. To save resources, DENS filters events at the source, instead of sending all potentially interesting updates to the service. A monitoring agent filters events based on the subscriptions relevant to the local node.

The DENS supports different subscription languages at run-time. The reason for this is that different applications have different needs regarding the level of sophistication, and resource wise it would be best not to force all the applications to use the most complex language. With respect to the need of handling resources efficiently, simple languages are better than complex ones. In a rescue operation, individuals from different organisations cooperate by sharing information. Each organisation may have its own vocabulary and DENS therefore supports subscriptions that are targeting different vocabularies. This support must also include the translation or mapping between different vocabularies since information in rescue operations is shared also across different organisations. For example, the coordinators of fire fighters and police officers might be interested in the temperature in a certain room. They may use different terms for temperature even if the sensor supports only the fire fighters vocabulary.

In summary, the DENS supports the following:

- content-based filtering at source nodes,
- flexible definition of source nodes in subscriptions,
- multiple vocabularies, and
- multiple languages at run-time.

All nodes run part of the service, but some of the nodes are chosen to run the full-fledged service and act as mediators, keeping information as consistent and in an up-to-date state as possible. These nodes form an overlay. Information about subscriptions and notifications is exchanged in a gossiping fashion between these nodes, and this implements a delay-tolerant delivery of notifications in case ordinary unicast provided by the routing protocol is not possible. On the publisher nodes, there are monitoring agents to filter what is of interest. The size of the overlay, and the degree of replication used is dependent on the mobility scenario and the application scenario.

4.3 Resource Manager Component

The component for Resource Management aims at enabling best possible resource sharing among the devices involved in the network. During a rescue operation the involved personnel

has a very strong incentive to collaborate and cooperate across organisations. This requires them to share knowledge and resources in order to fulfil their tasks. In a resource constraint environment such as a rescue operation, a distributed application needs the help of a resource manager in order to make the best out of the available resources. A resource manager's main duties in such environments, are to register, discover services and data sources, and make the information available through the network. For this, each node can maintain a sharing profile with information about locally available resources and running services. The physical resources need to be frequently monitored, which can be achieved by using mechanisms provided by the operating system. Resource availability information can be disseminated in the network by using a shared data space as the one provided by the Data Management. Other alternatives are to announce availability of resources as notifications by using DENS, or to discover resources by querying the other nodes.

The dynamic nature of MANETs poses a set of challenges for a resource manager from the point of view of a traditional resource reservation, such as hard state and exclusiveness. This might not work because the system is too vulnerable to disruptions in communication. Such disruptions can be caused by the movements of the nodes and possible changes in the environment, which might lead to changes in the network topology, or even partition of the network. One solution is to use time-outs for soft-state resource reservation that can be valid for a specified period of time and without exclusive reservation of the resource. The period of time can be anything from the time needed by a process to use a resource, to the estimated time a resource can be available for access. The allocation of resources can use fuzzy rules when allocating resources to other nodes.

In MANETs, the most important resources are the network bandwidth and the energy of the mobile devices; it is the duty of the resource manager to preserve as much as possible of both of these resources. In general most of the energy of mobile devices is consumed during communication; therefore the resource manager should minimise the communication overhead of the middleware. This can be achieved by predicting future availability of resources, not only to establish meaningful time-outs for soft-state reservations but also to increase the availability of information and services through replication and graceful degradation. The problem of predicting resource availability can be approximated by predicting adjacency of nodes for the network topology. One approach is to analyze the physical location and the movement of the nodes with GPS information; however, GPS devices might not always work, for example in buildings, tunnels, or dense forests. An alternative solution is to use history information on neighbourhood relations in order to predict the future neighbourhood of the node, i.e., adjacency or non-adjacency of node. This can be used to predict network partitions or even the future network topology.

The neighbourhood awareness information can be used together with information about resource sharing profiles to create a neighbourhood resource-oriented context. The application and frameworks to achieve pervasive data storage and increase service and quality can exploit this context. The neighbourhood resource-oriented context can be used by services to detect and react to the imminent loss of resources, and to increase availability and achieve graceful degradation. Further or current resource-oriented context can be used by services to adapt their behaviour to the current available resources, i.e., increase, downscale, migrate, or even terminate the service. Additionally, if a service knows about alternative resources and the next time period with high probability of its availability, the service can survive by attempting decomposition of services and delegation of execution to nodes with available resources. This kind of service decomposition and mobility can be achieved by using different techniques,

e.g., mobile software agents or proxy execution. Service composition facilities can help applications make better use of available services by supporting the composition of complex services from more simple ones. They can also enhance service availability by reacting to a missing service by replacing it with a similar service provided by another node.

