DEGREE PROJECT IN COMPUTER ENGINEERING,
FIRST CYCLE, 15 CREDITS
STOCKHOLM, SWEDEN 2016

Integrating Monitoring Systems - Pre-Study

MARCUS BLOM

KIM HAMMAR

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Abstract

Failures in networks that reside in business environments cause harm to organizations depending on them. Every minute of inoperativety is hurtful and as a network adminstrator you want to minimize the rates of failures as well as the time of inoperation. Therefore, a fruitful network monitoring system is of great interest for such organizations. This bachelor's thesis is the outcome of a pre-study performed on behalf of MIC Nordic and sought to advice them in the implementation of a new monitoring system.

The aim of this study was to investigate how a Network Operation Center (NOC) can be implemented in an existing monitoring environment, to integrate current monitoring systems to a central point for monitoring. This study takes an holitstic approach to network management and the research can be divided into two main categories: *Communication between network components* and *Presentation of information*. Our process involves an analysis of the environment of MIC Nordic and an in depth inquiry on the current state of network monitoring and interface design. The study then culminates in the implementation of a prototype. The prototype serves in first hand as a research tool to collect experience and empirical evidence to increase the crediblity of our conclusions. It is also an attempt of demonstrating the complete process behind developing a NOC, that we believe can fill a gap among the previous research in the field.

From our results you can expect a prototype with functionality for monitoring network components and a graphical user interface (GUI) for displaying information. The results are designed towards solving the specific network management problem that was given and the environment that it concerns. This pre-study suggests that the best solution for implementing a NOC in the given environment is to use SNMP for communication. From an investigation on how to present network management information in a effective way we propose to follow a user-centered approach and to utilize human perception theory in the design process. The authors recommend further research that maintain the holistic approach but applies more quantitative methods to broaden the scope.

Keywords— Network Management, NOC, Pre-study, GUI, SNMP, User-centered design, Human Perception Theory, MIC Nordic

Abstrakt

Störningar i nätverk som används i företagsmiljöer skapar problem för organisationer som är beroende av dess funktion. Varje minut som nätverket är verkningslöst är ofördelaktigt och som nätverksadministratör så vill du minimera antal störningar och tiden då nätverket är verkningslöst. Ett effektivt nätverksövervaknings system är därför av stort intresse för organisationer beroende av ett funktionerande nätverk. Den här rapporten är resultatet av en förundersökning som utfördes på uppdrag av MIC Nordic, för att ge en rekommendation om hur ett nytt övervakningssystem för deras nätverk kan implementeras.

Målet med studien var att undersöka hur ett Network Operation Center (NOC) kan implementeras i en existerande miljö för att integrera nuvarande övervakningssystem till en central punkt för övervakning. Den här studien tar ett holistiskt grepp om nätverksövervakning och undersökningen kan delas in i två primära kategorier: *Kommunikation mellan nätverkskomponenter* och *Presentation av information*. Vår process involverar en analys av MIC Nordics miljö och en djupgående utredning om nätverksövervakning samt gränssnitts design. Studien kulminerar i en implementaion av en prototyp. Prototypen är i första hand ett undersökningsverktyg för att samla på oss erfarenhet och empiriska belägg för att öka trovärdigheten i våra slutsatser. Prototypen utgör även ett försök av författarna att demonstrera den kompletta proceduren av att utveckla en NOC, avsikten är att det kan fylla ett behov bland tidigare avhandligar i ämnet.

Resultatet innehåller en prototyp med funktionalitet för att övervaka nätverkskomponenter samt ett grafiskt gränssnitt för att visa informationen. Resultaten är designade mot en lösning som är specifik för problemet som gavs av MIC Nordic och deras miljö. Denna förstudie proponerar att den bästa lösningen för att implementera en NOC i den givna miljön är att använda SNMP för kommunikation. Efter en granskning av hur man kan presentera information som rör nätverksövervakning på ett effektivt sätt så lanserar författarna en användarcentrerad metod som utnyttjar läran om hur människor uppfattar saker och ting. Författarna uppmuntrar vidare undersökningar som bibehåller det holistiska greppet men som applicerar mer kvantitativa metoder för att utöka undersökningens omfattning.

Nyckelord— Nätverksövervakning, NOC, Förstudie, GUI, SNMP, Användarcentrerad design, Läran om hur människor uppfattar saker och ting, MIC Nordic

Acknowledgements

We would first and foremost like to express thanks to our advisor Fadil Galjic of the Royal Institute of Technology. Galjic consistently provided us with his feedback and suggestions on the writing of this thesis, which allowed it to become what it did.

We also want to show our gratitude towards MIC Nordic, for giving us the opportunity to carry out this study, and for being confident in our proficiency. In particular we would like to thank Tord Sjölund and Parsova Khayatan, our supervisors at MIC Nordic, who have supported us througout this process.

Contents

List of Figures

List of Tables

Acronyms

ASN.1 Abstract Syntax Notation One. 14, 34, 36, 47, 52

BER Basic Encoding Rules. 14, 34, 36, 52

CGI Common Gateway Interface. 37

CMIP Common Management Information Protocol. 15, 35, 36, 62

CoAP Constrained Application Protocol. 14, 34–36, 62, 63

CORBA Common Object Request Broker Architecture. 16, 21, 36, 37, 40, 62

EMANICS European Network of Excellence for the Management of Internet Technologies and Complex Services. 19

FIFO First In First Out. 56

GUI Graphical User Interface. x, 4, 7, 16–18, 22, 25, 29, 30, 43–46, 49–52, 56–59, 64, 65, 69, 70

HTTP Hypertext Transfer Protocol. 54

HTTPS Hypertext Transfer Protocol Secure. 54

IDL Interface Definition Language. 16, 36

IEEE Institute of Electrical and Electronics Engineers. 19

IETF The Internet Engineering Task Force. 10, 14, 15, 45

IP Internet Protocol. x, 9, 10, 14, 15, 34, 57

IRTF Internet Research Task Force. 19

ITU-T Telecommunication Standardization Sector. 7, 41

M2M Machine to Machine. 14, 34

MD5 Message-Digest Algorithm 5. 37

MIB Management Information Base. x, 10, 12, 13, 34, 42, 45

NETCONF Network Configuration Protocol. 15, 21, 35, 36, 39, 40, 62, 63

NMRG Network Management Research Group. 19

NMS Network Management Station. 9

NOC Network Operations Center. 2–4, 9, 16, 25, 29, 33, 35–37, 40–47, 50–52, 54–66, 68–70

OID Object Identifier. 12, 13, 52

OMC Operations and Maintenance Centre. x, 2–4, 8–11, 27, 33, 35, 37, 38, 41–43, 45, 46, 54–58, 62–64

OSI Open Systems Interconnection. 15, 41

PDU Protocol Data Unit. 13, 14

REST Representational State Transfer. 34

RFC Request For Comments. 10, 12–14, 37, 38

RPC Remote Procedure Call. 15, 36

SHA Secure Hash Algorithm. 37

SNMP Simple Network Management Protocol. x, 9–15, 21, 22, 33–42, 47, 50–54, 58, 59, 61–64

SQL Structured Query Language. 54

SSH Secure Shell. 36, 39

TCP Transmission Control Protocol. x, 9, 10, 14, 15, 36, 38–40

TMN Telecommunications Management Network. 41, 42

UDP User Datagram Protocol. 9, 10, 13–15, 33, 34, 36, 38–40, 47, 51, 52

UI User Interface. 2, 16

URL Uniform Resource Locator. 34, 47

XML Extensible Markup Language. 15, 36, 40

Chapter 1

Introduction

With the increased use of interconnected wireless devices, capable of exchanging information over networks, it becomes a complex task to monitor larger networks. This is where specialized monitoring systems come in hand. The objectives of such systems is to enable users to monitor the status of the network in real-time and to provide customizable tools for configuring and proceeding on incoming alarms.

This demands reliable communication between networking elements that are being supervised and network management stations collecting the information. Furthermore, this also introduces the concern of how the collected information should be presented to the user in order to ensure a high readability of the network status. This thesis presents the task of developing a monitoring system and explores how numerous smaller networks can be monitored as a totality. The rest of this chapter introduces the specific problem that motivates this pre-study and defines the focus and purpose of this thesis.

1.1 Background

1.1.1 MIC Nordic

MIC Nordic's business idea is to provide the market with products and solutions for effective wireless communications in multi indoor environments[1]. The services that MIC Nordic sells include, among other things, monitoring, support, installation and deployment of systems.

[1]MIC Nordic, 2016.

1.1.2 Problem Background

With the services that MIC Nordic supply to their customers, they need systems to supervise the status of different networks that are on-site at their customers locations. They need tools and services for monitoring and receiving alarms from various on-site network elements in case of failure. At present, MIC Nordic monitor their network elements with a collection of Operation and Maintenance Centers (OMC), each responsible for monitoring a certain network containing a set of network elements.

Every OMCs has its own user interface (UI). The OMCs are connected to a network switch that allows for the user to work with one specified OMC at a time from a centralized workstation, see figure 1.1.

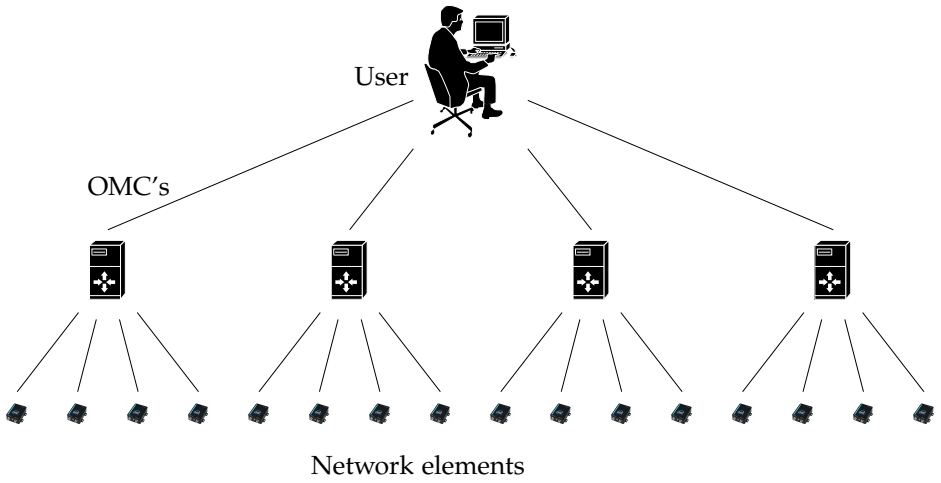

Figure 1.1: Current monitor setup.

MIC Nordic would like to have the information from each OMC be available in one system, a Network Operations Center (NOC), that could be accessed from more than one workstation, see figure 1.2. This setup would make for a solution that is more manageable and allows for scalability. A successful implementation of a NOC would enable MIC Nordic to manage their network more efficient, which would contribute to a more sustainable development for the future.

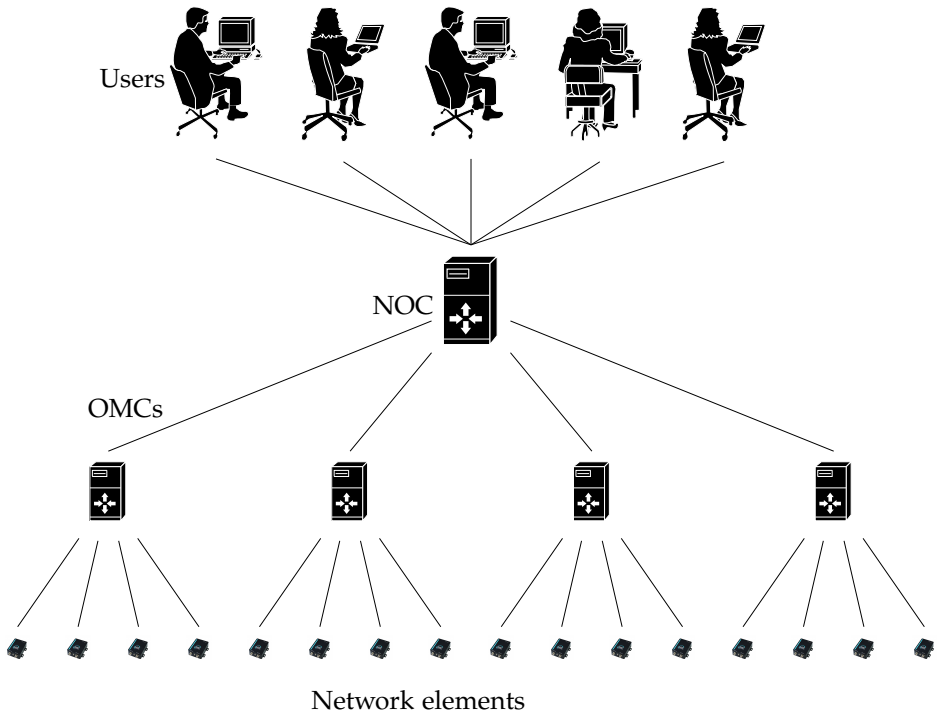

Figure 1.2: The desired monitor hierarchy.

1.2 Problem Statement

The problem can be defined by three underlying research questions.

- *How can four different Operation and Maintenance Centers (OMC) communicate with a Network Operations Center (NOC) and how can that NOC communicate with the four different OMCs?*

- *How can the communication between the NOC and the OMCs be done in a secure way?*

- *How can information from four different Operation and Maintenance Centers (OMC) be presented in a Graphical User Interface (GUI) and how can the GUI allow for manipulation of each OMC in a user-friendly and responsive way?*

1.3 Our Approach

This inquiry is based upon a qualitative approach. In the process of answering the problem statement, analytical research, interviews and prototyping are applied. Analytical research was carried out to acquire a perception on the current state of network management. Interviews were performed as a research on future users and lays a foundation for a user-centered design. A prototype was then developed to obtain real experience with the technologies that the problem statement concerns. The prototype is, although being narrowed to certain technologies, a way of collecting empirical evidence to justify the beliefs and understandings acquired through the analytical research.

1.4 Purpose

This pre-study aims at finding the best solution for integrating a specific set of monitoring systems that are in use at MIC Nordic to a NOC. The experiences from this study also aspire to lay a foundation for a more generic approach of integrating monitoring systems, with the intent of benefiting the industry at large.

1.5 Delimitations

There are many diverse types of monitoring systems and possible platforms for operating them. This study is delimited to finding solutions for integrating monitoring systems of the character that are in use at MIC Nordic.

1.6 Thesis Outline

In chapter 2 the reader is introduced to the problem area and necessary background information is presented, as well as related sources. In chapter 3 our research methodology is summarized and a brief description is given on how each method was carried out. Chapter 4 contains an analysis of the analytical research and related sources. In chapter 4 you will find the authors combined understandings and interpretations of related sources and how they stand against each other in relation to the problem statement. Chapter 5 contains the empirical results collected from building the prototype. Chapter 5 lay out the design decisions made and motivates

them in light of the results. In chapter 6 we introduce a discussion on the implications of our findings and we also present our own opinions. In Chapter 7 we summarize our research and comments it. Chapter 7 also contains suggestions for future research.

Chapter 2

Background information

A basic understanding of the technical concepts involved in network monitoring is a prerequisite to fully understand this document. This chapter covers the theoretical foundation as well as a few more advanced topics.

2.1 Technical background

2.1.1 Network Monitoring

2.1.1.1 Network

Network is a wide term for a collection of intercollected elements. In this thesis a network refers to a set of interconnected equipment that MIC Nordic uses to improve the in-building radio coverage in a definite area. This equipment typically consists of repeaters and alike devices.

2.1.1.2 Network Management

Alexander Clemm defined the term network management in his book "Network Management Fundamentals" as:

> *Network management refers to the activities associated with running a network, along with the technology required to support those acitivities. A significant part of running a network is simply monitoring it to understand what is going on, but there are also*

other aspects[1].

There are different types of network management systems, e.g. performance analysis systems and alarm management systems. In this study a network management system refers to an alarm management system.

2.1.1.3 Alarm Management System

Alarm management systems are specialized in collecting and monitoring alarms from the network[2]. A key aspect of alarm management systems is to enable users to rapidly process and make sense of the events and alarm messages that the system have received from the network.

2.1.1.3.1 Alarm Content

Alexander Clemm describes in "Network Management Fundamentals"[3] that every alarm is an indication of an underlying condition, and that it is a way of communicating unexpected events that occur. The content of an alarm is implementation-specific but typically adheres to a standard, termed X.733[4], defined by the Telecommunication Standardization Sector (ITU-T) standards organization. X.733 defines a rich set of standardized parameters that can be used in the alarm content. Usually you don't have a use for all of the parameters but rather pick out a few, some of the most common ones are listed in table 2.1.[5].

Parameter	**Purpose**
Event type	Categorize the alarm
Event information	Notification specific information
Probable cause	Probable cause of the alarm
Specific problems	Identifies further refinements to the probable cause
Perceived severity	Perceived impact on the object affected

Table 2.1: Alarm content following the X.733 Standard

Perhaps the most significant parameter for this study is severity, since it strongly relates to the presentation of alarms in a GUI. X.733 states the

[1]Clemm, 2006.
[2]Clemm, 2006.
[3]Clemm, 2006.
[4]Chisholm and D.Romascanu, March 2001.
[5](ITU), 1992.

following list of possible alarm severity labels, in order from least severe to most:

1. Cleared 2. Indeterminate 3. Warning 4. Minor 5. Major 6. Critical

2.1.2 Components For Monitoring Networks

2.1.2.1 Operation and Maintenance Center

An Operation and Maintenance Center (OMC) is a location from where you can operate, monitor and maintain a network. The actions that might be done from an OMC consists of security management, configuration tasks and administration. In the context of this study an OMC refers to a server where network equipment is supervised by MIC Nordic. The OMC interconnects analogous network-units and enables a single location for monitoring. The way of presenting information of an OMC is solely up to the implementation. At MIC Nordic graphical interfaces are used, see figure 2.1 for an example of such an interface.

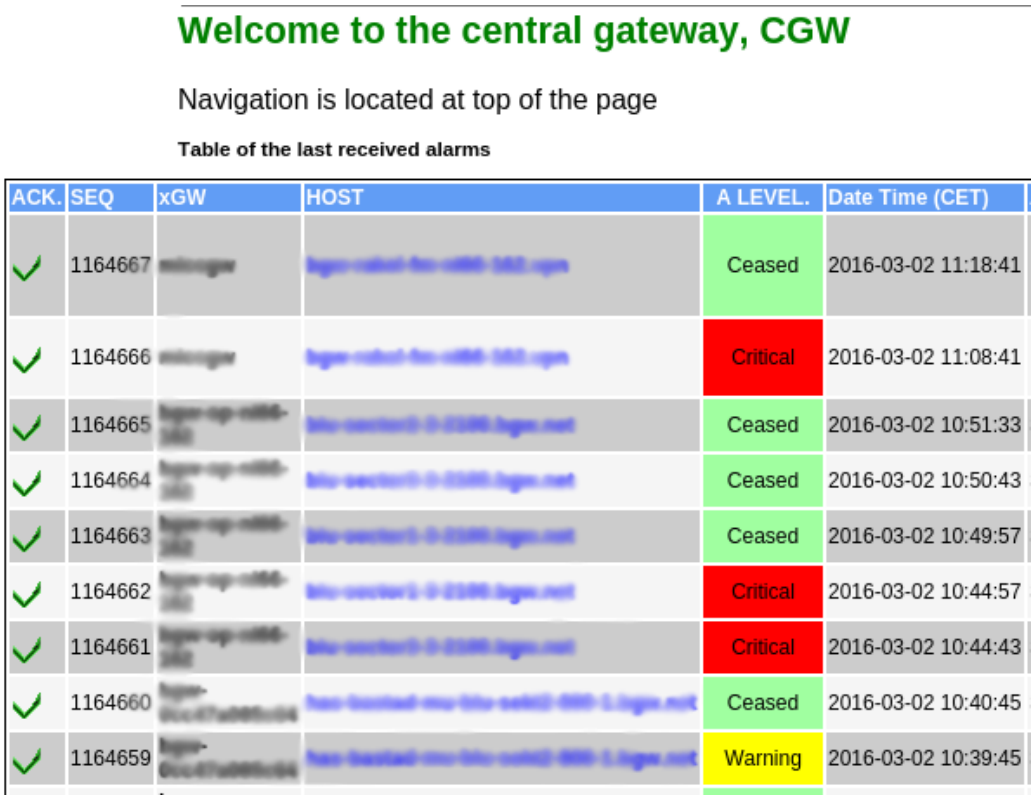

Figure 2.1: Screenshot from one of MIC Nordics OMCs

2.1.2.2 Network Operations Center

Network elements and management stations are all that are necessary to make the network management operate from a technical stand point. To make the management process more accessible and organized, a Network Operation Center (NOC) is required. A NOC is a place where networks can be monitored and managed. NOCs are in particular useful for maintaining and supervising networks at a larger scale, typically a NOC is used to enable a single point for monitoring distributed networks. While a NOC might also include a physical multi-hardware monitoring setup, this study refers to a NOC as a individual network component that connects several OMCs to a single location from where they can be managed jointly.

2.1.2.3 Network Management Station

In this thesis a Network Management Station (NMS) refers to some station that manages a set of network components. It can be an OMC or a NOC, it depends on the context.

2.1.3 Communication between network components

2.1.3.1 TCP/IP

The generic term "TCP/IP" usually means anything and everything related to the specific protocols of Transmission Control Protocol (TCP) and Internet Protocol (IP). It can include other protocols, applications and even the network medium[6]. A sample of these protocols are UDP and SNMP.

