Cape Peninsula
University of Technology

THE ENHANCEMENT OF NETWORK OPERATIONS CENTRES OPERATING MODELS THROUGH THE USE OF DATA ANALYTICS

by

DAVID MGCINA

Thesis submitted in fulfilment of the requirements for the degree

Master of Technology: Business Information Systems

in the Faculty of Business and Management Sciences

at the Cape Peninsula University of Technology

Supervisor: Dr Andre de le Harpe

Cape Town Campus
June 2020

DECLARATION

I, David Mgcina, declare that the contents of this thesis represent my own unaided work, and that the thesis has not previously been submitted for academic examination towards any qualification. Furthermore, it represents my own opinions and not necessarily those of the Cape Peninsula University of Technology.

26 June 2020

Signed (Student) **Date**

PROOFREADING CERTIFICATE

7 August 2020

DAVID MGCINA
Faculty of Business and Management Sciences
Cape Peninsula University of Technology
Cape Town

RE: CERTIFICATE - TECHNICAL EDITING AND PROOFREADING OF MASTER'S THESIS

I, the undersigned, herewith certify that the technical editing and proofreading of the MTech thesis of David Mgcina, entitled *"THE ENHANCEMENT OF NETWORK OPERATIONS CENTRES OPERATING MODELS THROUGH THE USE OF DATA ANALYTICS"*, has been conducted and concluded.

The finalised thesis was submitted to Mr Mgcina on 7 August 2020 and cc'd to Prof André de la Harpe.

Sincerely

Professor Annelie Jordaan
DTech: Information Technology
Ph: 065 990 3713

Member: SATI 1003347

South African Translators' Institute (SATI)

ACKNOWLEDGEMENTS

I wish to thank:

- My wife (Zimbini Mgcina), for moral support
- Dr AC de la Harpe, for great supervision – he demonstrated strong leadership and support
- Cape Peninsula University of Technology library services, for supporting me by making the required study material available
- Illze van den Bergh, for the support on advanced Microsoft Word usage
- Prof Annelie Jordaan, for the technical editing and proofreading
- The study participants for efforts made to contribute to the study

ABSTRACT

Telecommunications is an essential service as it enables people to communicate and supports business to function efficiently. This service demands that high availability is upheld continuously. Some network faults can be avoided by implementing preventative maintenance; however, certain faults cannot be avoided and must be handled efficiently when they occur. A Network Operations Centre (NOC) executes the role of detecting and analysing faults to identify the root cause. Network Operations Centre engineers perform analyses manually, which adds to the time duration of the fault. An increase in fault alarm volume is the underlying cause of long fault analyses.

This Design Science Research study aimed to examine data analytics solutions that enable network engineers to perform optimally during network outages. The three focus areas of the study were i) manual network analysis, ii) incident management analysis, and iii) the analysis of a smart grouping mechanism for alarm patterns. The collaborative rapid application development approach was adopted to introduce agility into the design, development and demonstration processes. Each of the three focus areas of the problem were handled as an independent delivery. Artefact development was based on data collected from users, alarm data, and incident data.

The main finding of the study is that the rule set based algorithms used in NOC automation are ineffective and cause long-term damage by hiding useful data needed for decision-making. A data analytics artefact developed in this study introduced 97% efficiency in alarm analysis and enabled unsupervised machine learning (ML) capability, allowing users to gain insight into the network with minimal effort. The proposed NOC advanced Information System is the operational model recommended for Mobile Network Operators.