There are many security concerns with respect to information accumulation, i.e., on nodes, services, and resources availability and access, and access rights to resources and services. The concerns are mostly concerned with the trustworthiness and quality of the information, and possible misuse and access rights to it. These concerns need to be addressed together with the Security Management. However, the main duties of a Resource Management component are to facilitate remote access to resources by requesting, negotiating, reserving, and administrating remote access to resources. The access to the remote resources needs to respect security-related measures imposed by the Security Manager, such as access control, user traceability, and data integrity.

Resource Management aims at enabling best possible resource sharing among the devices involved in the network; this requires to gather and disseminate information on resources availability, and to facilitate resource access and sharing. The MANETs can support the collaborative environment required in a rescue and emergency intervention. Remote access to resources has the potential to improve availability of data and services and provide graceful degradation or migration for services. This is especially useful for applications running in a network with a number of resource-limited devices and the necessity to utilise all available resources. To keep track of available resources and services and enable resource sharing, a distributed solution is necessary.

4.4 Security and Privacy Manager

As depicted in Figure 2, the Security Management has a direct impact on the functionality of all the other components and therefore has to be considered from an early stage of development. The Security Management has to make sure that all the security requirements, some of which are mentioned in Section 3.2, are fulfilled during the other components' operation. In addition to being something every other component depends on, the Security Management itself depends on some of the other components, such as for key distribution, storage of keys and certificates, getting information on the neighbourhood, etc. This, however, poses additional security issues that need to be taken care of.

Security can be implemented in either the traditional layered approach or a more adaptive cross-layered one, both having their advantages and disadvantages. The layered approach, by putting clear borders to data flow, offers a high level of security. In our case, a more flexible approach might be a better choice, provided that it does not significantly weaken the overall security level. A lightweight adaptable cross-layer middleware solution, based for example on the reflection technique [20], [4], would allow middleware services to adapt to the heterogeneous dynamic environment. Examples of such architectures are Open ORB 2 and ReMMoC [12]. The programming language Obol [2] can be used to face the security issues, and with help of Aspect Oriented Programming [18], the cross-concern integration can be faced early in the development phase.

5 Conclusion

In this report, we have presented background information on rescue and emergency operations, example scenarios, resulting issues for consideration and requirement analysis for developing middleware that can support information sharing during rescue operations. The report is based on work done in the Ad-Hoc InfoWare project, which is aimed at information sharing in emergency and rescue operations. The report is also relevant for the MIDAS project. Through example scenarios and descriptions of characteristics of rescue operations, their organisation and structure, we have shown that emergency and rescue operations are highly organised and structured ventures, where personnel from many different organisations cooperate (towards a common goal) in a very hectic, dynamic, and possibly volatile environment. Another characteristic is the heterogeneity inherent in the diversity of devices used, organisations, communication lines (information flows), applications, and the rescue operations themselves, which poses challenges to developing a middleware system for this kind of environment. Other challenges include instable network and poor infrastructure, and sensitive information. The three example scenarios presented illustrate accidents of different size and impact, and the settings are as different as an inner city earthquake, a railway accident in a remote mountain area, and an underground fire in the Paris metro. Similarities and differences found through analysis of these scenarios are used as basis for the requirement analysis and as being configurable functionality respectively.