TCP/IP is usually refered to as a "stack"; this terminology comes from its logical structure with layered protocols, that maps to the layers that a computer uses to communicate over internet. This structure is convenient since it lets us specify exactly in which layer each of the protocols we're using resides. The TCP/IP stack can be seen as five layers stacked on each other.

[6]T. Socolofsky, January 1991.

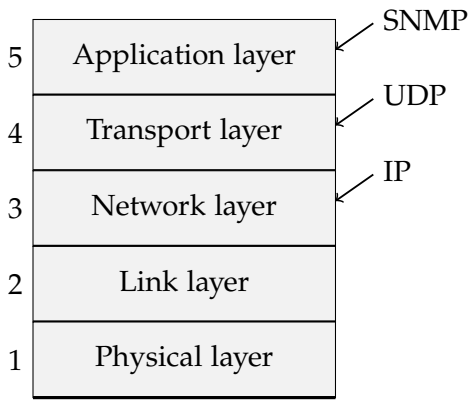

Figure 2.2: TCP/IP stack

2.1.3.2 User Datagram Protocol

User Datagram Protocol (UDP), defined in RFC 768[7], is a connectionless protocol for transmitting data in packages, called datagrams.

2.1.3.3 Simple Network Management Protocol

Simple Network Management Protocol, abbreviated to SNMP, is a protocol used to monitor and manage networks based on the TCP/IP stack. SNMP is defined by the Internet Engineering Task Force (IETF) and is an application-level protocol, specifically designed for network equipment to send alarms and status messages over a network using the UDP transport protocol. SNMP was first introduced in 1988, but is still very influential in network monitoring today. The SNMP architecture consists of network management stations and network elements, where the SNMP protocol is used to communicate information between the management stations and the elements[8]. The behavior of an entity using the SNMP protocol is defined in a Management Information Base (MIB)[9].

2.1.3.3.1 SNMP Agent

In order for the network elements and OMCs to send *SNMP traps* and respond to queries, they need to run a program called an *SNMP agent*, which is a program that can gather data about a piece of hardware and package

[7]Postel, August 1980.
[8]Case et al., May 1990.
[9]Mauro and Schmidt, 2005.

it into predefined SNMP-messages. In the context of this thesis, the SNMP agent software is run on each individual OMC. The data that the agent gathers from the device is defined in the Management Information Base (MIB).

2.1.3.3.2 SNMP Comunication

There are two ways for the management station to obtain the information that SNMP agents manages. One way is through *polling*, which means that the management station explicitly "asks" the agent for the information. This is done by sending a request defined by the SNMP-protocol, which the agent then responds to, see figure 2.3. The second approach is through event based communication using SNMP Traps.

An SNMP Trap is a signal that the SNMP Agent sends to a management station, see figure 2.4. Typical situations for sending SNMP Traps is when specific events occur or on predefined time intervals. Keep in mind that polls and traps can happen at the same time, there are no restrictions on when the management station can query the agent or when the agent can send a trap[10].

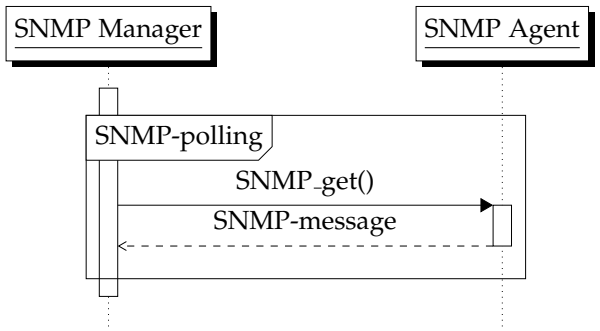

Figure 2.3: SNMP-polling communication

[10]Mauro and Schmidt, 2005.

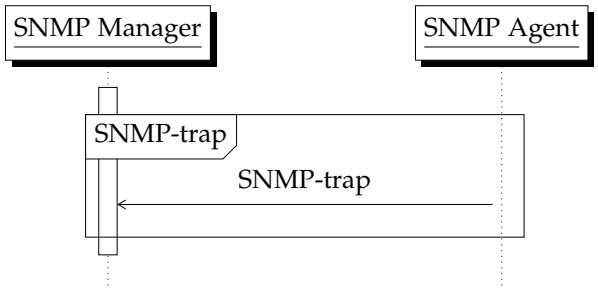

Figure 2.4: SNMP-trap communication

2.1.3.3.3 Management Information Base

As mentioned in previous sections, the purpose of monitoring systems is to display the status of network equipment that is in use in the network. This introduces the undertaking of defining what should be monitored.

When using the manager-agent model and a protocol like SNMP, this is defined in something termed Management Information Base (MIB). The MIB can be thought of as a database of managed objects that the SNMP Agent tracks[11]. Every SNMP Agent maintains an MIB. The MIB describes the properties and parameters of the device that is being monitored. Generally, network devices with support for SNMP have predefined MIBs constructed by the vendors, that contains a standard set of control values representing the status of the device. Any sort of status or information about the device that can be accessed by a monitoring-system is defined in the MIB[12]. In a way we can view the MIB as a set of answers to the questions that the monitoring systems can ask the SNMP agent.

MIBs are structured in a tree-like manner where each definition in the MIB is represented by a node in the tree. Nodes in the tree are named relative to their position in the tree, this name is called *Object Identifier* and identifies the definition in the MIB. There are standard MIBs for many use-cases, specific MIBs for the SNMP protocol can be found in RFC 3418[13].

2.1.3.3.4 Object Identifier

An SNMP agent contains different properties (managed objects) in its MIB. When those properties are included in SNMP-messages they need to be

[11] Mauro and Schmidt, 2005.
[12] Mauro and Schmidt, 2005.
[13] J. Case and Waldbusser, December 2002.

identifiable. This introduces the concept of *Object Identifiers* (OID). An OID is a unique identifier of a managed object. A managed object generally corresponds to a property of the device.

OIDs are constructed from a series of integers based on the nodes in a conceptual tree, and when presented, those integers are separated by dots[14]. If we need to identify a specific object, we form the OID by taking the value of each node from the root down to the desired object. For an example of this see figure 2.5. For a MIB with the hierarchy as depicted in the figure, the property "private" would have an OID of 1.3.6.1.4

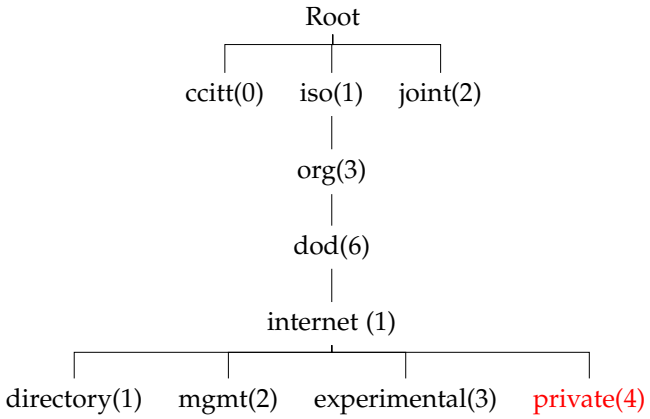

Figure 2.5: Illustrative MIB-tree

2.1.3.3.5 SNMP Messages

Messages between entities using the SNMP communication protocol is represented within a single UDP datagrams. This is sufficient for networks where alarms of smaller format are of interest, but is a restraint that makes SNMP inadequate for polling large volumes of data. Each message includes, apart from meta-information such as version number and similar, also an SNMP defined data unit. As described in RFC 1157[15], the different message types within the SNMP-protocol are divided into five Protocol Data Units (PDU).

- get-request
- get-next-request
- get-response

[14] Mauro and Schmidt, 2005.
[15] Case et al., May 1990.

- set-request
- trap

PDU describes what type of message it is. This study has mostly concerned itself with the trap PDU type.

2.1.3.3.6 Encoding and Representation of SNMP Messages

For SNMP messages to be understood by any SNMP device, a standard needs to exist for how these messages should be encoded and decoded. If no established standard was followed and instead softwares dealing with SNMP messages used the data-types of their programming language of choice, (e.g. Java data-types and their representation), then there is no guarantee that a SNMP-software developed in another language can understand the message.

SNMP uses the Abstract Syntax Notation One (ASN.1) standard to define the data types[16], and the Basic Encoding Rules (BER) that ASN.1 includes to define how a SNMP message should be encoded and decoded when transmitted over a transport medium such as Ethernet. SNMP uses only a subset of the ASN.1 standard for the sake of simplicity[17].

2.1.3.4 Constrained Application Protocol

The Constrained Application Protocol (CoAP) is a specialized web-transfer protocol for use with constrained nodes and constrained networks[18]. CoAP is designed for machine-to-machine (M2M) interactions, meaning that it's a web protocol that aims to face the challenges that come with that type of interactions, e.g. low memory usage and low power usage.

There are many similarities between CoAP and SNMP, both resides in the application layer in the TCP/IP stack (see figure 2.2) and both use UDP as their primary transport protocol. CoAP is in this context a modern protocol, approved and specified in an RFC developed by IETF, in late 2013, compared to SNMP who was first introduced in 1988.

[16] Bruey, 2005.
[17] Case et al., May 1990.
[18] Z. Shelby, June 2014.

2.1.3.5 Common Management Information Protocol

The Common Management Information Protocol (CMIP) is the Open Systems Interconnection (OSI) specified network management protocol, described in RFC 1095[19] and 1189[20]. CMIP was designed in competition with SNMP just a few years after SNMP's commotion and you could say that CMIP is to OSI what SNMP is to TCP/IP. Being that CMIP was designed for the OSI model, it has more features than SNMP and is considered more complete, which is why many expected CMIP to replace SNMP. CMIP provides greater control over a network than SNMP can provide, CMIP also provides satisfying security.

Paradoxically, the fact that CMIP is more complete than SNMP is the main reason CMIP never took of[21]. As a result of the fact that CMIP uses far more resources and is more complex than SNMP, most TCP/IP devices today support SNMP and not CMIP. CMIP is still around but can not be compared to SNMP in terms of number of implementations.

2.1.3.6 Network Configuration Protocol

The Network Configuration Protocol (NETCONF) is a network management protocol standardized by the IETF (who also developed SNMP). It was published in 2006[22] and later revised in June 2011[23]. NETCONF attempts to accomplish shortcomings of SNMP and is mainly dedicated to configuration management.

NETCONF is fundamentally different from SNMP in that it is based upon the TCP transport protocol and thus is session-based, compared to the message-based nature of SNMP. In comparison with SNMP, NETCONF is generally considered to be a protocol aimed towards configuration rather than monitoring, the opposite of what SNMPs typical use-case have come to be[24]. NETCONF enables configuration to be performed in a transactional manner[25], which SNMP that uses UDP for transport, don't. NETCONF adopts an XML encoded Remote Procedure Call (XML-RPC) which enables communication between managers and agents.

[19] H.-P. U. Warrier L., April 1989.
[20] L. L. U. Warrier L. and Handspicker, October 1990.
[21] Clemm, 2006.
[22] Enns, December 2006.
[23] R. Enns and Bierman, June 2011.
[24] Clemm, 2006.
[25] Clemm, 2006.

2.1.3.7 Common Object Request Broker Architecture

Common Object Request Broker Architecture (CORBA) is a standard to facilitate the communication of systems that are deployed on diverse platforms. CORBA enables collaboration between systems on different operating systems, programming languages and hardware. CORBA is based upon a distributed object paradigm. CORBA uses an Interface Definition Language (IDL) to specify the interfaces that objects present to the outer world. CORBA then specifies a mapping from IDL to a specific implementation langauge like C++ or Java.

2.1.3.8 Vicinity Sniffing

Vicinity sniffing is a technique where a wireless network is monitored by placing out "sniffers" around the network and whose task is to intercept data over the network and analyze the packages.

2.1.4 Network Monitoring Architecture

Network monitoring architecture refers to the structure and functionality of nodes that have some part in the network that is being monitored.

2.2 Graphical User Interface

User Interface (UI) is the space where interactions between humans and machines take place. Graphical User Interface (GUI) is a UI that allow users to interact with computer-based systems through graphical components and visualizations. One of the underpinning research questions we try to answer in this study is how to design suitable interactions between a NOC and its users, see figure 2.6.

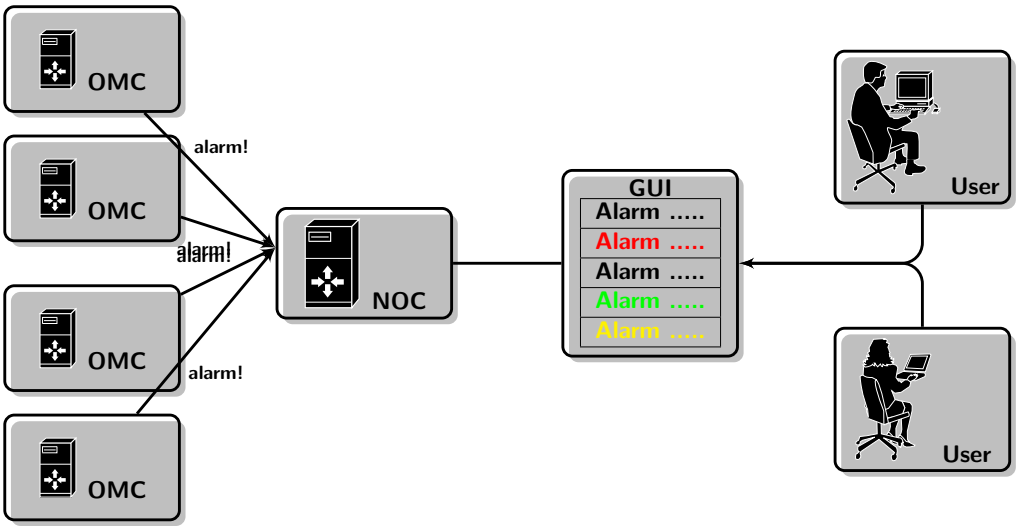

Figure 2.6: Presenting alarms in a GUI

2.2.1 User-centered design

Professionals in the field of human-computer interactions often lift forward the importance of being user-centered when designing interactive systems. What this means is to design systems from the perspective of how the system will be used by the user, rather than from the view-point of hardware capabilities and technologies.

User-centered design is more challenging than not being user-centered for the reason that to understand the future users of the system, many aspects need to be taken into account. The key concerns for designing interactive systems are, as stated by professor David Benyon[26]:

- Design: How should the interface be designed?
- Technologies: What type of devices will the system be interacted with?
- People: Who will use the system?
- Activities and contexts: What activities can the users perform and in what context does those activities take place?

These are some of the questions we try to answer in this pre-study in order to get knowledge about future users. This knowledge is then to be used

[26]Benyon, 2010.

as a base for designing the GUI and will most likely raise the usability of the interface.

2.2.2 Design and Human Perception

In terms of interactive systems design, understanding human perceptual abilities is important background for the design of visual experiences[27]. In particular significant theories on graphical user interface design comes from the Gestalts Principles of Perception[28]. These principles describe the ways that human minds perceive and organize parts into groups of unified wholes. The name "Gestalt" means "form" in german.

The laws from Gestalt Theory that are significant for computer screen design are identified as a total of eleven in the paper "Gestalt Theory in Visual Screen Design - A New Look at an Old Subject"[29]. The laws most relevant for this study are listed in table 2.2.

Law name	**Meaning**
Law of Proximity	Objects that are close to each other are perceived as forming a group[30].
Law of Similarity	Elements in a larger collection are perceptually grouped together if they are similar[31].
Law of Closure	Objects such as shapes, letters etc. are perceived as a whole even when they are not complete, our mind fills in the "rest"[32].
Law of Symmetry	The human mind perceives objects as being symmetrical and divide objects into numbers of symmetrical parts[33].
Law of Isomorphic Correspondence.	The human mind interpret the meaning of different images based on our experiences[34].

Table 2.2: Subset of Gestalt laws

2.3 Sources and Related Work

A great collection of work have been conducted on the area of network management and monitoring as well as in the area of designing graphical user interfaces. The work and conclusions of this study comes from a combined understanding of the primary sources outlined in this segment.

[27] Benyon, 2010.
[28] Benyon, 2010.
[29] Dempsey Chang and Tuovinen, 2002.

In his master thesis, Robert Bern Johnson evaluates the use of SNMP as a monitoring tool for wireless networks[35]. Bern Johnsson identifies the problem of deciding communication protocol for wireless network monitoring and addresses the question with controlled experiements to produce empirical evidence that strengthens his conclusions. The research was conducted by carrying out controlled experiments on a network at a football stadium, as well as comparing SNMP to other techniques for monitoring wireless networks, in particular Vicinity Sniffing. Bern Johnsson were from his experiments able to show that the use of SNMP have equivalent, if not better performance and reliability in capturing network traffic compared to Vicinity Sniffing. Bern Johnsson approaches the problem from a different angle compared to the research questions in this study. Nonetheless his work provides interesting results that have been considered in this study for the process of evaluating suitable communication protocols for implementing network monitoring.

In Alan Marshalls conference paper ”Network management performance analysis and scalability Tests: SNMP vs. CORBA”[36] the performance of the two communication protocols are compared and evaluated. By conducting comparative performance tests, Marshall was able to demonstrate typical bottlenecks of each protocol and give indications when to choose a protocol over the other. His results were a contribution to our discussion on SNMP and its performance.

In the paper “Protocol Efficiencies of NETCONF versus SNMP for Configuration Management Functions”[37] the authors presents an quantitative analysis of the performance characteristics of SNMP and NETCONF. The results are based upon controlled tests in a lab environment. The results illustrate the differencies between the two protocols, which was taken into account in this study when comparing different communication protocols.

Alexander Clemm has gathered many important underlying concepts for network management in his book “Network Management Fundamentals”[38]. The book served as a foundation on the concepts of network management upon which our study has based itself. Clemm also presents protocols and languages for communication inside network management, which relates strongly to one of the reasearch questions behind this study. Clemm has also written chapters about alarm management and the functionalities of alarm management systems, which relates to the specific use case motivating this study.

[35]Johnson, 2009.
[36]Gu and Marshall, 2004.
[37]Brian Hedstrom, 2011.
[38]Clemm, 2006.

In the paper "Securing SNMP: A Look at Net-SNMP (SNMPv3)"[39] from SANS institute, the author Michael Stump reviews the security aspects of the SNMP protocol and in particular presents a discussion on the disparity from a security standpoint between the different versions of the protocol. In many ways Stump describes the aspects considered in this study for our analysis on secure communications with SNMP. Stump draw the conclusion that SNMP is a great way to monitor network devices and by using SNMPv3 you're able to rightly implement secure authentication by the means of encryption.

Dempsey Chang, Laurence Dooley and Juhani.E Tuovinen concluded in their paper on Gestalt Theory in Visual Screen Design[40], that all of the eleven Gestalt laws were found to be useful for visual screen design and user recognition. The conclusions was drawn from an evaluation where a instructional multimedia application was redesigned according to gestalt principles and then tested on nursing students. Chang, Dooley and Tuovinen pinpoints the exercise of applying results from research on humans visual perception to visual screen design, which identify with one of the research questions of this study, "how network information can be present in a GUI in a effective manner".

Professor David Benyon have put together a comprehensive guide to human-computer interaction and interaction design in this book "Designing Interactive Systems"[41]. Benyon's compilation of design principles have been an inspiration in this study for the task of following a user-centered design methodology.

In the book "Essential SNMP"[42] the authors Douglas R. Mauro and Kevin J. Schmidt covers the Simple Network Management Protocol in extensive depth. When justifying the writing of the book the authors mentions, among other things, two broad questions that the book intend to answer: How can I best put SNMP to work on my network? How can I make managing my network eaiser? This book have been used effectively in our process of implementing the SNMP protocol for our prototype.

In the thesis "WEB-BASED NETWORK MONITORING WSING SNMP, CG1 AND CORBA"[43], the author Jizong Li describes the process of implementing a web-based network monitoring tool based on the SNMP protocol using the WWW and CORBA technologies. In the thesis he also reviews the different technologies used. His review of the SNMP protocol

[39] Stump, 2003.
[40] Dempsey Chang and Tuovinen, 2002.
[41] Benyon, 2010.
[42] Mauro and Schmidt, 2005.
[43] Li, 1999.

have been useful in our pursuance of understanding the protocol, the date of when the thesis was published have been in mind when analyzing the content.

While presenting detailed information about the fundamental concepts and parts of the underlying technologies of network management, the related works falls short in giving the reader a concrete insight on how to develop network monitoring systems. This study, with the central problem statement of integrating a specific set of monitoring systems to a network operations center, aims at doing precisely that. By applying qualitative research methods such as analytical research, interviews and prototyping, this study aspire to give a concrete picture of the process of implementing a monitoring system.

Chapter 3

Methods

This chapter describe and review the research strategy and methods that have been applied for this study.

3.1 Research Strategy

To answer the questions stated in section 1.2 *Problem Statement*, an applicable research strategy has been constructed. This section outlines an overview of our strategy and why it was chosen.

3.1.1 Main Methodology

The proposed methodologies and procedures in this study was designed, in consideration of the problem statement, to give a thorough understanding of *how* a wireless network monitoring system can be implemented in an existing infrastructure. The objective of applying the chosen methods was to understand what design choices in the development process should be considered to achieve a satisfying and sustainable product.

In this study qualitative research methods have been preferred to gain a profound understanding of the problem area, current solutions and possibilities for developing new solutions. This study used a combination of analytical research, interviews and implementation.

- **Analytical Research**.
  Analytical research uses facts and information that has already been collected and analyses that material to make a critical evaluation of

it[1]. The analytical parts of the research was carried out to comprehend the current state of network management and serves as a foundation for making decisions regarding the implementation part of the research. An essential part of our analytical research was to identify patterns and relations between hypotheses and conclusions from previous research related to the problem statement.

- **Interviews.**
  Being that this study concerns implementation of a NOC in a specific environment, the chosen research strategy includes methods to assemble information about future users of the system. This information has been collected by making semi-structured interviews on employees of MIC Nordic. The interviews aspire to capture the users point of view in order to lay a foundation for a user-centered design of the GUI.

- **Implementation.**
  The implementation part of the research in this study consists of development of a prototype. The prototype serves as an investigation of to what degree the conclusions from previous work can be utilized in the specific environment that the problem statement concerns. By developing a prototype, empirical evidence is acquired to justify or reject beliefs and understandings collected through the analytical research.

3.1.2 Maintaining Scientificality

It is important that a scientific approach is maintained in the design and implementation phase and forwards. Niclas Andersson and Anders Ekholm mention in their study "Vetenskaplighet – Utvärdering av tre implementeringsprojekt inom IT Bygg och Fastighet 2002"[2] (a study on scientificality in research projects) the importance of constructing theories in a scientific context. They mention two methods for systematic theory construction, induction and deduction. These two can combine into the Hypothetico-deductive model, in which you reason from the starting point of a hypothesis.

As of such, the knowledge acquired from our analytical research serve as the basis for the construction of several hypothesis that we attempt to verify or falsify during the implementation and analysis process.

[1] Håkansson, 2013.
[2] Andersson and Ekholm, 2002.

3.1.3 Process Overview

The work of this pre-study was divided into several serial phases containing parallel sub-phases, see figure 3.1.

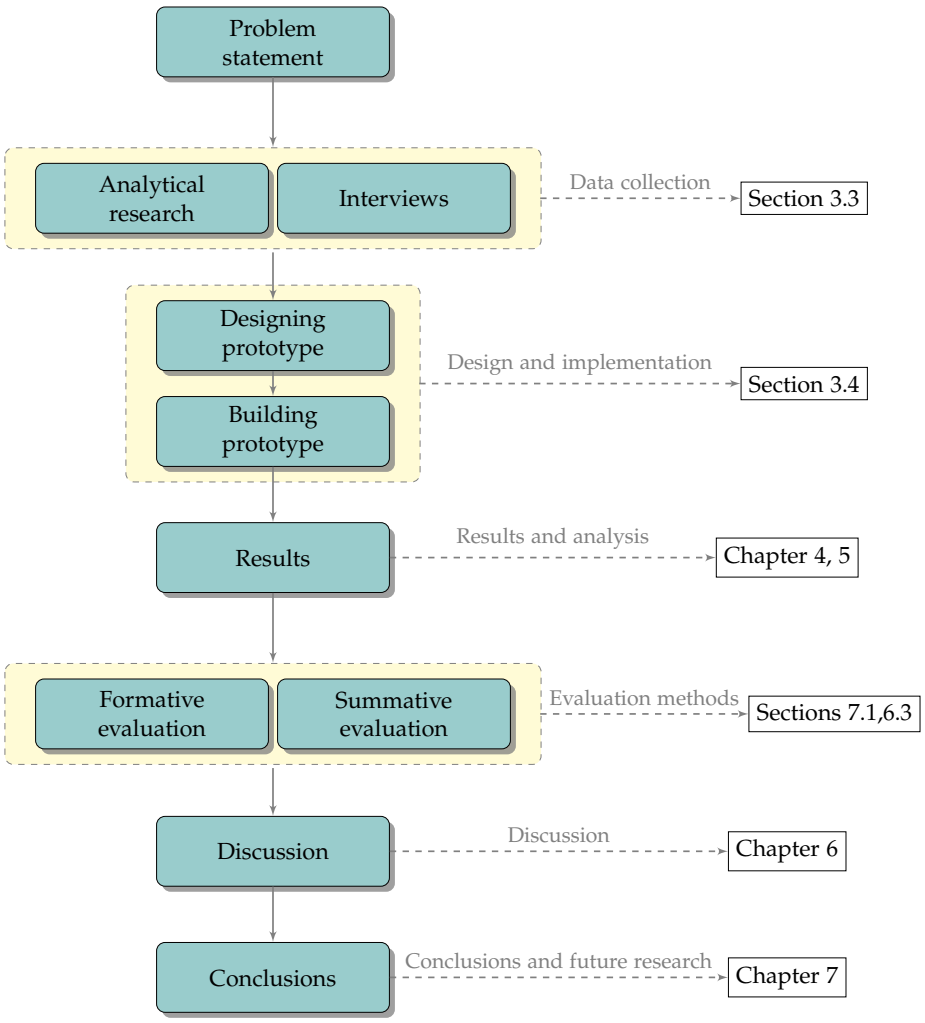

Figure 3.1: Research strategy overview.

3.2 Understanding The Problem Statement

Before the data collection phase, a proper understanding of the problem statement had to be acquired. This understanding commenced from discussions with MIC Nordic. At these meetings the initial problem was presented and an insight on MIC Nordics future visions was gained. The

shortcomings of the current solutions were also demonstrated which gave a valuable background of the problem area.

3.3 Data Collection

3.3.1 Analytical research

The information gathering process initiated with assembling documentation for the systems in place at MIC Nordic. From the breakdown of MIC Nordic's environment and the problem statement, relevant concepts for this study were identified. With the leading concepts figured out, related sources were assessed and organized. For an overview of this process see figure 3.2

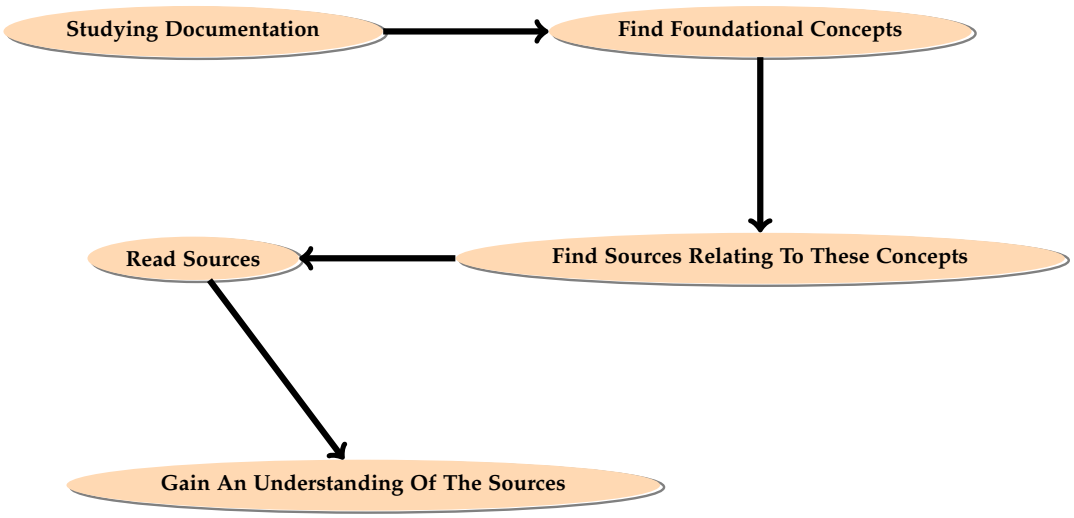

Figure 3.2: The data gathering process.

- **Studying documentation**
  Since the area of this study is delimited to the OMCs in place at MIC Nordic, the data collection was initiated by closely examining these systems corresponding documentation. This gave an understanding of how the systems work and which limitations they inhabit.

- **Find Foundational Concepts**
  With the problem statement established and a comprehension of the environment of MIC Nordic, the next undertaking concerned further breaking down the problem statement into areas to study. Hence the main purpose of this part of the data collection was to decide

what should be analyzed. This process included further examination of the research area and analysis of documentation related to the problem. With the purpose of discovering patterns that reveal the foundational concepts of the research area.

- **Find Sources Relating To The Concepts**
  In this study the primary means of finding sources have been carried out by browsing accessible libraries and the web for papers published on the research area. To ensure if a certain source is relevant or not a technique labeled by Mikael Berndtsson in his book "Thesis projects - a guide for students in computer science"[3] as "Bibliographic databases" have been used. Bibliographic databases is a method of browsing a source for a set of keywords, which our understanding of the problem have led us to believe are potentially relevant. This technique also includes following up on bibliography lists once a relevant source have been found.

- **Read Sources**
  When related sources was identified, the relevant parts were filtered out. This process included reading the literature and comparing against the foundational concepts of the problem statement.

- **Gain An Understanding Of The Sources**
  By reading multiple sources tackling in many ways the same problem but from different perspectives. We we're able to contrast the conclusions and results against each other to get an impression of common conclusions and ways of solving the problem.

3.3.2 Interviews

When composing the interview questions, the aim was to construct questions that relates to key concerns for designing interactive systems, as was listed in previous chapter. The interviews have been undertaken in a way that is termed *open inteviews*. An open interview is as described by Mikael Berndtsson[4], a form of interview where, even though the purpose of the interview is clear to the researchers, the specific issues to be covered during the interview is not planned in advance. Instead the interviews are directed according to what the interview subject answers. This way of inteviewing have been selected to get a more focused interview with the intent of collecting answers suitable for deeper analysis. The interview questions can be found in appendix A.

[3] Berndtsson, 2008.
[4] Berndtsson, 2008.

3.3.2.1 Credibility and relevance of interview results

Credibility of the interview results refers to a measurement of how truthful and valid the results are. The interviewees in this study have been employers of MIC Nordic, meaning that there shouldn't be any doubt that the interviewees intend to supplying honest answers and opinions in order to contribute to a better end result (future development of the system).

The interviewees have been selected so that they are acquainted with the already existing monitoring systems and are also likely to be future users of the system to be developed. As follows, the relevance of the interview results can be considered high.

3.4 Design and Implementation of a Prototype

A prototype is a concrete, but partial representation or implementation of a system design[5]. In the context of this study the prototype refers to a implementation of a NOC. The intent of the prototype is to demonstrate two essential concepts of this study. The concept of communication between monitoring systems and the concept of GUI-design, which both relate to the main research questions.

3.4.1 Design of Prototype

The design of the prototype can be separated into two different branches:

- **System design**
  System design refers to the process of determining the architecture, functionality and components of the prototype. The system design was derived both as a consequence of the analytical research phase and from a dialogue with MIC Nordic to investigate typical use-cases of the NOC, functional and non-functional requirements.

- **GUI design**
  Results from the analytical research and interviews serves as a base for establishing a GUI design. The GUI design process followed a user-centered approach. This process included development of multiple versions of the same interface and evaluation in association with future users. Each version follow its own pattern for presenting the network management information. By developing multiple

[5]Benyon, 2010.

versions, evaluating and comparing them against each other, we acquired a foundation for making conclusions regarding presentation of network management information.

Figure 3.3 depicts the different methods and how they relate to each other for implementing the prototype.

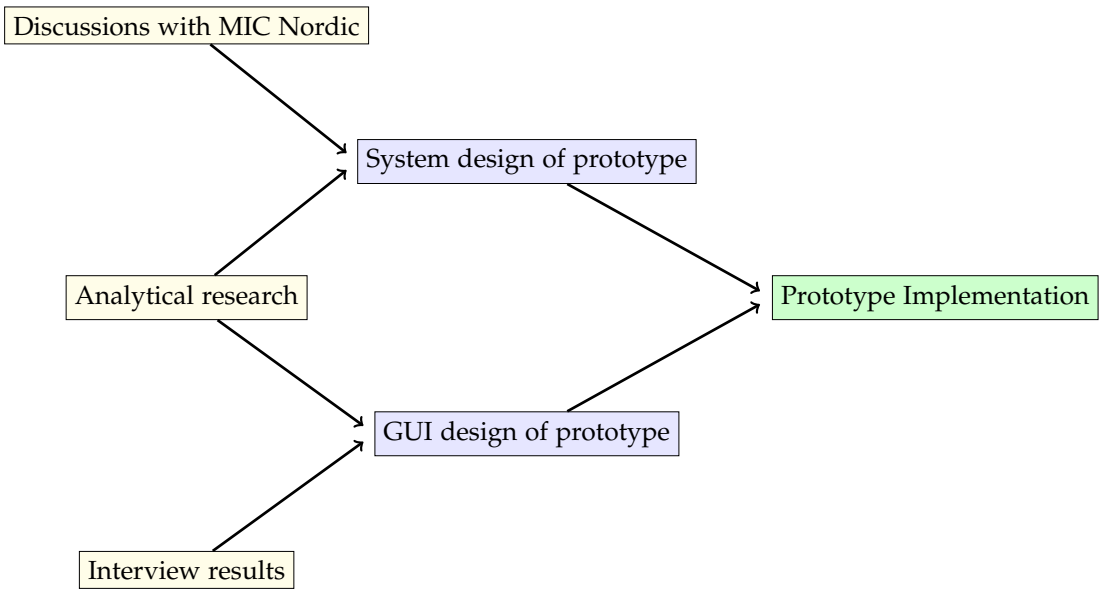

Figure 3.3: Implementation process

3.4.2 Implementation of Prototype

The phase of putting together the prototype was completed iteratively, in parallel with the design process and with partial deliveries of the product to MIC Nordic. As is further discussed in the following chapters, the specific technologies, programming languages and environment used for development is not of importance in this study. The prototype was rather developed to examine what protocols, system design and GUI design should be used to best answer the problem statement.

3.5 Evaluation Methods

The results from the implementation have been evaluated to determine whether the prototype fulfills its purpose and overall effectiveness in meeting the requirements and objectives. The evaluation results are a crucial

part of the pre-study since it lays a foundation for future development of the system.

3.5.1 Formative evaluation

During the process of implementing and designing the prototype, multiple evaluations have been conducted. The implementation and design phase was done in an iterative cycle and included evaluation and feedback from stakeholders of the project. The intention of this evaluation was to verify that the project was progressing in the right direction. This practice of continuous evaluation during the implementation and design phase, is in this study referred to as *Formative evaluation*.

3.5.1.1 Heuristic evaluation

Heuristic evaluation refers to methods were a person examines a proposed design to see how it measures up against a list of principles and guidelines (or in short "heuristics") for good design[6]. The heuristic evaluation have been considered a suitable method for evaluating usability and design due to its flexibility. In the context of this study the heuristics have been constructed as a combination of general principles for good design together with principles that in specific relates to network monitoring.

3.5.1.2 Evaluation of Functional Requirements

Evaluation of the fulfillment of functional demands have been organized in a manner where a checklist of functional demands have been constructed and at every iteration the prototype was measured against that list. This evaluation have also been conducted in conjunction with demonstrations of the prototype to the involved stakeholders, MIC Nordic.

3.5.2 Summative evaluation

When approaching the endpoint of this pre-study, a *summative evaluation* was completed. The aims of the summative evaluation in the context of this study was to evaluate fulfillment of the purpose of the study.

[6]Benyon, 2010.

Chapter 4

Data Analysis: Communication and Presentation of Information

In this chapter the results from the analytical research and interviews, described in the previous chapter, are put together and analyzed. The results of this analysis provides a basis for our implementation of a prototype and the conclusions drawn from this study.

4.1 Communication

4.1.1 Protocol

When deciding upon a communication protocol to use in network monitoring there are few things to consider, depending on the requirements of the system. In this study the main concerns were:

- Integration with established network equipment
- Security
- Reliability
- Performance

4.1.1.1 Why SNMP

The implicit strategy of the Simple Network Management Protocol (SNMP) is to keep things simple. The SNMP core comprises of a small set of operations and a general *manager-agent* architecture that can be used to compose monitoring hierarchies. SNMP also has the convenience of being widely implemented and supported. The majority of vendors of internet-hardware design their products to support SNMP today[1]. We will look into if SNMP is a worthwhile protocol to use for its characteristics, beyond its wide support and popularity.

SNMP is not without caveats. As was briefly mentioned in section 2.1.3, SNMP-messages are represented within single UDP datagrams. This means that the protocol is not suited for polling large volumes of data. Being that the underlying transport protocol, UDP, is a connectionless protocol also entails that no guarantee for arrival of messages are inherited from the transport layer. Additionally, the simple design of SNMP means that its way of dealing with data and information is not organized and can be difficult to deal with when the data is complex.

Earlier versions of SNMP have been inadequate by the means of security, this have been sorted with the most recent version of SNMP but the insufficiency of earlier versions still need to be considered, seeing that they are still in use. This is further discussed in section 4.1.1.3.

Yet, in the circumstances of this pre-study, SNMP have been concluded to be the best fit as a communication protocol for implementing a NOC. The benefits of SNMP outweigh the negative in this situation. The reasons being:

- The already established OMCs all have support for SNMP
- Being that the NOC will act as an alarm management system, the limitation on SNMP when it comes to polling large volumes of data is not an immediate issue. Most of the communication can be event-based, using traps, without the need for massive data transfers at once.
- We found the unreliability of UDP and SNMP not to be a particular complication. The OMCs involved in this study provides alternative ways of assuring reliability. This subject is further discussed in section 4.1.1.4.
- Early versions of SNMP can coexist with the most recent version, which implicates that in the case of confidential communication,

[1] Johnson, 2009.

security can be fully implemented without having to upgrade the whole network to another version of SNMP.

- SNMP is flexible and lightweight in that it's not dependent on any specific technology apart from regular network communication over UDP/IP. Thus when developing a solution for SNMP, by implementing our own parsing of messages, we're not dependent on any library or specific environment. This entails that all technology stacks that have access to regular network-socket communication are applicable. Which is an advantage when developing for different platforms. This is further discussed in section 5.4.

- Lack of options. During the rise of the internet no other network management protocol have managed to replace SNMP. As a result of SNMP being developed early, it's the most popular and extensively used protocol for network monitoring today. Alternative protocols are discussed in the next section.

4.1.1.2 Alternative Network Protocols

We have found the Constrained Application Protocol (CoAP) to be a considerable alternative to SNMP when deciding upon the communication protocol. As was previously discussed in section 2.1.3.4, CoAP have many similarities with SNMP, but differ in what problems the protocol have been developed to solve.

CoAP is explicitly designed for Machine-to-Machine(M2M) interactions where the devices and networks are limited in terms of resources. CoAP is a modern protocol that has its own implementation of a RESTful API, thats facilitates integration with HTTP and other web applications. CoAP servers uses resources that are mapped to URL's and allow for clients to access these resources using methods such as GET, PUT and POST. This can be utilized to implement a close replica of a SNMP communication with SNMP GET/SET requests and a MIB as the resource.

Property	**SNMP**	**CoAP**
Transport protocol	UDP	UDP
Existence	Since 1988 (ish)	Since 2013 (ish)
Security	Sufficient (v3)	Sufficient
Objective	Network Management	Internet of Things
Encoding	ASN.1 BER	Custom binary format
Communication Model	Request/Response	RESTful API

Table 4.1: CoAP and SNMP comparison

The main drawback of CoAP in the context of this study is the usage. SNMP has been around for much longer than CoAP and is widely used for network management implementations, CoAP is not. Considering that the OMCs in MIC Nordics setup uses SNMP and that the benefits of CoAP is typically highlighted in other situations than network management, SNMP was chosen as the underlying communication protocol for the implementation of this NOC.

Other protocols that were looked at in our research are:

- **CMIP**
  We can ignore the fact that CMIP isn't really an option due to its limited support in general network devices[2], and compare it to SNMP in terms of their specifications. The largest advantage of SNMP over CMIP is that its design is simple, the same can not be said of CMIP. In the context of this study, CMIP wasn't really a contender when chosing communication protocol. CMIP is not supported by the OMCs in MIC Nordics current setup and even if it were, CMIP would bring unneccessary complexity and resource usage. The tradeoff for more control have been considered superfluous in the context of this study.

- **NETCONF**
  We consider NETCONF to be a viable option for complementing SNMP in terms of congifuration. Yet in terms of monitoring, there is no real tradeoff in comparison with SNMP. Being that SNMP is supported in MIC Nordic's current monitoring setup and that the main use-case of the NOC will be monitoring, not configuring, the choice of SNMP was rather clear. In Clemms book "Network Management Fundamentals"[3] he describes NETCONF as a protocol that is designed for configuration, and is suited for environments where a protocol such as SNMP is assumed to be around to handle monitoring. Hence, if the requirements on the NOC would change in the future, NETCONF could be worth looking at for complementing SNMP in configurement.

[2]Clemm, 2006.
[3]Clemm, 2006.

Property	SNMP	NETCONF
Transport protocol	UDP	SSH/TCP
Existence	Since 1988 (ish)	Since 2006 (ish)
Security	Sufficient (v3)	Sufficient
Objective	Network Management	Network Configurement
Encoding	ASN.1 BER	XML
Communication Model	Request/Response	RPC

Table 4.2: NETCONF and SNMP comparison

- **CORBA**
  CORBA is generally considered too heavyweight and expensive in comparison with other techniques for network monitoring (more about this in section 4.1.1.5). When deciding upon a communication protocol in this study, CORBA have been considered but concluded not to be a favorable option due to its heavyweight nature and that SNMP have been thought-out to be superior.

Property	SNMP	CORBA
Transport protocol	UDP	TCP
Existence	Since 1988 (ish)	Since 1991 (ish)
Security	Sufficient (v3)	Up to the implementation
Objective	Network Management	Distributed Objects
Encoding	ASN.1 BER	IDL
Communication Model	Request/Response	RPC

Table 4.3: CORBA and SNMP comparison

- **Vicinity Sniffing**
  A technique for network monitoring that was extensively compared against SNMP in Bern Johnsons thesis[4], is Vicinity Sniffing. Vicinity sniffing have not been considered as a sufficient technique for the type of monitoring that a NOC, in the context of this study, requires. The main reason being that the network monitoring follows a request - response relationship between network components, which is not suited for Vicinity Sniffing. From our understanding, Vicinity Sniffing solves a different problem than what the typical use cases of SNMP, CMIP and CoAP does.