The challenges and requirements analysis was presented from five different perspectives; organisational, security, network, communication, and application development constraints. Following is a brief summary of the challenges. The participation of several organisations, with different focus, information needs, standards and policies, creates challenges to support interoperability regarding syntactic, structural, and semantic heterogeneity, as well as some intra- and inter-organisational structure. To allow a more efficient use of limited resources and reduce problems of information overload, sharing (meta-) information indicating where to find relevant information and making use of filtering and context awareness can be beneficial. Several challenges related to security arise when using wireless networks, particularly considering the presence of sensitive information. External and internal attacks require mechanisms and policies for different aspects of security like authenticity, integrity, and confidentiality. In addition, different security requirements and policies in organisations may introduce extra demands related to confidentiality. The diversity in devices and dynamic topologies in sparse MANETs and delay tolerant networks requires non-intrusive communication, position independency, and protocol diversity. In addition, different kinds of user mobility further complicate communication issues. A reduction in communication overhead and demands for specific functionality from devices is necessary, and the middleware should exploit information extractions from routing protocols. Due to blocking, e.g., by buildings or tunnels, asynchronous communication is needed, as provided by the publish/subscribe paradigm. In addition, there is a need for “routing in time”, i.e., store-carry forward or epidemic routing. Demands for different subscription languages means the middleware should support a variety of languages, as well as mapping between these. Application and middleware development for rescue and emergency operations faces a number of constraints related to nodes’ movement on the scene and communication needs and patterns. Hence, it is important to test if solutions handle system dynamicity through different mobility models. A testing environment will benefit from a network with multiple transmission sources, using different communication models and routing protocols.

Finally, the requirements resulting from the challenges were presented as a set of general requirements or goals for the middleware; intra- and inter-organisational information flow, service availability, context management, profiling and personalisation, group- and organisational support, dynamic security, communication, resource sharing, graceful degradation, and data sharing and storage. These requirements are addressed by a set of middleware components comprising the Ad-Hoc InfoWare architecture. These are the Distributed Event Notification Service and Watchdogs components dealing with communication infrastructure; Resource Manager handling resource management; *Security* and Privacy Manager for security management, and Knowledge Manager taking care of knowledge management and context management.

References/Sources

[1] Alavi, M., and Leidner, D., "Knowledge Management and Knowledge Management Systems: conceptual foundations and research issues"; MISQuarterly Vol. 25 No.1, pp.107-136, March 2001

[2] Andersen, A. et al..,"Reflective Middleware and Security: OOPP meets Obol", in Proceedings of the 2nd Workshop on Reflective and Adaptive Middleware, Rio. June 2003.

[3] Aschenbruck, N., Frank, M., Martini, P, and Tölle, J., "Human Mobility in MANET Disaster Area Simulation - A Realistic Approach", The 29th Annual IEEE International Conference on Local Computer Networks (LCN'04); pp. 668-675, 2004.

[4] Blair, G. S. et al.,"The design and implementation of Open ORB 2", in IEEE Distributed Systems Online, 2(6), 2001.

[5] Carver, L. and Turoff, M., "Human-Computer Interaction: The Human and Computer as a Team in Emergency Management Information Systems", Communications of the ACM March 2007, Vol 50, No. 3, pp. 33-38. March 2007.

[6] CEN/TC 251, European Standardization of Health Informatics, http://www.centc251.org/

[7] Clausen, T., and Jacquet, J, "RFC 3626:optimized link state routing protocol (olsr)," Oct. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3626.txt

[8] Davies, V., "Evaluating mobility models within an ad hoc network,", Master's thesis, Colorado School of Mines, 2000.

[9] Earthquake scenario for western Nevada, http://www.nbmg.unr.edu/dox/nl/nl30.htm.

[10] Eugster, P. T., Felber, P.A., Guerraoui, R., and Kermarrec, A-M., "The many Faces of Publish/subscribe", ACM COmpuing Surveys, Vol. 35, No. 2, pp. 114-131. June 2003.

[11] Gorman, J.. "The MIDAS Project: Interworking and Data Sharing.", Interworking 2006, Santiago, Chile, January 2007.

[12] Grace, P., Blair, G. S. and Samuel, S., "Interoperating with Services in a Mobile Environment", Technical Report (MPG-03-01), Lancaster University, 2003.

[13] Health Level Seven (HL7), http://www.hl7.org/

[14] Hong X, Gerla, M, Pei, G, and Chiang, C., "A group mobility model for ad hoc wireless networks," in MSWiM '99: Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems, Seattle, Washington, USA, 1999, pp. 53–60. TC6 World Computer Congress, 2004, Toulouse, France, August 2004.

[15] Jernbaneverket, http://www.jernbaneverket.no/

[16] Kärpijoki, V., "Security in Ad Hoc Networks", Tik-110.501, Seminar on Network Security, HUT TML 2000.

[17] Kashyap, V., and Sheth, A. "Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies", 1996/1998.