[4]Johnson, 2009.

4.1.1.3 SNMP and Limits On Security

The earlier versions of SNMP have restrained functionality when it comes to security. We will in this section consider if the security limitations that SNMP brings would affect an implementation of a NOC alarm management system.

RFC 1157[5] mentions how SNMP implements authentication by using community strings. A community string serves as a type of password in SNMP communication. If the community string sent with the actual message matches the one on the receiving end, that message is authenticated. In SNMPv1 and SNMPv2 this community string is sent in plain text. This means that the community string can be understood by anyone who has access to the network. A person with access to the network can run a packet-sniffing application to acquire your community string and then use it to inventory all of your devices that use this community string for authentication.

In more recent SNMP versions this is no longer a problem. SNMPv3 supports additional features for security. Including enhanced authentication and encryption, as discussed by Michael Stump via the SANS Institute InfoSec Reading Room[6]. With SNMPv3 you can use Secure Hash Algorithm (SHA) or Message-Digest Algorithm (MD5) encryption when carrying out your SNMP communication. Also, in version 3, SNMP moves away from the authentication with community strings to a user-based security model, where you authenticate yourself with username and passphrase over an encrypted communication channel.

Yet, one of the OMCs at MIC Nordic only support SNMPv2[7], so we cannot work under the assumption that all systems will support the features of SNMPv3. It is possible to have coexisting SNMPv2 and SNMPv3 devices in the same network monitoring hierachy. Considering the security benefits of SNMPv3 it is recommended that if not old devices running SNMPv2 are upgraded, newer devices will adopt SNMPv3.

Jizong Li argues that SNMPv2, for security reasons, is better suited for monitoring (GET-requests), rather than controlling (SET-requests), in his thesis "WEB-based Network Monitoring Using SNMP, CGI and CORBA"[8]. Since we have considered the typical use case of SNMP in the system that this study concerns to be forwarding of the status of subsystems (alarms), this falls into the category of monitoring.

[5] Case et al., May 1990.
[6] Stump, 2003.
[7] Solutions, n.d.
[8] Li, 1999.

We could also consider the likelihood of falsified messages using the correct community string to be fairly unlikely. Even if a falsified message would occur and leads to a false alarm appearing in the system. The alarm would eventually be handled by one of the users and the situation would get resolved. As of such we consider SNMP secure for monitoring the messages from OMCs. However, a different protocol would need to be considered for controlling said OMCs.

4.1.1.4 SNMP Reliability

SNMP uses the UDP protocol for transport. UDP was preferred over the *Transmission Control Protocol* (TCP, defined in RFC 793[9]) because it is connectionless, which is not the case with TCP. The attribute of being connectionless makes UDP an unrealiable protocol since no end-to-end connection is made between the two ends of the communication. By extension this also makes SNMP unreliable. In the book "Essential SNMP"[10], the authors argues that the unreliable nature of UDP isn't a real problem. Because of UDP's unrealiability it's up to the SNMP implementation to determine if datagrams are lost and then take a suited action if so is the case. This is typically accomplished with a timeout, where the SNMP Management Station can send a UDP request to a SNMP Agent and if no response is received in a certain amount of time a timeout is triggered.

SNMP Traps are more tricky and is where the unreliability of UDP makes a disadvantage for SNMP. When an SNMP Agent sends a Trap there is no guarantee that the SNMP Management station responds and thus we cannot rely on the timeout mechanism. A way to deal with this is having the SNMP Agent sending traps multiple times.

So why is UDP chosen then? The profit of using UDP and its unreliable nature is that it requires a low overhead, so the impact on the networks performance is reduced. In "Essential SNMP"[11] it's mentioned that SNMP has in fact been implemented over TCP but that it have been concluded to be a bad idea to use TCP for transport in SNMP. It should be mentioned that TCP isn't magic, it is just a guarantee that failed transmissions will be retransmitted in hope that it will work the second time. If your network is failing, TCP won't help more than UDP.

> *If your network never failed, you wouldn't need to monitor it*[12]

[9] Southern California, September 1981.
[10] Mauro and Schmidt, 2005.
[11] Mauro and Schmidt, 2005.
[12] Mauro and Schmidt, 2005.

For a closure on the discussion of the reliability of SNMP, we can conclude from our analytical research that SNMP has acceptable reliability and that when communicating over the network there is no such thing as 100% reliability. The reason SNMP and other similar network protocols uses UDP is for the upside of low overhead and no risk of flooding the network in case of failures.

4.1.1.5 SNMP Performance

In the paper "EVALUATING THE USE OF SNMP AS A WIRELESS NETWORK MONITORING TOOL FOR IEEE 802.11 WIRELESS NETWORKS."[13], SNMP performance is evaluated by conducting controlled experiements of network monitoring at a football stadium. The results of the experiement are then compared against the result of the same experiement with a different technique for network monitoring, Vicinity Sniffing.

The outcome of the conducted experiements demonstrated that, compared to Vicinity Sniffing, SNMP is more performant in the means that SNMP can capture a higher amount of data. The experiements also showed an inconvenience with SNMP in that the protocol is generally not able to provide as detailed information as Vicinity Sniffing. The result of the data captured by the Vicinity Sniffing technique depends on the placement and quantity of sniffers.

> *We also compared the number of frames that each technique was able to capture, and in addition, we were able to provide statistical evidence to show that in some instances SNMP is capable of more accurately capturing network traffic than Vicinity Sniffing.*[14]

In the paper "Protocol Efficiencies of NETCONF versus SNMP for Configuration Management Functions"[15], the authors put NETCONF head-to-head with SNMP in terms of performance and efficiency. The results illustrate that SNMP is more lightweight than NETCONF in that the required data transfer is smaller. The authors conclude that this is likely due to several reasons. NETCONF that uses TCP for transport have a larger overhead than SNMP that uses UDP, also the session-based nature of TCP requires a connection to be established. Another aspect to consider is that in the tests NETCONF was running over SSH with security built in, while no such security was used for the SNMP version.

[13] Johnson, 2009.
[14] Johnson, 2009.
[15] Brian Hedstrom, 2011.

In terms of operation times, the results illustrated that when the number of managed objects in the test exceeded $\sim$ 500, NETCONF's operation time efficiency surpassed SNMP. The results highlighted the profits of NETCONF in terms of configurement, compared to SNMP.

> *NETCONF is most powerful when it stacks up against SNMP for number of transactions. NETCONF is able to configure 100,000 managed objects in a single transaction, using XML configuration data as its payload, while SNMP's best case scenario is 2779 transactions for the same number of managed objects.*[16]

The results give us a further understanding regarding the benefits of using NETCONF for configurement compared to SNMP.

Alan Marshall showed in the paper "Network management performance analysis and scalability tests: SNMP vs. CORBA"[17] that when manipulating smaller amount of data (singe objects), which is the typical use-case of the NOC in this study, SNMP costs less network bandwidth and has less latency than CORBA. Marshall concludes that this is mainly a result of the different transportation layer protocol. SNMP that uses a message-based protocol like UDP has less overhead than the session-based counterpart TCP that CORBA uses. Another interesting outcome of Marshalls tests were that it pinpointed the defiency of SNMP in polling larger volumes of data. The tests showed that when the data volume got larger, SNMP revealed clear performance issues while CORBA stood strong in performance (up to 26 times better than SNMPv2 for a data-table of 2000 rows), Marshalls research show that this also is partly a consequence of the different transport protocols. SNMP that uses independent UDP packages scale very poorly when larger volumes of data is to be transferred while CORBA that uses TCP, scales in a satisfactory manner.

> *For example, to retrieve a table with 2000 rows, CORBA uses 43 IP (TCP) packets from client to server and 77 IP (TCP) packets from server to client. SNMP V1 uses 2000*8=16000 IP (UDP) packets in both directions.*[18]

Marshalls results and conclusions reinforce our understanding that SNMP constitutes as a competitive option compared to other network protocols when used in environments of smaller amount of data, but have drawbacks when bigger volumes of data is involved.

[16] Brian Hedstrom, 2011.
[17] Gu and Marshall, 2004.
[18] Gu and Marshall, 2004.

Considering that the work that this thesis is based upon is of a pre-study nature, own performance tests and evaluations have not been performed. The study has instead acknowledged previous research on the subject. The results and conclusions from the sources outlined in this section are dependent on the environments and objectives used when carrying out the tests and cannot be applied directly to this study. However by critically examining and comparing multiple sources and their corresponding conclusions we acquire a substantial base for making conclusions regarding the performance of SNMP.

To sum up the discussion on SNMP performance, the preferred technique of choice depends on the situation, and SNMP is not always a good fit. In the context of this study SNMP has been considered more suitable than other alternatives considering that the main use-case of the NOC is monitoring not configuring. Moreover, the monitoring concerns presentation of alarms and status, not detailed statistics about network packages. There is also no reason to believe that the NOC will need to carry large volumes of data in single operations.

4.1.2 Monitoring System Architecture

4.1.2.1 Monitor Hierarchy

In the means of this pre-study one of the underlying research questions is how multiple networks can be supervised jointly, which requires that the network monitoring is distributed. A distributed network monitoring system can be viewed as a set of nodes in a monitoring-hierarchy. A typical architecture for distributed network monitoring systems is to distribute the monitoring on several network management stations, where each station is responsible for one section of the network. This is also what have been considered the best solution in this study. The main reasons being the structure of MIC Nordics current monitoring setup and the identified benefits of using a distributed hierarchical model.

The monitoring hierarchy is built upon one central NOC that interacts directly with OMCs, without the concern of interacting with the actual network elements that are being monitored. This type of architecture makes it possible to add new OMCs monitoring different networks independently of the current ones. There is also a possibility of extending the network monitoring by adding new layers in the hierachy. For example, the OMCs that the NOC monitors could in turn monitor other OMCs that then monitors different sets of network elements.

From the analytical research our understanding is that this is the dom-

inant way of structuring larger monitoring setups and there aren't any other viable options than structuring it in a hierarchical way. Alexander Clemm[19] mentions *Telecommunications Management Network* (TMN), developed by ITU-T, as a well-established way of categorizing different layers in your monitoring hierarchy. The TMN model is based upon the OSI management specifications and consists of four logical layers, see figure 4.1.

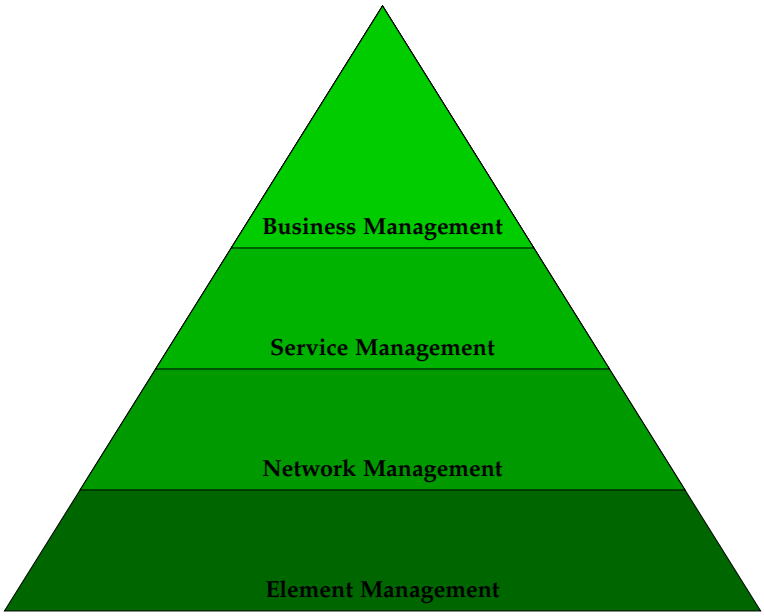

Figure 4.1: Logical layers of the TMN-Model

The TMN-model is intended for monitoring setups on a larger scale than what this study concerns itself with. If we would place the monitoring hierarchy used in this study in the TMN-model, the NOC would reside in the second layer, *Network Management*, and the OMCs would reside in the first layer, *Element Management*. The TMN-model should be considered if the monitoring setup is to be extended in the future, but it is not in the scope of this study.

4.1.2.2 Scalability Of Network Components

A NOC needs to allow scalability in regards to the amount of network components to monitor. In the case of this study, it's a prerequisite that the NOC allows for the addition or removal of OMCs during the lifetime

[19] Clemm, 2006.

of the system. Looking through the documentation for the initial OMCs, we found that all of them operate on unique MIBs. This entails that the information that is sent towards the NOC will differ. One of the OMCs also requires that the NOC responds to each sent message in order to acknowledge that it has been received. This response is an SNMP-message that must contain a specific set of variable bindings.

It is then a necessity that the NOC can handle the different types of configuration and information that the OMCs demands. Therefore, the process of adding OMCs to the system need to take into consideration possible unique setup requirements for each OMC. To make this process efficient, the system should have a flexible setup process for new OMCs.

4.2 Presentation of Collected Information

4.2.1 Accessing the NOC

Given the requirement that the NOC should be accessible from multiple workstations of different character (such as a computer or a phone), how should the user access the NOC?

By presenting the GUI as a web application and designing the application to be responsive for screen resolution and input methods, users using devices of different size can access and use the site in a compelling way. Using a web application means that no significant extra consideration needs to be taken to the potential running platforms, as would otherwise be the case with an application based GUI.

4.2.2 Graphical User Interface Design of a NOC

Our conclusion from the analytical research is that the most severe pitfall when designing GUIs is to forget about the user. As stated by professor David Benyon when motivating the need for being user-centered when designing interactive systems:

> *It is necessary if we are to have safe, effective, ethical and sustainable design.*[20]

Being user-centered is about involving users in the design process and thinking about what people can do rather than what technology can do.

[20] Benyon, 2010.

Through our appliance of design principles to the domain of network monitoring. We have come to the understanding that in the context of developing a NOC, the main concerns of presenting information in a GUI is that the interface should grant a high readability of the network status. This enables the user to get a swift understanding of the information and be more precise and effective in its assessment. This includes communicating both the raw information and where the information comes from. The main challenge we have found when designing a GUI for a NOC is to, through data visualization, communicate to the user an overview of the system status. Without forcing the user to closely examine every alarm to gain that same understanding.

The approach that we have found suitable to designing a GUI for a NOC is to follow a user-centered methodology and to apply human perception theory. When declaring that a user-centered methodology have been used, it is implied that, rather than relying on own guesses and beliefs, design decisions are based on research about future users. The intent with applying human perception theory is to design a GUI that promotes learning and ease of understanding for the user.

4.2.3 The Role of Human Perception Theory In Design

When applying human perception theory to the design of a NOC, we have found the laws from Gestalt Theory to be particular useful. In the research paper "Gestalt Theory in Visual Screen Design — A New Look at an old subject"[21], eleven of Gestalt Laws where evaluated by conducting a participant-based experiment. In the experiment an old GUI was redesigned with the Gestalt laws in mind and the results were vigorously positive.

> *In the redesign of the main WoundCare screens all of the eleven Gestalt laws identified were found to be useful by an overwhelming number of respondents.*[22]

An interesting aspect from their research is that the redesign in the experiment was conducted on an existing GUI, that were used regularly for practical reasons and was not designed with human perception in mind. This closely relates to this study and one of the research questions.

Their results are linked to a different type of UI then the one that this study concerns, thus we need to review their results for what they are, and

[21] Dempsey Chang and Tuovinen, 2002.
[22] Dempsey Chang and Tuovinen, 2002.

not expect a direct mapping to UIs for network management. Nonetheless their results gave an insight on how human perception theory can be applied to the design process by using gestalt theory to design UIs for usability.

4.2.4 Definition of an Alarm

In previous chapter we mentioned Alexander Clemms abstract description of what an alarm is:

> *Every alarm is an indication of an underlying condition*[23].

Although it would be false to claim that there is a consensus on how alarms should be constructed, our understanding from the analytical research is that it is recommended to follow some established and researched standard. We have chosen to go with the X.733 standard for our implementation in this study. The main advantages of X.733 is that it is by many considered as the *de facto* standard for alarm contents. X.733 have been around for decades and many later standards are based upon it. Other standards that have been considered are: 3GPP[24] and IETF Alarm MIB[25].

4.2.4.1 Classification of Alarms

As a result of following the X.733 standard, there exists a natural classification of alarms by severity and alarm-type. In addition, our research on the specific context of MIC Nordics network monitoring has led us to a requirement that the alarms also should be classified according to the source from where the alarm originated.

4.2.5 Interview results

The collected answers of the conducted interviews with employees of MIC Nordic are presented in appendix A, together with the interview questions. The results from the interviews led us to a judgement on the four main elements, that we had setup as a framework for getting to know our users and the system environment. The purpose of the results is to facilitate the design of a user-friendly GUI.

[23]Clemm, 2006.
[24]COMMUNICATIONS, 2007.
[25]S.Chisholm, September 2004.

- **Design:**

  The interview results included some evaluation of the GUI design of the current OMCs and the result were largely positive, but with a few requests on how the GUI of the NOC can be improved.

  A distinction between the OMCs and the NOC will be that the NOC collects alarms from multiple OMCs. It is then desirable that when presenting the alarms, the user can rapidly distinguish the alarms on two aspects, the source of the alarm and the severity. A proposition in the interview was given to utilize colors to achieve this.

  In the interview there were also a conclusion that the OMCs falls short in giving useful statistics, and a request that the NOC should collect and present statistics was made. The more statistics the better were the agreement, but some statistics were listed as being particular useful:

  - Number of alarms per severity
  - Source of alarms by system
  - Source of alarms by physical location

- **Technologies:**

  The system should be usable on every screen size that is the size of a smart phone or larger. This call for a GUI-design that is responsive and usable with different input methods.

- **People:**

  The OMCs are currently being used by 3-4 people daily and the typical use-session commonly consists of viewing the status of network equipment or restarting a network element (a repeater).

  An ambition is that the new NOC will be more accessible and provide a better overview than the OMCs. This would enable multiple users to use the system at the same time. By providing a better status overview from the NOC, it can encourage the users to view the status more frequently.

- **Activites and Contexts:**

  Typical activities for the NOC and their context will be:

  - *View incoming alarms.*
    Either in the context of obtaining general knowledge or in a context where the user is searching for a specific alarm.
  - *Request status of a specific network component.*

- *View statistics of the NOC.*
  In the context of general information or in the context of evaluation.
- *Configure a network component.*
  This acitivity will be under a test-period when the system goes into production to conclude in what way this can be done eficently. One option is to redirect the user to the OMC from where the alarm originated, another strategy would be to handle the alarm directly from the NOC.
- *Configuring the NOC.*
  This could include extending the set of supervised OMCs, configuration on what statistics to present or configuration of what saving policy to be used (e.g. which alarms should be saved and for how long).

4.3 Current Research and Development

Network management has always been important for the industry but only in the last few years have it recieved a similar level of attention from many research communities[26].

In late 2007 there was an article published by the IEEE on the subject of current research in network management. The article "Key Research Challenges in Network Management"[27], presents major findings from a two-day workshop organized jointly by the IRTF/NMRG and the EMANICS Network of Excellence at which researchers operators, vendors and technology developers discussed the research directions to be pursued over the next five years. The article summarizes some of the main discussions at the workshop. The topics that we consider relevant for the alarm management system that this study concerns are presented briefly in this section.

- **Architectures**
  The article mentions that many researchers have worked on the definition of network management architectures in the past. As a result, the industry now have a good understanding of different architecures, including the agent manager model and other distributed architecture approaches. The IETF, for example has been working on three different kinds of distributed management, including the MIB-based approach. Further more the ITU-T have in their TMN-series,

[26] Stiller; 2007.
[27] Stiller; 2007.

defined functional, physical, information and logical layered architectures for network management.

- **Data Analysis and Visualization**
  The task of visualizing and analyzing data from network monitoring, as identified in this study, is devoted some discussion in the article. The article mentions that network management systems operate on large data sets that must be aggregated, filtered and visualized with the goal of making meaningful information easily accessible to human network operators[28].

  The article mentions that the first generation network management systems used two dominant approaches for visualizing the data. Topological network views and time series plots to visualize the evolution of key metrics over time. The authors too mentions that most systems today are accessible via web interfaces (periodically updating web pages), but experiements also have been made using TV channels to make network graphs accessible to a large number of network users.

  A quite interesting note from article is that the authors mentions, as also is our impression, that altough data analysis and visualization is an old network management topic, it seems that available techniques and interfaces for human network operators are not really satisfying. The article mentions multiple reasons for this.

  Regarding visualization of data sets and statistics the article describes that they are often visualized in a rather static way. There is typically no or only very limited support to explore data sets (e.g. by applying filters, zooming functions or correlation functions) in an interactive way. The article estimate that recent technology (2007), such as Google Earth, will act as an enabler for the development of new techniques for visualizing the data through geographic maps where zooming and on-the-fly data aggregation can be explored.

- **Behaviour of Managed Systems**
  The article also presents a engaging discussion on the behaviour of managed systems.

  > *Management is fundamentally about deciding and delivering behavior. We want to model and manage the behaviors of hardware, software, and even users within a system. Without the ability to make predictions about behavior, we cannot make service guarantees*[29]

[28] Stiller; 2007.
[29] Stiller; 2007.

The discussion is about how the previous data models such as MIB, have been far from successful in modelling behavior. MIBs are rather models for registering the configurable parameters of existing hardware and software.

> *The effort to model data is based on an unwritten assumption that configurations correspond to behaviors, that knowing the attributes that are programmed into a device is sufficient to learn what it will do (at some appropriate level of approximation). Unfortunately, this is incorrect except for the simplest automata*[30]

The conclusions is that future research is required to investigate this subject and that promise theory seems to be an interesting tool for this.

The article also mentions that one of the most recent protocols for network management, NETCONF (published in 2006), commenced from another, similar workshop. That workshop was unlike the focal workshop of the article, not focused on research, but rather standardization. The workshop was organized by the Internet Architecture Board (IAB) with the purpose of guiding IETF in their work on standardizing network management protocols, which later paved the way for new protocols, including NETCONF.

[30] Stiller; 2007.

Chapter 5

Analysis of The Prototype

After analyzing our data, we applied our learnings by building a prototype of the system. This chapter analyzes said prototype. Results are analyzed in aspects of how well the design decisions, that was favored as a result of the analytical research and the interview results, worked in practice.

5.1 Implementation Results

5.1.1 System Design

- **Communication Protocol**

  The protocol for communication between network components is SNMP. This introduces the tasks of:

  - Receiving SNMP-Traps (UDP messages)
  - Parsing UDP-messages to a human-readable format.
  - Sending SNMP-Requests (UDP messages)

- **Presentation Format**

  The NOC is developed as a responsive web application. This comes with typical concerns when developing websites:

  - Encryption of network traffic
  - Authorization and authentication
  - Routing
  - Mapping URL to resources
  - Presenting data in a format that the browser can interpret

- **Monitoring hierarchy**

  The prototype follows the monitor hierarchy in figure 5.1. This is a direct result of the environment at MIC Nordic and our design decisions.

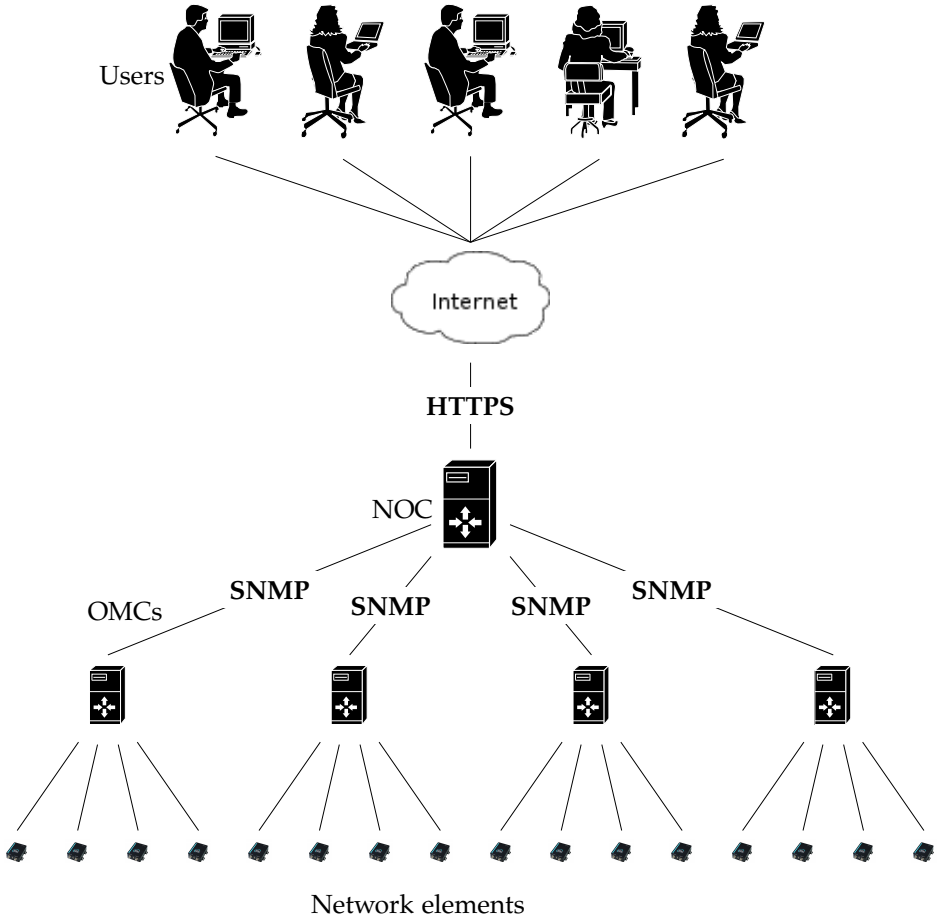

Figure 5.1: Monitor hierarchy of the NOC prototype.

5.1.2 GUI Design

The GUI design of the application's main page is shown in figure 5.2.

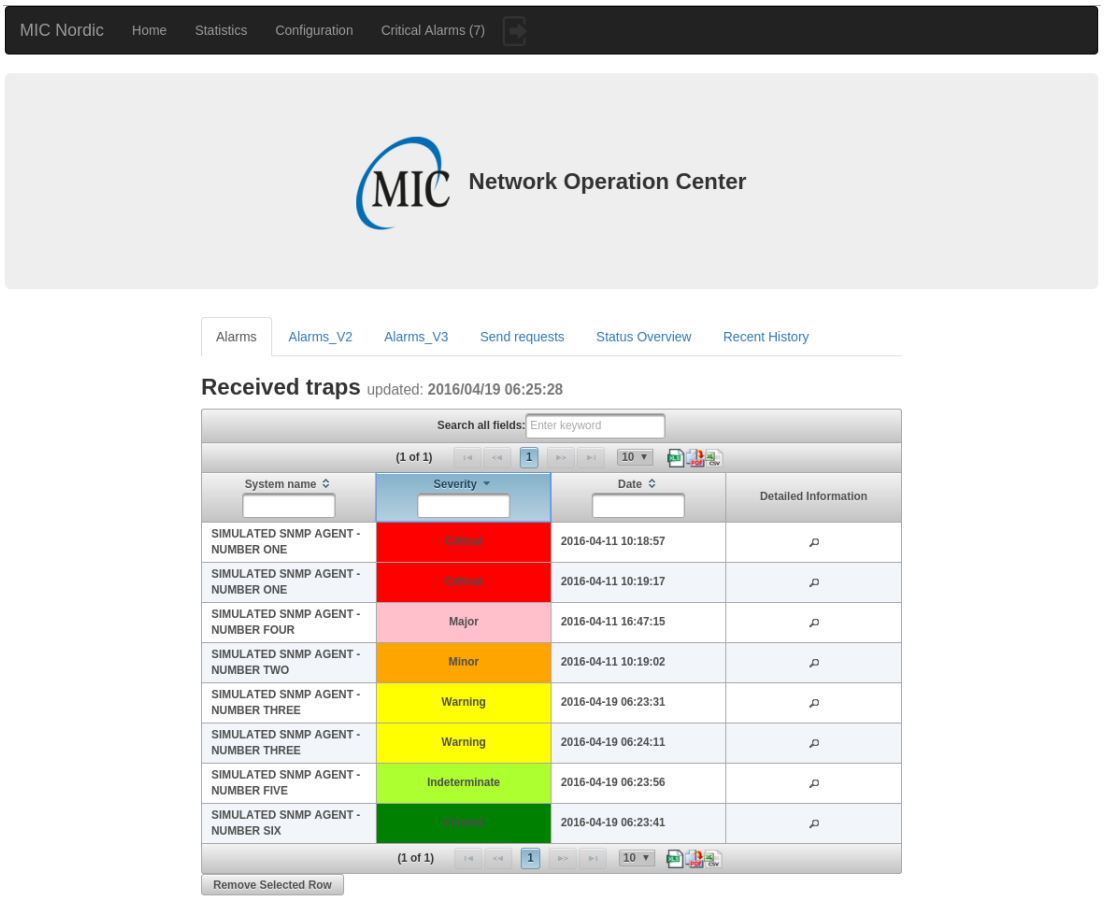

Figure 5.2: Screenshot from the NOC prototype

In figure 5.2 the version of the GUI that was considered the most effective way of presenting the management information is shown. In appendix B you can find detailed evaluations of the GUI and comparisons between different practices of presenting the information.

The perks of this version of the GUI are:

- **Color coding**: By utlizing a color coding that takes advantage of Gestalts Law of Isomorphic Correspondence, users can distinguish alarms by their severity in a practical way.

- **Sorting and Filtering**: By enabling users to sort and filter the data it's possible to focus on certain aspects of the information. This empower users to get an overview even when the volume of alarms are relatively large.

- **Small Information Space**: At first sight, the information presented is

sporadic. Only the most relevant information is displayed. By doing this, the user can grasp the information quicker, which is desirable in the context of presenting network management information.

In spite of the benefits by presenting the alarms this way, we can conclude after our research and evaluations that the GUI still fall short in some aspects of displaying an overview of the status of different subsystems. Hence in our prototype we complement the table of alarms with a page solely dedicated to presenting the status overview(figure 5.3) as well as a page for statistics (figure 5.6).

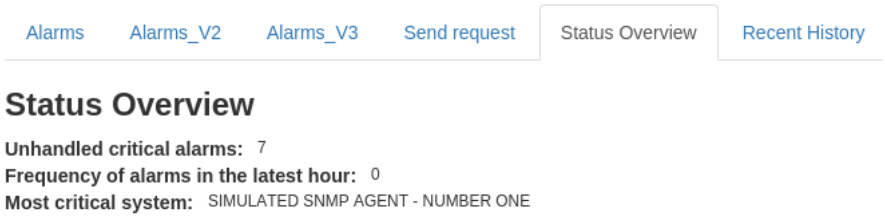

Figure 5.3: Screenshot from the NOC prototype on the status overview page

5.2 Analysis

5.2.1 NOC as a Web Application

After investigation on how to structure the NOC, a web-application was preferred as a result of the requirement that the NOC should be usable on different screen sizes and devices. Another approach would be to develop native applications for each device, this approach gives the possibility to in depth customazitation of the GUI for each device. We have considered the web-application approach to be more sustainable considering that it requires less development costs and that the application can, as new devices are constructed, be adjusted to support it.

An advantage with our approach is that the core SNMP-functionality that is embedded in the web application can be reused for different implemenations, e.g. a mobile application or desktop application. It is not platform dependent, which makes for an uncomplicated extension of the system to another platform.

5.2.2 Language For Implementation

From our research and the background of implementing this prototype we have theorized that the programming-language and build environment for implementing a NOC for alarm management purposes does not have a significant impact on its development and final capabilities.

Being that SNMP uses UDP for its underlying transport protocol, SNMP messages are contained within UDP-packages. This imply that SNMP messages can be parsed on a low-level and that any programming environment with access to regular socket communication is sufficient. For the prototype in this study we utilized this and implemented our own SNMP-parser. Another alternative that was considered were to use existing high level libraries for parsing the UDP messages. We found that implementing the parser required relatively low effort and time investment. Hence using our own parser was preferred, seeing that it facilitates customizability and promotes a design with less dependencies.

The prototype that this chapter concerns was developed in Java. The choice of Java in this case is a consequence of our previous knowledge in the language, and that Java offers a complete infrastructure for developing enterprise web applications.

5.2.3 Parsing SNMP-messages

The essential part of the communication between the NOC and network components is initiated from the network components and then received by the NOC. The received chunk of data then need to be parsed in order to extract neccessary information.

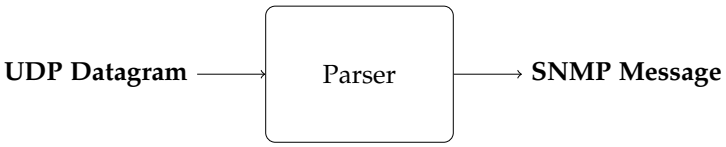

Figure 5.4: Parsing UDP Datagrams

The process of parsing UDP datagrams have been implemented as a step-by-step practice, where each byte in the datagram is examined in order and compared against the Basic Encoding Rules (BER) of the ASN.1 standard. A typical SNMP Message is depicted in figure 5.5. The sought data are kept in so called “Variable Bindings” which are OID-value pairs. These variable bindings depends on what source that sent the SNMP Message

and what type of data the message contains. The Variable Bindings in fig 5.5 are just an example.

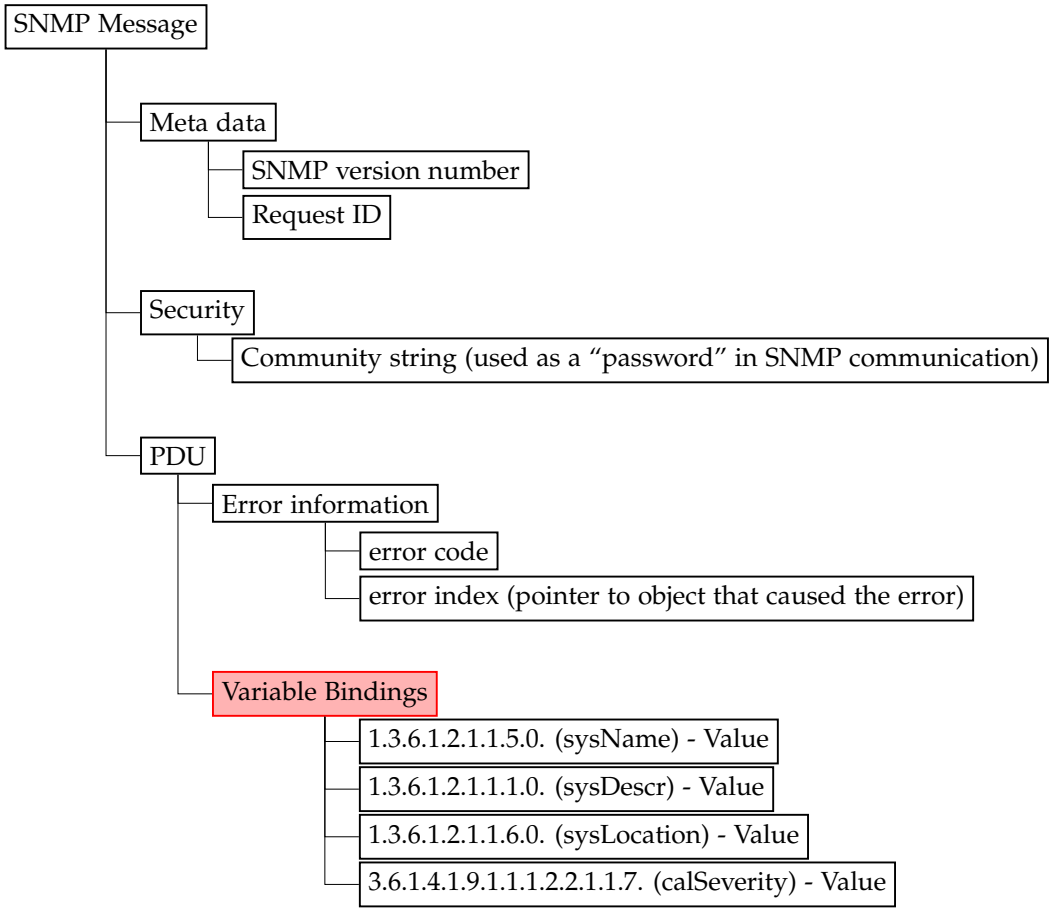

Figure 5.5: SNMP Message

5.2.4 Security

Being that the system is a prototype and not a production ready application, the development process have not been dedicated to implement advanced security techniques. Nonetheless, security aspects have been considered, the main considerations are listed in table 5.1.

Security Aspect	**Solution**
Authentication	User credentials are encrypted and stored in the database
Authorization	HTTP-Sessions, long random float numbers as session-ID to cease "guessing" of ids
Cross-site Scripting	Escaping special characters, never insert untrusted data in the view
SQL-Injection	Pre-compiled SQL-statements and separation of code and data
Communication browser - server	Encrypted communication with HTTPS
Communication server - OMCs	SNMPv2 and SNMPv3, both versions should be supported, for full security the OMC need to use SNMPv3

Table 5.1: Security Aspects

Since not all of the OMCs in place support SNMPv3, it cannot be expected that traps forwarded to the NOC are encryptable. The security implications of this can be examined by looking at the content of the messages and the environment in which the messages are sent. The messages will be alarm data, concerning information of possible hardware failures across the monitored network. Consequently, the data is considered to have a sensitive nature. To increase security the network can be structured in such a way that the OMCs and the NOC exist within the same local network. This means that only the NOC need to be exposed to outside networks. The effects of this is that even if it cannot be assumed that the OMCs encrypt their communication, the data can be encrypted from a single exit point of the local network. This solution can "cover up" the security caveats of using SNMPv2, under the assumption that the NOC and OMCs reside within the same local and private network.

5.2.5 Applying Human Perception Theory

The presentation of alarms is the aspect of the GUI that we mostly focused on during the implementation, see figure 5.2. In particular the theory of human perception have been applied to achieve a design that facilitates sense-making of a large amount of data in a short amount of time. The Gestalt laws that have been found useful and how they are utilized in this prototype are presented in the list below.

1. **Law of Similarity** - According to this law, similar object will be counted as the same group and this technique can be used to draw a viewer's attention[1].

   The law of similarity has been applied by color coding the alarms. This means that when a user is looking for alarms of a specific severity, the user can understand to look for alarms of the color associated with the severity. This also makes it convenient for the user to acquire an overview of all the alarms of that severity in the table of alarms.

2. **Law of Proximity** - This law states that items placed near each other appear to be a group[2].

   When implementing sorting functionality of the collection of alarms, this law have been used to decide upon what parameters to sort upon and how to present the sorted collection.

3. **Law of Isomorphic Correspondence** - Humans interpret the meaning of images based on our experiences[3].

   This relates to the process of designing colors and shapes for displaying alarms in the GUI. A typical example of this that have been considered in the design is that the color green and the color red have predefined meanings for many people based on their prior experiences.

5.2.6 Producing Statistics

The availability of statistics was a requested feature for this NOC. The data set that the statistics are built upon is the total set of received alarms. The prototype produces the following statistics.

- *Number of alarms per day.* This includes all incoming alarms that the NOC receives and parses. Useful to observe trends.
- *Number of alarms per severity.* This gives an overview of the distribution of alarm severity.
- *Source of alarms by system.* Present an overview of how the alarms are distributed among different systems (OMCs).
- *Alarms per hour in recent time.* Provides feedback on the current load of incoming alarms.

[1]Dempsey Chang and Tuovinen, 2002.
[2]Dempsey Chang and Tuovinen, 2002.
[3]Dempsey Chang and Tuovinen, 2002.

We found that access to these statistics greatly helped in getting a complete overview of the current system status. Returning data from the complete set of alarms returns a more exact view of the status than a manual guess from the table of recent alarms. Also, this places the data in contrast with other time periods, giving additional information to the user. The NOC could in the future be extended to include a query system for more specific situations.

Figure 5.6: Screenshot from the NOC prototype that demonstrates how statistics are presented

5.2.7 Persistence

Considering the probability of occational system restarts, the system should be able to recover the list of alarms as the system starts up. Persistence is therefore an important aspect of the NOC, that introduces a few design decisions. After discussions with MIC Nordic about the usage of the NOC and how the alarms that their current network equipment transmits, we have concluded to adopt the following persistence strategy for the prototype:

- Following the X.733 standard for classifying alarms, alarms with a severity equal or higher than “Indeterminate” is persisted in the database until the alarm instance is marked as “Cleared” by one of the users of the NOC.

- Alarms with a lower severity than “Indeterminate”, namely alarms with the severity “Cleared” are not persisted, but instead reserved in a temporary storage that works as a First-In-First-Out (FIFO) queue, and will be updated when new alarms arrive. These types of alarms are just for general information and is mainly valuable to confirm that the network component is functioning properly. Further more, the “Cleared” alarms have little value when they’re not recent, which motivates our decision to not persist these type of alarms.

This is the default strategy for persistence in the prototype, there is however possibilities to customize the settings through a configuration page in the application.

5.2.8 Configuration

As for the configuration of a NOC, there is a design decision to make regarding how the NOC should allow for configuration through the GUI or by a nature of command line tools. Enabling configuration through the GUI means extra effort for the implementation but brings benefits in terms of usability and accessibility, hence there is a trade-off to consider. In the context of this study we have found it to be advantageous to allow for configuration through the GUI. Our research about the future users and typical use-cases have recognized that such configuration might be necessary rather often, making it worthwhile to compose this process as convenient as possible for the user. We believe that the predominant configuration aspects for a NOC are:

- **Add or delete an OMC**. This subject is reviewed in section 4.1.2.2. The NOC should allow for extending or reducing the set of OMCs to monitor.

- **Configuration of existing OMCs**. OMCs have individual properties that should be possible to manage and configure through the NOC.

- **Saving policies of alarms**. As was mentioned in section 5.2.7, the NOC follows a policy for persisting received alarms. This policy should be configurable through the NOC.

5.2.8.1 Adding OMCs

When adding an OMC to be monitored, and to configure it from the NOC, the NOC require the following information about the OMC to be submitted.

- *OMC Name* - Displays as the source on the alarm list.
- *IP-address* - The network address of the OMC.
- Response Message Bindings - List of variable bindings in case a response message must be sent upon receiving an alarm.
- *Ack-Message Bindings* - List of variable bindings to send to the OMC when an alarm has been acknowledged.

5.3 Formative Evaluation of the Prototype

The prototype was developed in iterations during the implementation process. Both functional and non-functional requirements were evaluated continously. At the end of the development process, a comprehensive heuristic evaluation of the GUI was performed, see appendix B for the results of that evaluation.

As we initially underwent the heuristic evaluation, we found some limitations in how you would find specific data in the table of alarms. We then addressed this issue by adding a way to filter out entries in the table. This serves as an example of how design evaluation has fed into the design of a new iteration during development. Furthermore we can use the results of the evaluation to try and understand the important aspects of a GUI for a NOC through the parts of our design that were less successful.