[18] Kiczales, G. et al., "Aspect-oriented programming", in Proceedings of the 11th European Conference on Object-Oriented Programming (ECOOP'97), LNCS 1241, Springer Verlag, pp. 220-242, Jyväskylä, Finland, June 1997.

[19] Isabelli, M., "Application Scenario - Emergency", MIDAS Project Deliverbale, www.ist-midas.org, 2006.

[20] Maes, P., "Concepts and experiments in computational reflection", in OOP-SLA'87: Conference proceedings on Object-oriented programming systems, languages and applications, ACM Press, Orlando, Florida, USA, 1987, 147.

[21] Mendonça, D. , Jefferson, T., Harrald, J.,"Collaborative adhocracies and mix-and-match technologies in emergency managment". ACM .. Vol. 50, No. 3. 2007.

[22] Munthe-Kaas, E., Drugan, O., Goebel, V., Plagemann, T., Pužar, M., Sanderson, N., Skjelsvik, K. S., "Mobile Middleware for Rescue and Emergency Scenarios, Mobile Middleware". Paolo Bellavista and Antonio Corradi, ed., CRCPress, 2006.

[23] Navidi, W., Camp T., and Bauer, N. “Improving the accuracy of random waypoint simulations through steady-state initialization,” in Proceedings of the 15th International Conference on Modeling and Simulation (MS ’04), Marina Del Rey, CA, USA, March 2004.

[24] Norwegian Centre for Informatics in Health and Social Care (KITH), http://www.kith.no/

[25] The Norwegian Medical Association. (The Norwegian Index for Medical Emergencies). Norsk indeks for medisinsk nødhjelp. Stavanger: Aasmund S. Laerdal A/S, 1994.

[26] The Norwegian SAR Service: http://odin.dep.no/filarkiv/183865/Infohefte_engelsk.pdf (norsk: http://odin.dep.no/filarkiv/183864/Infohefte_norsk-lang.pdf)

[27] Perkins, C., Belding-Royer, E., and Das, S. “RFC 3561: Ad hoc on-demand distance vector (aodv) routing,” July 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[28] Plagemann, T., Goebel, V., Griwodz, C., and Halvorsen, P. “Towards Middleware Services for Ad-Hoc Network Applications”. In the 9th IEEE Workshop on Future Trends of Distributed Computing Systems, San Juan, Puerto Rico, May 2003, pp. 249-257.

[29] Rapport fra Åsta ulykken: NOU 2000: 30, Åsta-ulykken, 4. januar 2000, ISBN 82-583-0543-3, Oslo 2000. http://odin.dep.no/jd/norsk/dok/andre_dok/nou/012001-020007/hov003-bn.html

[30] Ratsimor, O. et al., “Allia: alliance-based service discovery for ad-hoc environments”, in Proc. of the 2nd ACM Mobicom Int. Workshop on Mobile Commerce (WMC’02), Atlanta, GA, September, 2002, pp. 1–9.

[31] R. S. Project, “The RoboCupRescue simulation project,” 2005. [Online].Available: http://www.robocup2005.org/roborescue

[32] Royer, E, Melliar-Smith, P, and Moser, L., “An analysis of the optimum node density for ad hoc mobile networks,” in IEEE International Conference on Communications, vol. 3. Helsinki, Finland: IEEE Computer Society. June 2001, pp. 857–861.

[33] Stuckenscmidt, H. and van Harmelen, F., “Information Sharing on the Semantic Web“, Springer-Verlag Berlin Heidelberg 2005, ISBN 3-540-20594-2.

[34] Vaardal, B. , Lossius, H.M., Steen, P.A., and Johnsen, R. “Have the implementation of a new specialised emergency medical service influenced the pattern of general practitioners involvement in pre-hospital medical emergencies? A study of geographic variations in alerting, dispatch, and response.” Emergency Medical Journal 2005;22;216-219.

[35] Vahdat, A., and Becker, D. “Epidemic Routing for Partially Connected Ad Hoc Networks”, Technical Report CS-2000-06, Department of Computer Science, Duke University, 2000.

[36] Westrheim, V., Praktisk beredskap i norsk jernbanedrift, http://www.sikkerhetsdagene.no/Tidligere%20konferanser/2006/Westrheim.pdf

[37] Zhao, W., Ammar, M., and Zegura, E. “A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Netowrks”. MobiHoc’04. Roppongi, Japan. 2004.