One of the issues identified in the evaluation was that the GUI made it much more difficult to easily recognize the source of an alarm than it was to recognize the severity of the alarm. We concluded that this was a consequence of severities being color coded while the sources were not. One approach that we have looked into in order to cope with this issue is to assign a shape to each OMC. By doing this the GUI would communicate a clear separation of the alarms by their source and allow users to recognize elements by shape in the table. However, such an approach would also present supplementary issues. Every user would need to associate each shape with its corresponding OMC. While this association also holds true for the alarm severity, there are cultural expectations in place regarding the color coding of alarms which helps alleviate this problem. The NOC

also needs to be scalable in regards to the amount of OMCs it can manage. The shape association approach loses effectiveness with more OMCs in the system. Given the disadvantages that this apporach brings, we have chosen not to use it in our final GUI design. Another approach considered was to implement color coding for the sources, however this was after a few experimentations concluded to be a bad idea. Mainly because of the clash with the colors used for coding the severities.

Another issue has to do with presenting larger datasets. A table view was found to be sufficient for smaller datasets but not in the case of massive amounts of alarms. We found through our evaluations that if the amount of alarms grew too big it would become troublesome to get a overview of the information.

Our initial objective when we undertook the design experiments was to have the GUI communicate all the information through a single view. However, from the evaluations we found that there is a limited amount of data that can be presented through a single view. Thus in this experimentation process we examined other styles of presenting the information that would be more suited for larger datasets. For instance a style of using multiple tables to present the information. Still, we found that presenting the information in multiple tables introduced other undesirable deficiencies that made it inferior to the style of presentation in figure 5.2, complemented by two other views, one for statistics (figure 5.6) and one for status overview (figure 5.3). For a comparison of the different presentation styles see appendix B.

5.4 Implementation Challenges

SNMP have prevailed to justify its name of being simple. Although requiring some substantial knowledge, SNMP was relatively effortless to implement on top of regular network communications. The main challenge that we have found with implementing SNMP is to encode and decode the SNMP data in a convenient way. As was discussed in section 5.2.2, when implementing the encoding and decoding you can either utilize existing open-source libraries or construct your own parser. What is beneficial depends on your situation.

Our findings from the practice of implementing this prototype are similar to that of related research in that we concluded SNMP to be the most qualified choice as an communication protocol for network monitoring. Also in terms of our experiences of using SNMP there are many similarities with thoughts expressed in previous research and literature. After

the practice of implementing this prototype we have the perception that there is room for new communication protocols to be adopted in the space of network monitoring, where SNMP currently is so dominant. Yet, we have not got the impression that there are any trends of replacing SNMP at present. The previously declined attempts to replace SNMP is likely a contributing reason. Also it might be worth looking at protocols like NETCONF, that are designed to complement SNMP, rather than replace. NETCONF takes an approach that retain the benefits of SNMP in terms of monitoring and support, while still opening up the possibility of using the modern NETCONF protocol for tasks where SNMP is restricted, e.g. configurement.

From the experience of building this prototype of a NOC, we found that the main difficulty regarded presentation of collected information through a GUI. This was in some sense unexpected to us. In the related sources it is not added particular emphasis on that aspect of network monitoring. However in the article summarizing the current research in the field the subject was brought up and declared as a part of network management research where improvements are necessary[4].

[4]Stiller; 2007.

Chapter 6

Discussion

This chapter presents a discussion on the study and our findings. A substantial part of this chapter is devoted to our interpretations and opinions on the collected results and how the results answers the problem statement from section 1.2.

6.1 Our Methodology and Consequences of the Study

The overall purpose of this study was to investigate the best solution for integrating a specific set of monitoring systems that are in use at MIC Nordic to a NOC, which essentially can be boiled down to the problem statement phrased in section 1.2.

The study sought to present a complete insight on the process of implementing a monitoring system and the aspects involved, which then would make for a meaningful contribution to the industry. We believe that the results of this study, by being practical and comprehensive, fills a gap in the previous research in the field which from our perception takes a more theoretical approach.

Applying a practical approach combined with qualitative research methods to answer the problem statement have been challenging in that the implementation of a monitoring system is very dependent on the implementation-environment. When deciding upon communication protocol, system implementation, presentation of data and the like, the type of network components to be monitored, the data to be monitored and other aspects and constraints play a central role. This is perhaps why there is a gap in the previous research when it comes to the practical aspect. This have also raised the need for this study to be delimeted to the environment of MIC

Nordic.

Our ambition with the study has been to acknowledge the previous research in the area and use it to accumulate a strong base that can be extended upon to derive insights that can be applied to farther fields of network management than those that this study was delimited to.

The study was conducted in two phases. The first phase - *Data collection*, involved analytical research and studying of documentation and related sources. This research was conducted in the early stages of the project. During the analytical research, key issues and aspects that comes with development of a monitoring system were discovered. The data collection phase also included interviews with employees of MIC Nordic. These interviews were performed in a halfway stage of the project and gave a better knowledge of the environment where the system was to be implemented and the users who will be using it.

A considerable amount of time in the data collection pahse was devoted to reading documentation of the systems at MIC Nordic that the NOC were to integrate. The documentation of the systems were neccessary to properly understand the environment we had to work with and how the existing systems related to the requirements from MIC Nordic. From a scientific perspective, the outcome of this approach might have led us to focus too much on the aspects related to those systems. Initially, little regard was given to other communication protocols than SNMP since that was the one supported by the systems at MIC Nordic.

The second phase - *Implementation*, consisted of development of a prototype of the system. This phase was conducted in the final stage of the project. Although being narrowed to certain technologies, the prototype was developed to collect empirical evidence to justify beliefs and understandings acquired through the analytical research. The implementation phase was useful in that it gave us concrete results from areas where we previously were limited to theoretical knowledge.

A potential disadvantage with our approach for implementation is that we were restricted to a specific way of developing the prototype, inherited from our knowledge and experiences from previous development projects. We developed the prototype as a three-tier architecture web application and we found strong motivations for doing it that way. However, we did not consider if architectures or approaches existing outside our scope of knowledge could have been useful in this situation.

After conducting this study and reflecting over our methodology, we have come to the conclusion that network management and SNMP is a broader field than we first expected. There is lots to learn in order to implement a

fruitful network monitoring system, at the same time there exists a great collection of work that you can learn from.

We believe that the effort currently necessary to implement a network monitoring system could be vastly shortened by making the source and documentation of such systems accessible. We consider our results to be a substantial contribution in that sense.

6.2 Problem Statement Revisited

In this section we review our conclusions and by looking back at the problem statement stated in section 1.2, that has permeated this study, we evaluate our results.

6.2.1 Communication Protocol

How can four different Operation and Maintenance Centers (OMC) communicate with a Network Operations Center (NOC) and how can that NOC communicate with the four different OMCs?

6.2.1.1 Discussion and Conclusions

Many issues need to be considered when implementing a NOC. Deciding how the communication between the NOC and other network components should be designed is likely the most critical decision since it requires a certain commitment. The chosen communication protocol affects both the NOC itself and existing network components as well as future installments of components.

In our analytical research we found that there isn't as many established options for communication protocols as one could expect. As of today the Simple Network Management Protocol (SNMP) is the *de facto* standard for network management. The conclusions from many related studies is that the simple nature of SNMP is the main factor that have made the protocol outlast other, more complex and resource demanding protocols. Feasible options that we have found while doing this study is CoAP, CMIP, NETCONF, CORBA and Vicinity Sniffing. All whom fell short in comparisons with SNMP in the context of this study. As a result of the environment where the network management system where to be implemented and various other reasons discussed in chapter 4 and 5, SNMP is our answer to the question formulated in the problem statement. However, it is neces-

sary to note that the delimitations of this study have caused us to spend a disproportionate amount of time looking at SNMP compared to its alternatives. This could be a potential source of bias in the result.

We have found that SNMP met every requirement that we faced when implementing this NOC. However we assume that if it weren't for the fact that SNMP is so convenient by being widely adopted, more modern protocols, like CoAP, could be real contenders when deciding upon a communication protocol. Also for systems more weighted towards configuration than monitoring, a protocol like NETCONF is worth looking at as a complement to SNMP. If we were to attempt to predict the future we would expect that SNMP will be the dominant protocol for a forseeable future but that more modern protocols will, as they get more adopted, be as good of choice if not better due to their more modern design.

6.2.2 Security

How can the communication between the NOC and the OMCs be done in a secure way?

6.2.2.1 Discussion and Conclusions

As a consequence of favoring SNMP as the communication protocol and designated the system design as a web application, this question relates to security aspects of SNMP as well as general web security. Web security is a whole subject on its own and not in the scope of this study.

Our understanding from the analytical research as well as the implementation of this study is that SNMP have historically offered very little when it comes to security. Yet this inadequacy by SNMP when it comes to security have been acknowledged and the most recent version of SNMP (version 3) addresses the security shortcomings of previous SNMP versions. The alternative communication protocols that were considered all had equal if not better security support than SNMP.

It was quite a revelation to us to see that SNMPv3 have been around since 1999 (although not accepted as a internet standard until 2002[1]), considering that our impressions are that SNMPv2 is still in extensive use, despite its security shortcomings. This could be an indication that in many cases, the type of information that is transferred in network management is just regular status updates that have no direct usage of strong security and encryption. Also, as was discussed in section 4.1.1.3, it is possible to have

[1] J. Case and Stewart, December 2002.

coexisting SNMPv2 and SNMPv3 devices in a network monitoring hierarchy.

The conclusion we can draw is that modern versions of communication protocols for network management provides strong security mechanisms that is sufficient for implementing secure network management. If you are to use SNMP as your communication protocol you are recommended to use version 3 over the older versions.

In our pursuance of secure communications we also found that by selecting an architecture of your network that limits your SNMP-communication to reside only within a local network, a greater control over the communication is achieved. This can make usage of early versions of SNMP acceptable since it is not necessary to expose unencrypted data to outside networks (like the internet).

6.2.3 Presenting Data in a GUI

How can information from four different Operation and Maintenance Centers (OMC) be present in a Graphical User Interface (GUI) and how can the GUI allow for manipulation of each OMC in a user-friendly and responsive way?

6.2.3.1 Discussion and Conclusion

As a consequence of the interview results, were it was stated that the NOC preferably should be accessible from multiple different workstations, including phones, tablets, laptops and desktops, we chose to develop the NOC as a web application. With the NOC being a web application, we were able to adjust the GUI to be responsive and usable on all the different devices stated above. However, the NOC developed is just a prototype and if the users feel that a web application is not sufficient and it would be advantageously with native applications for each device, that is something that could be considered and would allow for reuse of much of the core functionality that the web application already has.

Presentation of information is an essential piece of a fruitful network monitoring system. As of today we haven't yet reached the point of autonomic networks, network monitoring still require the human factor to be useful. If network administrators can't comprehend or manage the interface of the network monitoring system it doesn't matter how well setup your actual monitoring is, the system wont be of much use.

To develop a user interface in a user-friendly way is a complex task and there are a lot of theories and research on the subject. In the early stages

of the analytical phase of this study we concluded that the most important aspect, apart from general best practices when it comes to interface design, is to provide a high-readability of the data that the NOC presents. For the task of presenting the data in a effective way we took a perspective of human perception, were we utilized previous studies on humans perception. A much valuable source for this process was, among others, Dempsey Chang, laurence Dooley and Juhani.E Tuovinens paper *Gestalt Theory in Visual Screen Design*[2].

The resulting GUI of the NOC came to apply Gestalt principles of perception. Essentially this ment to promote the use of colors, grouping, shaping and humans previous experience, to design the presentation of data (alarms) in the GUI.

Our understanding is that it doesn't exist much previous work in the area of GUI design specifically for NOC's. To reach conslusions about the topic we performed our own experiements and utilized previous research about general GUI-design. During development, various ways of displaying the management information were tested and compared in terms of how efficient the information is communicated to the user. From those evaluations we found that when the amount of information reaches a certain size, the interface need to prioritize parts of the information. Otherwise it's not possible for the user to grasp it. Our conclusion came to be that the best solution is to enable users to swap the focus between distinct parts of the information. This way the interface allows the user to quickly get an understanding of the alarms, even when the information volume is large. We also concluded that there is only as much information that can be communicated by presenting the alarms. In order to provide complete overviews of the system status it is desirable to have separate views that are dedicated to that purpose only.

Finally, our answer to the question formulated in the problem statement is to develop the NOC as a web application, to utilize theories on human perception when presenting data, and making your design decisions based on research about the user.

6.3 Summative Evaluation

This section evaluates wether the purpose of the study, stated in section 1.4, have been fulfilled.

[2]Dempsey Chang and Tuovinen, 2002.

6.3.1 Fulfillment of the Objectives of the Study

Our results provide a complete overview and suggestions for implementing a NOC for the company MIC Nordic. We believe that the problem statement have been resolved in a comprehensive manner and we are confident that MIC Nordic can benefit from our results.

When critically evaluating our results we can conclude that our solution is weighted specifically towards a solution for MIC Nordic, since that was the focal point of the problem statement. However we do still consider our results to be an evident contribution to the industry. Let us elaborate on that.

While our solution is delimited to technologies that are accessible for an implementation at MIC Nordic, the study as a whole includes an analysis of important design choices you are faced with when implementing a NOC for alarm management. As well as presenting a compilation of important aspects that regards network monitoring. Hence our ambitions of providing a general solution are partially achieved.

6.3.1.1 Problem Relevance

The problem is of high relevance due to two factors:

- The problem is designed after needs of a company that can benefit from the solutions in their business.

- The problem tackles issues of network monitoring, which is a general and actual problem that concerns many research areas, peoples and companies.

6.4 Ethical Aspects

The leading actors to regard when it comes to ethical aspects is MIC Nordic and their customers and suppliers. To carry out this study a presumption was that MIC Nordic provide us with essential documentation on their operations and the equipment they're using. This documentation contains both general and confidential information that has been given to us in confidence by MIC Nordic. The documentation given entails a ethical aspect of maintaining the integrity and privacy of involved actors.

6.5 Sustainability

Network management systems have, just as any interactive systems, impact on the world environment. This study have been carried out with minimal resources and has not explicitly given concerns to environmental aspects. By following a user-centered approach the study aspire to provide a design for network management systems that empower users to use the system more efficient. A successful implementation of the product would implicate that the network can be managed with minimal resources. Which would make for a more sustainable development. This would also implicitly affect financial aspects, seeing that development and maintainment costs for MIC Nordic would be reduced. Additionally, a performant implementation of the system will reduce the required electricity demands and contribute to a more sustainable development for the future.

6.6 Observed Trends

Seing that NOCs have been around since the 1970s[3] and that SNMP was first introduced in 1988, network management can be seen as a fairly stable research area. Still today 2016, a NOC and SNMP as a communication protocol is a recommended way of implementing network management. However as was mentioned in section 4.3, more research is being devoted to the subject.

As long as network management is still on the form as it is today, we would expect the concepts of NOCs and usage of SNMP to continue. Partially because of the legacy factor. There is however a forthcoming topic wether network management is moving towards an autonomic approach. Autonomic network management is an innovative vision promising new horizons of efficient networking systems free from human control[4]. Adoption of the autonomic computing paradigm in network management would hold a promise of network management systems that diagnose and circumvent possible impairments in the functionalities of the underlying network, in an independent and autonomic manner[5].

Autonomic network management would mean big changes to how we view the field of network management. It would likely be an attractive way for companies to be able to manage their networks in an autonomic manner. It would also introduce new ethical and social aspects for the

[3] AT&T, 2016.
[4] Samaan and Karmouch, 2009.
[5] Samaan and Karmouch, 2009.

industry.

However we would not expect such an adoption in recent times. In an article published by the IEEE in 2009, the authors acknowledge the trend and the research being devoted to the subject but still conclude autonomic network management not to be expected any time soon. Although in theory autonomy seems to provide the ultimate solution for the complex network management problem, in general, research efforts towards autonomic network management systems are still in their infancy and many challenges remains to be solved before realizing a successful solution[6].

[6]Samaan and Karmouch, 2009.

Chapter 7

Conclusions and Future Research

This study set out to investigate the best solution for integrating a specific set of monitoring systems that are in use at MIC Nordic to a NOC, while also presenting parts of a general solution with the intent of benefiting the industry at large.

7.1 Contributions

The primary contribution of this study is:

> *Concrete and complete insight on the process of implementing a monitoring system.*

This was achieved by starting from a requirement specification and then by the means of qualitative research methods deciding upon design choices and concluding in a prototype to try out the reasoning in practice.

This process included the following contributions:

- **Guidance in choosing network monitoring hierarchy**. A reasoning of different monitoring hierarchies and what a monitoring hierarchy consists of were given in section 4.1.2.1.
- **Analysis of communication between a Network Operation Center and other components in a network monitoring hierarchy**. One of the primary research questions stated in the problem statement

is the concern of implementing the communication between a NOC and other network components. Specifically we introduced a exhaustive analysis of monitoring communication in terms of implementation, performance, security and reliability. This is a central theme throughout the report and were discussed in various sections, including section 4.1.1.1, 4.1.1.2, 4.1.1.3, 4.1.1.4 and 4.1.1.5. The conclusions and issues discussed are also presented from a practical point of view in the process of implementing a prototype.

- **Implementation aspects**. Discussions about what is a suitable technical environment to implement a NOC and what requirements this introduces were discussed in sections 4.2.1, 5.1.1, 5.2.2, 5.2.3 and 5.4. This aspect were discussed both from a thorough analytical research as well as experience from implementing a prototype.

- **Methodologies for definition and classification of alarms**. A fundamental problem of alarm management systems is how to classify alarms. We defined this problem area and provided methods for classifying alarms and also a demonstration on how the theories presented can be used in practice. These discussions were given in section 4.2.4.

- **Guidelines for presenting monitoring information in a GUI that is responsive and user-friendly**. This is another central theme of this study that relates to one of the research questions. Concerns of presenting monitoring information were analyzed and tested in practice (section 4.2.2, 4.2.3, 5.1.2, and appendixB).

7.1.1 Deliverables

This study was performed in cooperation with the company MIC Nordic and academia. The study included the following deliverables.

- **Scientific thesis**

  Delivered to the academia and the company.

- **Prototype**

  Delivered to MIC Nordic. Includes:

  - Source code
  - Architectural document
  - Test report

7.2 Future Research

The results introduced in this study makes for a natural future research in both the area of network communication and the area of implementation and GUI design.

This study is based upon qualitative research methods such as interviews, analytical research and prototyping. These methods can be complemented by quantitative research methods.

The network communication analysis of this study have been carried out with a delimitation that the communication need to be suitable for integration with the network equipment in use at MIC Nordic. This means that the analysis of network communication can be further complemented by doing a more general and extensive comparison between different means of communicating monitoring information over a network. Hence a logical direction for further research would be to do extensive comparisons by means of conducting case studies, load tests or similar. This would give a more profound knowledge base to stand on when it comes to choosing means of network communication in future development of Network Operation Centers.

Much research remains to be done on the topic of GUI-design of a NOC. The interview results from this study are narrow in that it provides information about users of MIC Nordic specifically. Further inquiries need to be carried out to get a general perception of the users point of view in the context network monitoring. A suggested starting-point would be to extend the interivew method used in this study by conducting interviews with interviewees of different backgrounds and analyze the commonoalities of the answers.

Bibliography

Andersson, Niclas and Anders Ekholm (2002). *Vetenskaplighet – Utvärdering av tre implementeringsprojekt inom IT Bygg och Fastighet 2002*. Tech. rep. Lunds Tekniska Högskola.

AT&T (2016). *History of Network Management*. http://www.corp.att.com/history/nethistory/management.html. [Online; accessed 07-May-2016].

Benyon, David (2010). *Designing Interactive Systems: A Comprehensive Guide to HCI and Interaction Design (2nd Edition)*. 2nd ed. Pearson Education Canada. ISBN: 9780321435330. URL: http://amazon.com/o/ASIN/0321435338/.

Berndtsson, Mikael (2008). *Thesis projects a guide for students in computer science and information systems*. London: Springer. ISBN: 978-1-84800-009-4.

Brian Hedstrom Akshay Watwe, Siddharth Sakthidharan (2011). *Protocol Efficiencies of NETCONF versus SNMP for Configuration Management Functions*. University of Colorado.

Bruey, Douglas (2005). *SNMP: Simple? Network Management Protocol*. Tech. rep. Rane Corporation.

Case, J. et al. (May 1990). *A Simple Network Management Protocol (SNMP)*. English. Available at https://www.ietf.org/rfc/rfc1157.txt. Network Working Group. 36 pp.

Chisholm, S. and D.Romascanu (March 2001). *ITU Alarm MIB*. English. Available at rfc768. Disman Working Group. 18 pp.

Clemm, Alexander (2006). *Network Management Fundamentals*. Cisco Press. ISBN: 1587201372.

COMMUNICATIONS, GLOBAL SYSTEM FOR MOBILE (2007). *Fault management requirements (Release 7)*. English. GLOBAL SYSTEM FOR MOBILE COMMUNICATIONS. 19 pp.

Dempsey Chang, Laurence Dooley and Juhani E. Tuovinen (2002). "*Gestalt Theory in Visual Screen Design – A New Look at an Old Subject*". Monash University.

Enns, R. (December 2006). *NETCONF Configuration Protocol*. English. Available at https://tools.ietf.org/html/rfc4741. Network Working Group. 95 pp.

Gu, Qiang and Alan Marshall (2004). "Network management performance analysis and scalability tests: SNMP vs. CORBA". In: *Managing Next Generation Convergence Networks and Services, IEEE/IFIP Network Operations and Management Symposium, NOMS 2004, Seoul, Korea, 19-23 April 2004, Proceedings*, pp. 701–714. DOI: 10.1109/NOMS.2004.1317758. URL: http://dx.doi.org/10.1109/NOMS.2004.1317758.

Håkansson, Anne (2013). *Portal of Research Methods and Methodologies for Research Projects and Degree Projects*. Tech. rep. The Royal Institute of Technology.

(ITU), International TeleCommunication Union (1992). *Systems Management: Alarm Reporting Function*. English. International TeleCommunication Union (ITU). 18 pp.

J. Case R.Mundy, D. Partain and B. Stewart (December 2002). *Introduction and Applicability Statements forInternet Standard Management Framework*. English. Available at http://www.ietf.org/rfc/rfc3410.txt?number=3410. Network Working Group. 27 pp.

J. Case R.Presun, K. McCloghrie and S. Waldbusser (December 2002). *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*. English. Available at https://tools.ietf.org/html/rfc3418. Network Working Group. 26 pp.

Johnson, Robert Bern (2009). "*EVALUATING THE USE OF SNMP AS A WIRELESS NETWORK MONITORING TOOL FOR IEEE 802.11 WIRELESS NETWORKS*". MA thesis. Clemson University.

Li, Jizong (1999). "*WEB-based Network Monitoring Using SNMP, CG1 and CORBA*". MA thesis. University of Manitoba.

Mauro, Douglas R. and Kevin J. Schmidt (2005). *Essential SNMP, Second Edition*. O'Reilly Media, Inc. ISBN: 0596008406.

MIC Nordic (2016). *MIC Nordic*. http://www.micnordic.com/. [Online; accessed 18-Jan-2016].

Postel, J. (August 1980). *User Datagram Protocol*. English. Available at https://tools.ietf.org/html/rfc768. ISI. 3 pp.

R. Enns M.Bjorklund, J. Schoenwalder and A. Bierman (June 2011). *NETCONF Configuration Protocol*. English. Available at https://tools.ietf.org/html/rfc6241. Internet Engineering Task Force. 113 pp.

Samaan, Nancy and Ahmed Karmouch (2009). "Towards Autonomic Network Management: an Analysis of Current and Future Research Directions." In: *IEEE Communications Surveys and Tutorials* 11.3, pp. 22–36. URL: http://dblp.uni-trier.de/db/journals/comsur/comsur11.html#SamaanK09.

S.Chisholm, D.Romascanu (September 2004). *Alarm Management Information Base (MIB)*. English. Available at https://tools.ietf.org/html/rfc3877. Network Working Group. 75 pp.

Solutions, DeltaNode. *DeltaNode Central Gateway*. English. Version Version 13.10. DeltaNode Solutions. 96 pp.

Southern California, Information Sciences Institute University of (September 1981). *TRANSMISSION CONTROL PROTOCOL - DARPA INTERNET PROGRAM - PROTOCOL SPECIFICATION*. English. Available at https://tools.ietf.org/html/rfc793. Internet Standard. 85 pp.

Stiller; Aiko Pras; Jurgen Schonwalder; Mark Burgess; Oliver Festor; Gregorio Martinez Perez; Rolf Stadler; Burkhard (2007). *Key research challenges in network management*. IEEE Communications Magazine.

Stump, Michael (2003). *Securing SNMP: A Look at Net-SNMP (SNMPv3)*. SANS Institute InfoSec Reading Room.

T. Socolofsky, C. Kale (January 1991). *A TCP/IP Tutorial*. English. Available at https://tools.ietf.org/html/rfc1180. Network Working Group. 28 pp.

U. Warrier L.Besaw, Hewlett-Packard (April 1989). *The Common Management Information Services and Protocol over TCP/IP (CMOT)*. English. Available at https://tools.ietf.org/html/rfc1095. Network Working Group. 67 pp.

U. Warrier L.Besaw, L. LaBarre and B. Handspicker (October 1990). *The Common Management Information Services and Protocols for the Internet (CMOT and CMIP)*. English. Available at https://tools.ietf.org/html/rfc1189. Network Working Group. 15 pp.

Z. Shelby K. Hartke, C.Bormann (June 2014). *The Constrained Application Protocol (CoAP)*. English. Available at https://tools.ietf.org/html/rfc7252. Internet Engineering Task Force. 112 pp.

Appendix A

Interview Results

Interview Questions

With Answers

Kim Hammar
kimham@kth.se

Marcus Blom
marcblom@kth.se

May 3, 2016
Royal Institute of Technology, Stockholm

How many people are using the monitoring systems you have setup today?

3-4 persons.

How often are they used?

Typically daily

How does a typical use-session with the monitoring systems look like and how long is the average use-session?

A typical session is short, to check the status of an alarm or to perform a small operation like restarting a repeater.

What purpose does the current monitoring setup make?

The main purpose is getting an overview of the status of network equipment that are in use and discover defects before customers do.

What type of information does it give you?

Status information of network equipment.

How does the interaction with the system to retrieve this information look like?

Right now each OMC is hooked up to a monitor so to get an overview you would go through each OMC and check their status.

What do you think about the graphical presentation of the information in the current system?

They look professional and do what they should.

What do you think about the usability of the current system?

There's nothing particularly wrong with it.

What would you like to change with the current systems?

The functionality to have the positions of networked nodes be drawn out on a geographical map.

What type of benefits do you see with centralizing the current OMC's to a NOC?

Mainly the centralization aspect. It would make for a more scalable solution in the future if more OMC's were to be added.

What information do you think is most important that the user-interface of the NOC can present?

Which sub-system the alarm comes from.

If the NOC where to put together statistics of incoming alarms, what type of statistics would you consider useful?

- Number of alarms per severity
- Source of alarms by system
- Source of alarms by physical location
- In general, the more statistics we can get the better.

What devices should be able to interact with the system?

It's our ambition that the system can be usable, if not for configuration than atleast for information purposes, on every screensize that's as big or bigger than a typical smartphone.

Will it be the same user group that's using the OMC's that are going to use the NOC?

Largely, yes. But hopefully the new system will be more accessible and user-friendly which might encourage other people to start using the system regurarly as well.

Appendix B

Evaluation Results

Design Evaluation

Kim Hammar
kimham@kth.se

Marcus Blom
marcblom@kth.se

May 7, 2016
Royal Institute of Technology, Stockholm

Contents

1 Heuristic Evaluation

1.1 Overview

This section describes the results of a heuristic evaluation of the user interface of a Network Operations Center used for alarm management. In the investigation for how alarm management information can be presented to the user in the most effective way, three different versions of the GUI-presentation have been produced and then evaluated and compared against each other. As of such the results are presented as follows:

- Common Functionalities - The evaluation results of the parts of the system that are the same for all variations.
- GUI 1 - Results for the parts of the system that are unique to the first variation of the system.
- GUI 2 - Results for the parts of the system that are unique to the second variation of the system.
- GUI 3 - Results for the parts of the system that are unique to the third variation of the system.

1.2 Methodology

The heuristic evaluation was conducted with two evaluators. Each evaluator inspected the interface individually and evaluated in according to the following heuristics:

1. **Visibility of system status**.
   This Heuristic implies that the interface should offer some kind of visible feedback to the user to inform what is going on and that the application is actually responding.
2. **User control and freedom**.
   Since users often choose system functions by mistake the interface should support undo and redo as well as providing obvious ways to go back to the previous state.
3. **Consistency and standards**.
   This heuristic says that words and labels should be used in a way that is expected by the user and mean the same thing across the whole interface.
4. **Error prevention**.
   This heuristic means that the interface should have a design that, to the highest degree possible, prevents errors from occuring.

5. **Recognition rather than recall**.
   This principle says that the interface should be designed in a way that doesn't require the user to remember a lot of information.
6. **Aesthetic and minimalist design**.
   This principle means that the website should not contain unnecessary or redundant information.
7. **Help users recognize, diagnose and recover from errors**.
   Design for errors, construct your interface such that when errors occur it is easy to recognize the error and recover from it.
8. **Sense-making of network management information** This heuristic relates in specific to network management and means that the interface should communicate an overall view of the management information in a way so that the user quickly can get a feel for the status of different systems. It also means that the presentation of specific information should be clear and understandable.
9. **Understanding classification of alarms without prior knowledge in the domain** This implies that the way alarms are classified (e.g. severity or source) should be recognizable and understandable even for people that is not familiar with the network.

The heuristics are a combination of general usability heuristics listed by Jakob Nielsen[1] and custom heuristics that relate strongly to network management. The usability problems found during the evaluation have been rated on a scale 0-4, the scale used is inspired by Jakob Nielsen's paper "Severity Ratings for Usability Problems"[2]:

- **0**: I don't agree that this is a usability problem at all
- **1**: Cosmetic problem only: need not be fixed unless extra time is available on project
- **2**: Minor usability problem: fixing this should be given low priority
- **3**: Major usability problem: important to fix, so should be given high priority
- **4**: Usability catastrophe: imperative to fix this before product can be released

1.3 Results for common functionalities

This section contains the result for all the parts of the UI that remain the same in all included UI variations.

[1] Nielsen, 1995a.
[2] Nielsen, 1995b.

1.3.1 Visibility of System Status

On the alarm-page the GUI provides clear feedback to the user about the system status by providing a label "updated", see figure 1. The "updated"-label informs the user of exactly when the alarm table was last updated. This label is necessary for informing the user that live-updates are being done continuously as well as providing the specifics regarding the exact timings of the updates.

However, since the website offers multiple pages there also need to be a way of communicating system status when the user is viewing other pages. This is achieved with live-notifications, as shown in figure 2.

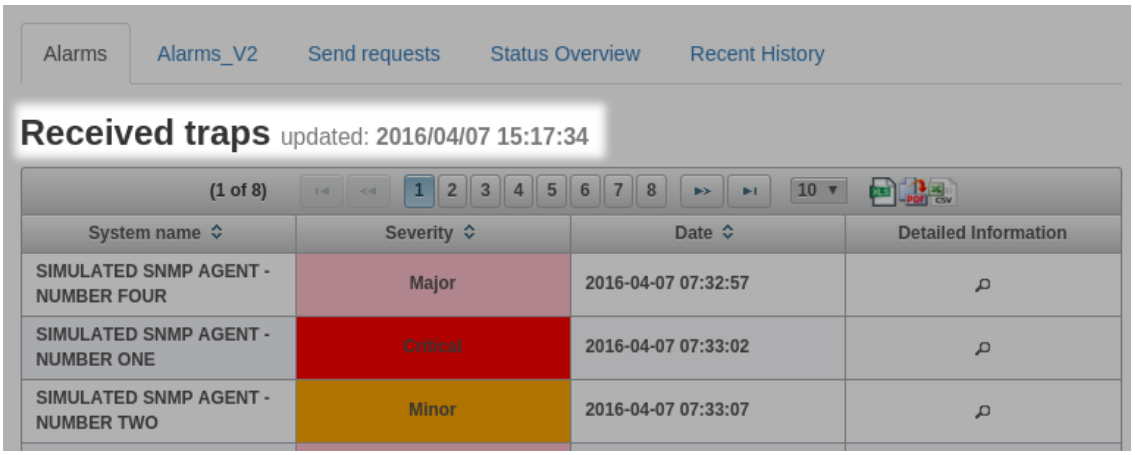

Figure 1: Screenshot from the alarms page

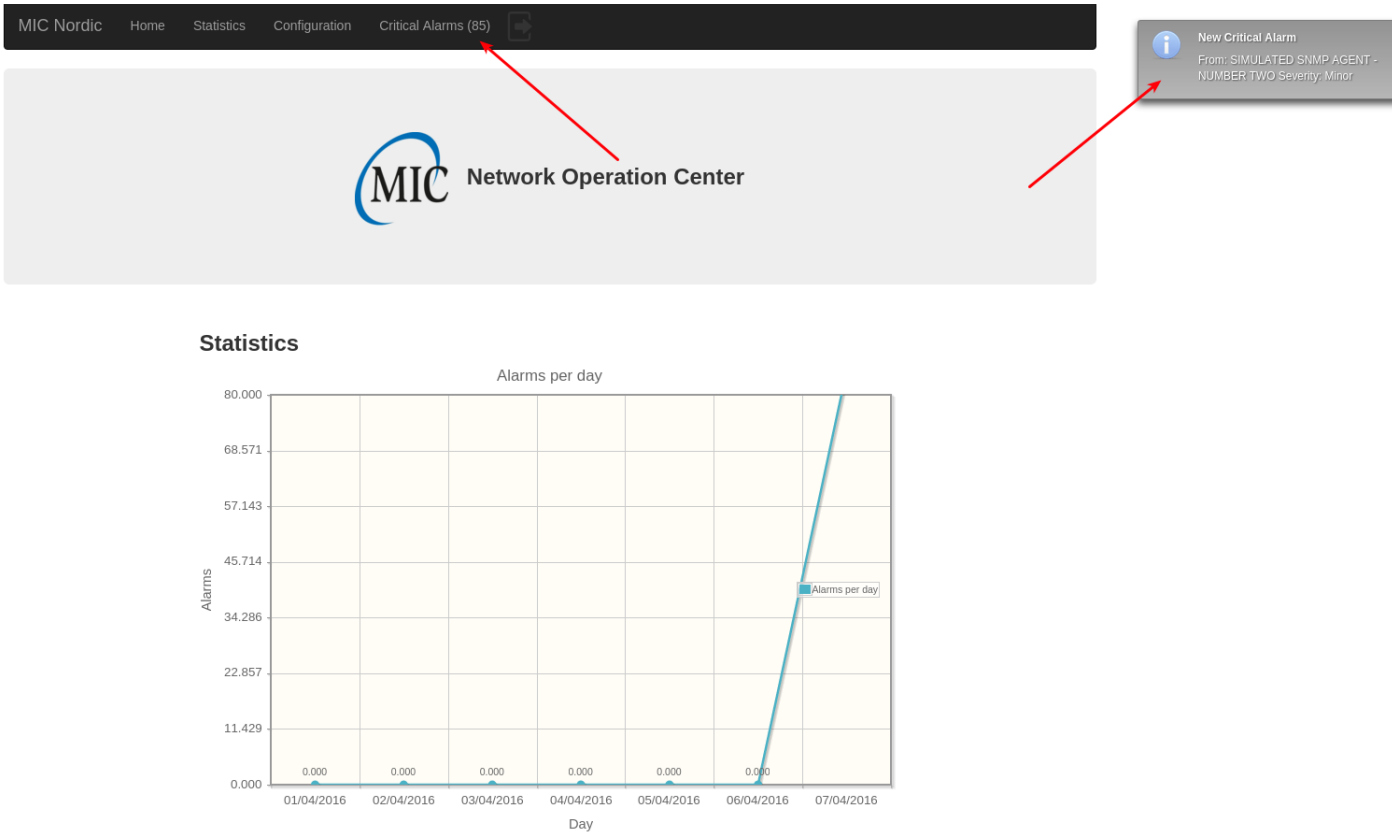

Figure 2: Screenshot that illustrates how system status is communicated to the user when on a different page

1.3.2 User Control and Freedom

For system-states where the browser button can't take you back to the previous state in a satisfactory manner, the interface provides exit-options that are designed to be easily recognizable by utilizing cultural constraints. For example the fact that an x is a universal sign for exit.

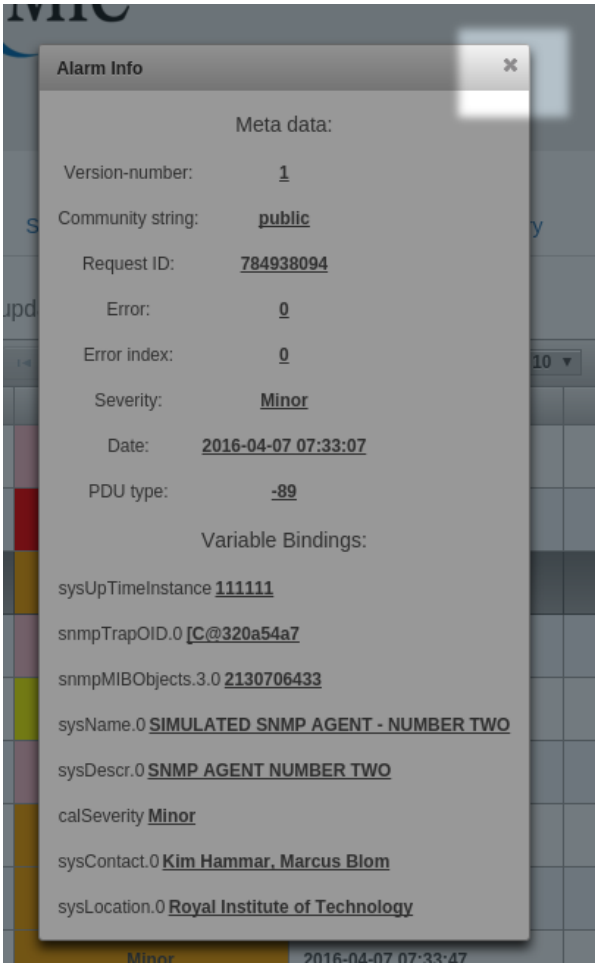

Figure 3: Screenshot that illustrates how user control is implemented

The website utilizes hyper links in order to make use of the browsers general back and forward buttons, this provides undo and redo of general navigation that is consistent with typical web-standards.

1.3.2.1 Problems

Notification bar design Severity: 1.

The GUI provides a notificationbar that drops down as a result of a specific user action, to communicate to the user where to press to hide the notification bar a link labeled with “Hide” is visible, see figure 4. This makes it very clear for the user how to hide it, but you could argue that it shouldn’t be neccessary to explicit write down

the label “Hide” to communicate this. It can be looked over how this could instead be communicated to the user through affordance of some object.

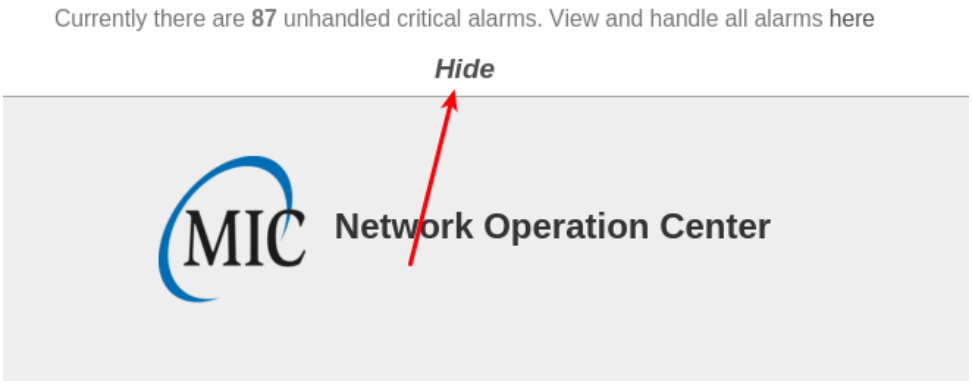

Figure 4: Screenshot that ilustrates how the GUI communicates to the user how to hide the notificationbar

1.3.3 Consistency and Standards

The website uses consistent terms across the whole site and there is no case of the same term meaning different things in different places.

1.3.3.1 Problems

Unnecessary Technical Terms Severity: 3

The application is in first hand aimed towards network professionals and thus it should be considered ok to use network terminologies when neccessary. However some terms that are currently used to label different things are diffuse and should be replaced by more intuitive terms, see figure 5.

Figure 5: Screenshot that shows some of the terms used in the site

The term “send request” is a very technical term and refers to the technical aspect that is performed “under the hood” but is not really relevant to the action that the interface provides for the user. “send request” could be replaced by two labels: “configure network element” and “retrieve status of network element”.

Diffuse Term Severity: 1

The term "Recent History" is not clear and could be replaced by something more explicit.

Important User Action Hidden Severity: 4

The sign used to communicate to the user where to log out is almost camouflaged and should be made easier to spot, perhaps through using a different color, see figure 6.

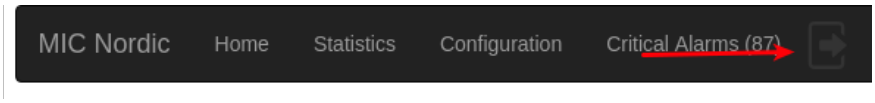

Figure 6: Screenshot that shows the sign used for logging out

1.3.4 Error Prevention

We believe that the interface through its simple design makes it easy to prevent errors. Altough to further minimalize the chance of errors, confirmation dialogs are implemented for every user action that could be erronous, see figure 7.

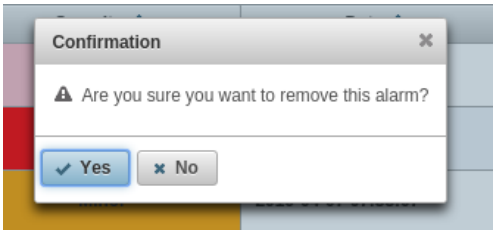

Figure 7: Screenshot of a confirmation dialog

Another mechanism implemented to prevent errors is warning labels, see figure 8

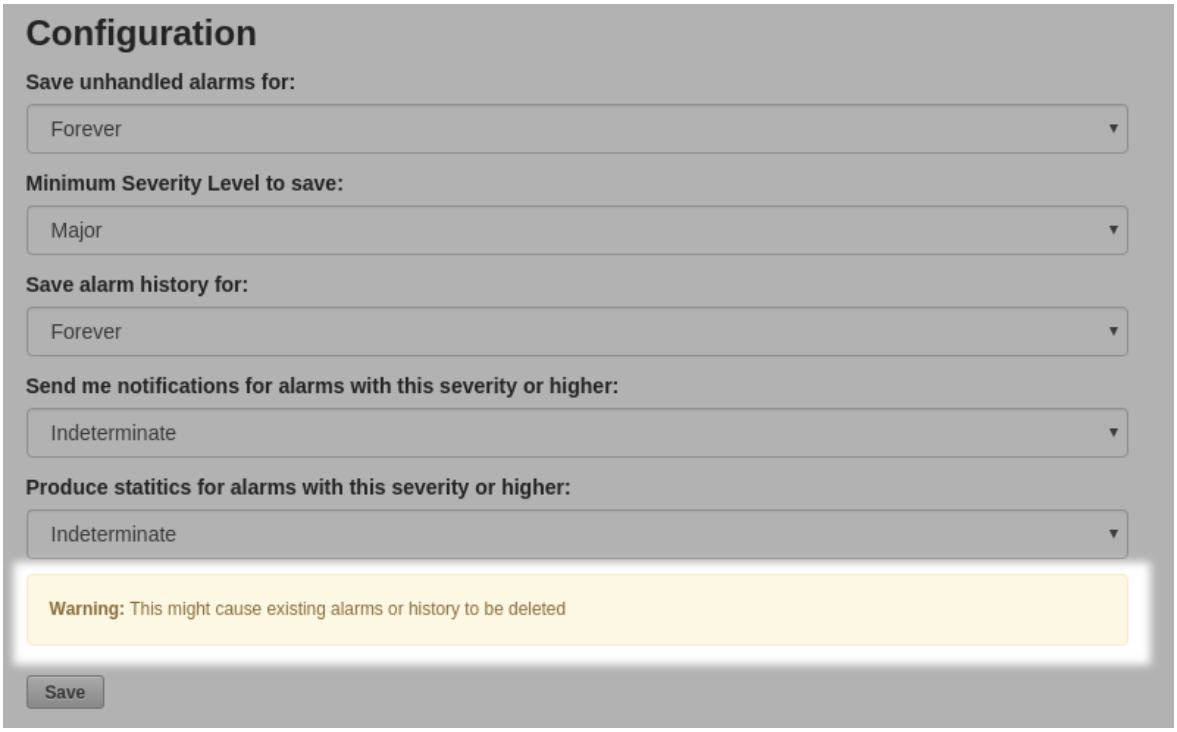

Figure 8: Screenshot of a warning label

1.3.5 Recognition Rather Than Recall

By utilizing colors and cultural constraints for visualizing severities of different alarms the user can through previous experiences recognize that certain alarms are more severe than others by the colors, see figure 9.

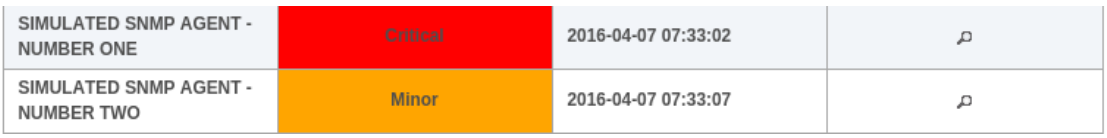

Figure 9: Screenshot of alarms with different severities

1.3.5.1 Problems

Remembering Network Stations/Elements Severity: 3

When the user is to configure or retrieve current status of a specific network station/element he or she need to recall the address of the station/element, see figure 10.

The recognition rather than recall heuristic should be applied here to provide mechanisms to facilitate recognition and not require the user to recall. This could for example be implemented by letting the user insert predefined network stations/elements with given addresses, or at least visualizing previously used addresses.

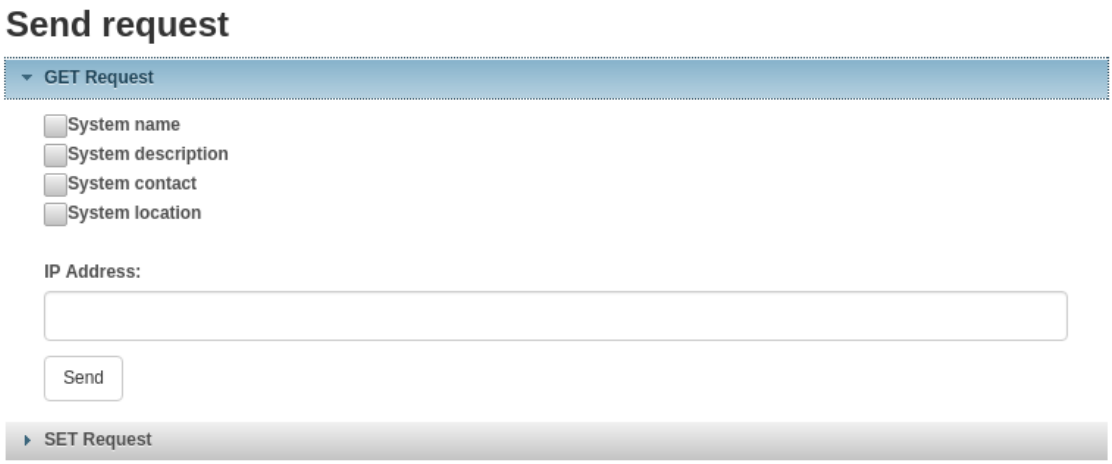

Figure 10: Screenshot that shows how the user need to recall the address of network stations/elements

1.3.6 Aesthetic and Minimalist Design

The interface fulfills this heuristic to the max, there is basicly no text or extra information that is not directly relevant to the purpose of the website. Also we can not find any case of redundant information.

1.3.7 Help Users Recognize, Diagnose and Recover From Errors

This heuristic have been applied by implementing error messages that are easy to understand and that direct the user in how he or she can recover from the error, see figure 11 and 12.

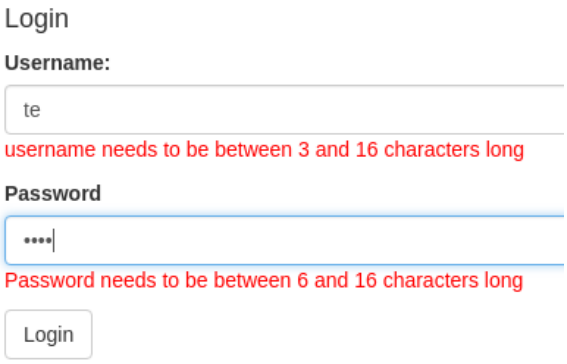

Figure 11: Screenshot that shows error messages to help the user recover from errors

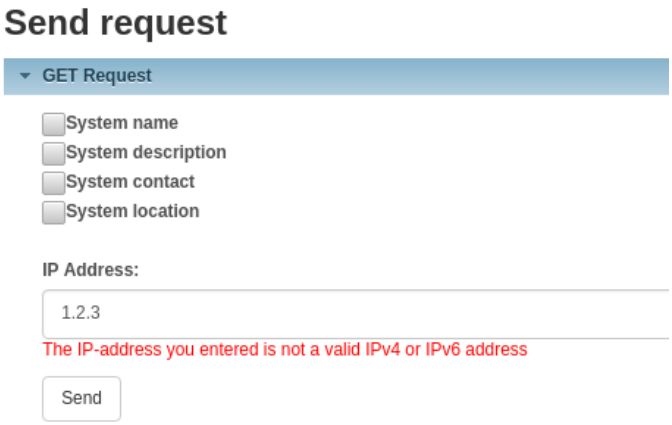

Figure 12: Screenshot that shows error messages to help the user recover from errors

1.3.8 Sense-making of network management information

Detailed information about each alarm can be viewed in separate windows, see figure 13.

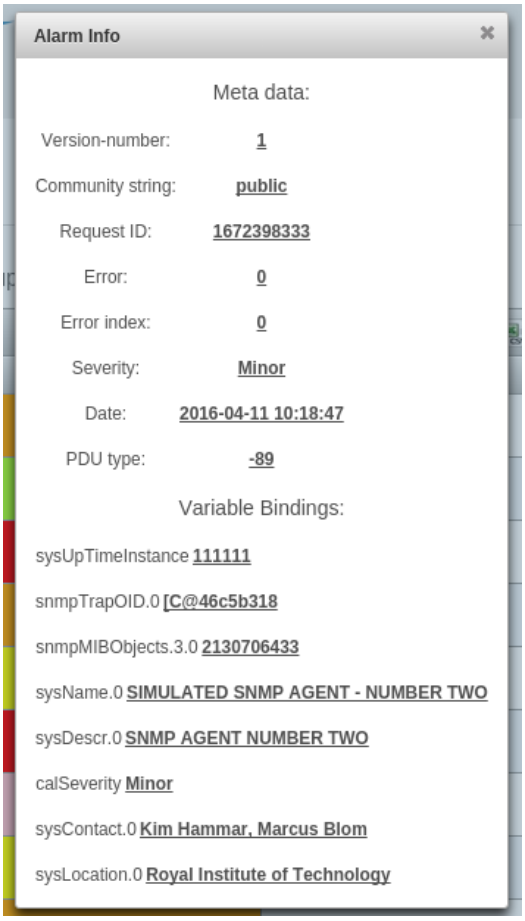

Figure 13: Screenshot that shows how alarm specific information is presented

1.3.8.1 Problems

Alarm specific information is to raw Severity: 3

While the information specific to each alarm is displayed in a complete way where all the information available is presented to the user in a fashion that is comprehensable, it's in a too raw format. The data is presented with their technical terms that are inherited from the underlying protocol (SNMP). The user that interpreters the information should not need to know anything about the underlying protocol, but rather the information should be pre-processed and presented in a more user-friendly manner.

1.4 Results for GUI version 1

1.4.1 Description

This version of the GUI uses a compact way of presenting alarm management information. All the alarms appear in one single table that then can be sorted on different parameters.

1.4.2 Sense-making of network management information

The primary means of presenting network management information to the user is through a data-table that the user can navigate through and if wanted to, inspect specific alarms, see figure 14. An overview of the network status is communicated through a specific page, see figure 15 and through a statistics-page that contains graphs that communicates the system-status over longer time periods.

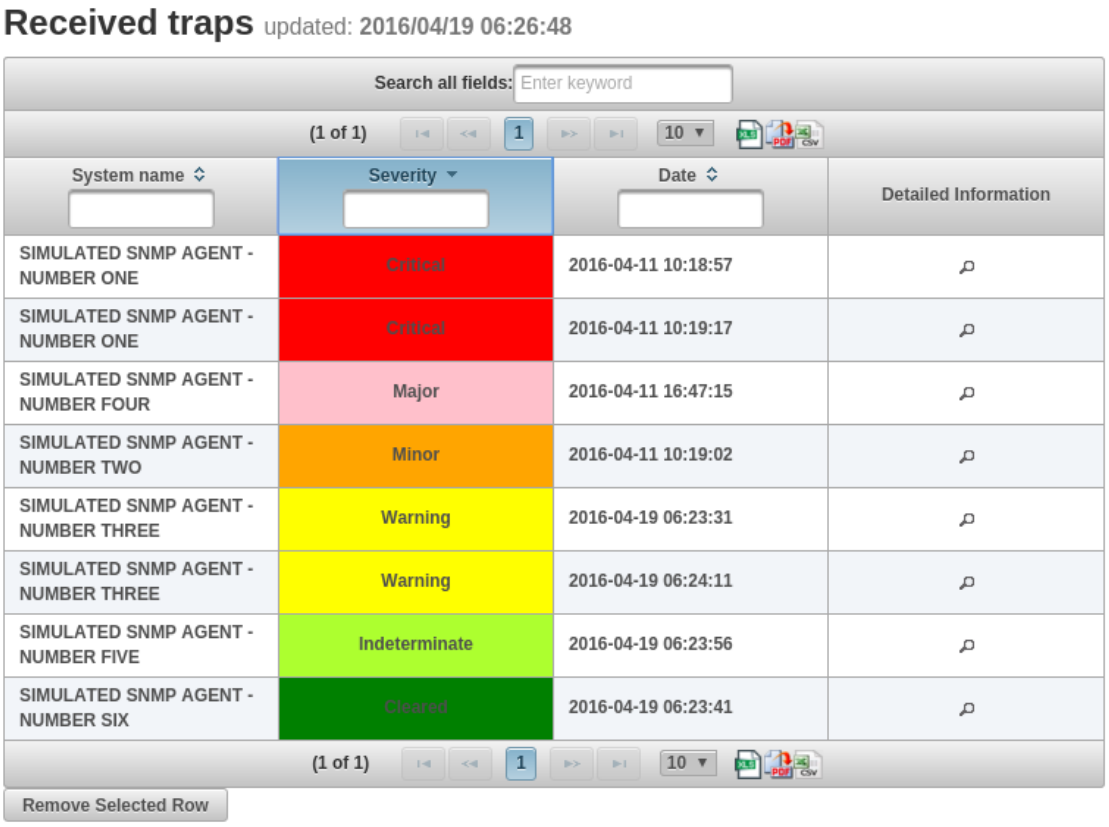

Figure 14: Screenshot that shows how network management information is presented

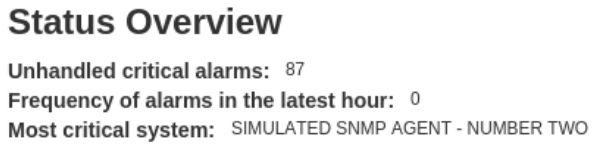

Figure 15: Screenshot that shows how the interface communicates the network status

1.4.2.1 Problems

Data-Table not Sufficient for Large Amount of Data Severity: 3

We consider the data-table to be a very effective way of presenting network management information to the user when the amount of data is of a reasonable volume. When the amount of data gets to large the data-table is not sufficient since it becomes to time consuming to look through the given table. If it can be expected that the data volume will be of a larger scale it should be considered splitting up the table in multiple tables or other presentation formats, that then could make it easier and more time efficient for users to sift through it.

1.4.3 Understanding classification of alarms without prior knowledge in the domain

The interface utilizes color coding to distinguish alarms of different severities, which is easy understandable even for users not acquainted with network management. This is achieved by utilizing three of Gestalt laws. The law of similarity (classification of alarms), the law of isomorphic correspondence (understanding the color coding) and the law of proximity (sorting of alarms).

The interface enables sorting on multiple parameters, including: date, system name, severity. Figure 14 demonstrates sorting on severities.

Alarm Source Visibility Severity: 3

While the color coding of the alarms provide an easy overview of the severities of the alarms it is harder to quickly gain that same understanding about the source distribution. While there is a text field for the source of an alarm there is no clear differentiator between different sources.

1.5 Results For GUI Version 2

1.5.1 Description

This GUI differs from the first in that it separates the alarms into multiple tables, one for each severity.

1.5.2 Sense-making of network management information

In this version of the interface the network management information is divided in multiple tables according to their severities. There is also a bar at the top of the interface that displays the most recent alarms with a critical severity, see figure 16.

Figure 16: Screenshot that shows how network management information is presented in GUI version 2

1.5.2.1 Problems

Hard To Get Full Network Overview Severity: 3

While the strength of this GUI approach lies in the ability to easily overview specific severities, it is not as easy to get a complete overview of the distribution of the severity of the latest alarms. Each table shows the newest alarms at the top, and also include the date and time of arrival, but it is hard to get a quick overview of how old alarms in one table are compared to those of another.

1.5.3 Understanding classification of alarms without prior knowledge in the domain

Severity: 2

This version utilizes Gestalt laws to communicate alarms that belong together and which alarms that should be distinguished from each other. In particular the law of proximity is used in the means of partitioning the alarms into different tables according to their severity.

1.5.3.1 Problems

Alarm Source Visibility Severity: 3

The distribution of alarms in terms of their source is not visible when first looking at the interface. To view the source the user is forced to inspecting each alarm.

1.6 Results For GUI Version 3

1.6.1 Description

This gui is characterized by its partition of alarms into multiple tables based on the source of the alarm, i.e one table for each source.

1.6.2 Sense-making of network management information

This version is similar to verion 2 in the layout of the GUI but differs in how it sorts and dispalys the information. By presenting the information with one table for each source it facilitates a rapid comprehension of how critical the status of each source is, see figure 17

Figure 17: Screenshot that shows how network management information is presented in GUI version 3

1.6.2.1 Problems

Hard To Get Full Network Overview Severity: 2

This version of the GUI have the same disadvantage as version 2. The GUI communicates to the user a good overview of the distribution of the alarms, i.e the source of the alarms, but it is not sufficient in presenting an overview in terms of severities and dates.

1.6.3 Understanding classification of alarms without prior knowledge in the domain

This version, just as the other versions utilizes Gestalt laws to communicate alarms that belong together and which alarms that should be distinguished from each other. In particular the law of proximity is used in the means of partitioning the alarms into different tables according to their source.

1.6.3.1 Problems

Alarm Severity Overview Severity: 3

The interface uses color coding to communicate that certain alarms belong together. However this color coding is done for multiple tables which has the drawback that if you want to view all alarms sorted by their severity it's not possible.

Appendix C

Project Methods

Project Methods

Kim Hammar
kimham@kth.se

Marcus Blom
marcblom@kth.se

May 9, 2016
Royal Institute of Technology, Stockholm

1 Overview

This documents outlines the project management methodologies for the study 'Integrating Monitoring Systems - Pre Study' and is meant to be used as complementary information to the methods described in the associated thesis. The methodologies described within the thesis concern how the pre-study's research questions were answered while this document instead focuses on the higher level aspects of performing the work of the study.

1.1 Background

The pre-study was part of a degree project, carried out at the Royal Institute of Technology (KTH). The project is the final examination in a three year program and upon completion awards a bachelor degree in computer engineering. The purpose of the project is to acknowledge the students ability to act as engineers in a real work environment.

1.2 Project Stakeholders

- *Students*. The students carrying out the project to fulfill the national degree objectives.

- *The Company, MIC Nordic*. The project was carried out on behalf of MIC Nordic and sought to advise them in the implementation of a new monitoring system.

- *The Academia, Royal Institute of Technology (KTH)*. An examiner at KTH assesses and reviews the results with respect to the national degree objectives.

1.3 Goals

The purpose of the project is to fulfill the expectations of all involved stakeholders and at the same time be a contribution to the industry. The goals can be divided into three categories.

1.3.1 Effect goals

The project should constitute as a pre-study in how a new monitoring system can be integrated in conjunction with MIC Nordics existing systems. The pre-study should result in conclusions based on research and implementation, that can be utilized as guidelines for how the final system should be built and implemented. Further more

the study should be a contribution to the industry and fulfill a gap among previous research in the field.

1.3.2 Result goals

The project has two primary result goals.

- Scientific thesis that presents the study and its conclusions.
- Prototype that MIC Nordic can use as a starting point for the development of a new monitoring system.

1.3.3 Project goals

Acknowledge the students ability to act as engineers in a real work environment. The project should also produce research and conclusions that lays a foundation for implementation of a new monitoring system at MIC Nordic.

1.4 Method

For this project we took an agile approach, that focuses on adaptability through regular feedback from stakeholders. This approach was preferred, as in the early stages of the project it was not possible to clearly define time estimates. Time estimates were not feasible considering that the students, although having experiences of similar engineering tasks, did not have any experience in the particular research area that the study concerns, apart from general knowledge. Making time estimations were also hard due to the students inexperience of working on a scientific research project of this scope.

The lack of experience for making accurate time estimations did not exclude exhaustive planning from the project methodologies, but meant that the planning needed to take these limitations into consideration. During the initial phase a thorough planning of the project was performed in collaboration with the stakeholders. This phase included identification of risks, proposal of research methodologies, definition of the projects purpose, goals, delimitations, work ethics and construction of a timetable for the project.

The agile project approach consisted of iterative work where overall goals for the project were decomposed into smaller tasks that built on each other. With this decomposition, the project was then divided in multiple phases, that facilitated incremental deliverements and presentations for stakeholders throughout the project. By following an agile approach, the stakeholders were involved and updated from start to finish of

the project. This resulted in continuous feedback to assure that the project was progressing towards the fulfillment of the expectations of each stakeholder.

This approach also allowed us to continuously reevaluate our progress as our knowledge in the field and the study progressed. Due to this we could ensure that we could delegate the work flow in such a way that we were able to deliver the project results on time.

1.5 Project Structure and Organization

The undertaking of this project mainly took place in KTH's facilities. The small group size, of only two students, removed the necessity of traditional project roles. It is very easy to delegate tasks on the fly between both members in such a group and the time overheads from using a stricter organizational model would not have benefited the project.

2 Project Phases

The project consisted of four subsequent phases, each containing several serial, or parallel, sub-phases.

- **Prestudy**
  - Defining problem statement
  - Project definition and planning
  - Identifying important aspects related to problem statement
- **Data collection**
  - Studying system documentation
  - Indentifying related sources
  - Literature studies
  - Interviews
- **Construction of a prototype**
  - System Design
  - GUI Design
  - Implementation
- **Analyze results and make conclusions**
  - Compile results
  - Evaluate results
  - Make conclusions

3 Risk Management

Risk	Precautions	Workarounds
The project work exhibits insufficient independence. According to the degree objectives, the student should perform independent work in the form of a degree project.	Continuous contact with supervisors from academia and the company in order to be able to identify if the risk is considered severe in an early stage.	Complement the work according to directives from the examiner in aspects of fulfilling the degree objectives.
Missed deadline for submission of the report.	Regular meetings with supervisor from academia to verify that the report is developed in a desired pace.	Discuss sanctions with the supervisor.
Missed deadline for presentation of results to the company.	Work iteratively with partial deliverables and presentations of results to the company.	Discuss sanctions with the company.
The work does not fulfill the company's expectation.	Maintain a project definition document and project proposal document that is approved by the company. These documents ensures that the expectations of the company and the students are on the same level. Have continuous meetings with the company to discuss the current results.	Negotiate with the company for initiating a process to complement the work.

Table 1: Identified risks

4 Deliverable

- Scientific thesis

  Delivered to the academia and the company. The thesis is assessed and reviewed by the examinator from academia.

- Prototype

  Delivered to MIC Nordic. Includes:

  - Source code
  - Architectural document
  - Test report

TRITA TRITA-ICT-EX-2016:26