Keywords: Design Science Research, Network Operation Centre, Telecommunications, Data Analytics, Clustering, KMeans

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

Acronyms	Definition
2G	Second Generation Cellular Network
3G	Third Generation Cellular Network
3GPP	3^{rd} Generation Partnership Project
4G	Fourth Generation Cellular Network
AI	Artificial Intelligence
DSR	Design Science Research
EMS	Element Management System
ETSI	European Telecommunications Standards Institute
FMS	Fault Management System
HLD	High Level Design
ICASA	Independent Communications Authority of South Africa
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IMS	Internet Protocol Multimedia Subsystem
IS	Information System
ITU-R	International Telecommunication Union Radio Communication Sector
LLD	Low Level Design
ML	Machine Learning
MNO	Mobile Network Operator
MOM	Manager of Managers
MSISDN	Mobile Station International Subscriber Directory Number
MTTR	Mean Time To Repair
MSP	Managed Services Providers
MSS	Mobile Switching Server
NE	Network Element
NFV	Network Function Virtualisation
NGN	Next Generation Network
NMC	Network Management Centre
NMS	Network Management System
NOC	Network Operations Centre
OMGW	Open Media Gateway
RAN	Radio Access Network
SLA	Service Level Agreement

Acronyms	Definition
SDN	Software Defined Networking
SME	Subject Matter Expert
SPOC	Single Point of Contact
SQL	Structured Query Language
TX	Transmission

DEFINITION OF TERMS

Term	Definition	Source
Alarm	Abnormal network entity condition that categorises an event as a fault	3GPP (2013)
Event	An exceptional condition occurring in the operation of hardware or software within the network managed; an instantaneous occurrence at a time	(Maździarz, 2018)
Failure	A loss of ability to operate to specification, or to deliver the required output or outcome	(Axelos.com, 2019)
Failure or Fault	Considered to be a kind of an error	(Maździarz, 2018)
Fault Management (FM)	The set of tasks required to detect cell faults and then identify and implement corrective actions to restore full operation	(Mulvey et al., 2019)
Fault Management (FM)	Detecting failures as soon as they occur and limiting their effects on the network Quality of Service (QoS) as far as possible	3GPP (2013)
Incident	An unplanned interruption to a service or reduction in the quality of a service	(Axelos.com, 2019)
Incident Management	The practice of minimising the negative impact of incidents by restoring normal service operation as quickly as possible	(Axelos.com, 2019)
Alarm Severity	specifying the importance of the fault and describing the alarm class	(Maździarz, 2018)
Monitoring	Repeated observation of a system, practice, process, service, or other entity to detect events and to ensure that the current status is known	(Axelos.com, 2019)
Event Correlation	The process of establishing relationships between network events	(Maździarz, 2018)
Machine Learning (ML)	Extracting relevant information, suggestions, and predictions from the datasets that are too large and too complex	(Kibria et al., 2018)
Artificial Intelligence (AI)	Replicating human intelligence or some aspects thereof and other cognitive functions	(Kibria et al., 2018)
Clustering	A descriptive task where an algorithm tries to identify a finite number of categories or clusters that can describe the data	(Kumara, 2018)
Machine Learning (ML)	A popular technique for extracting knowledge from data	(Ayoubi et al., 2018)
Development Environment	An environment used to create or modify IT services or applications	(Axelos.com, 2019)

CHAPTER ONE: INTRODUCTION

The chapter flow is as follows:

- Introduction
- Problem
- Background to the research problem
- Statement of research problem
- Objectives of the research
- Research questions
- Hypothesis
- Research design and methodology
- Limitations of the research
- Significance of the research
- Ethical considerations
- Outline of the research

1.1 Introduction

A Network Operations Centre (NOC) is an essential part of a telecommunications organisation that primarily focuses on detecting and analysing network anomalies. This is important in the context of the study because network irregularities that take a long time to be resolved prevent customers from using services such as making mobile calls and connecting to the Internet. This study concentrated on activities carried out by Network Operations Centre engineers during the detection and analysis stages of managing a fault.

In this chapter, the researcher introduces the problem statement as well as the study aims and objectives. To ensure clarity, the context of the study in relation to academic content from prior studies and industry practices is covered. The researcher further details the problem addressed in the study and questions answered by the study.

1.1.1 Network Operations Centre

1.1.1.1 NOC description

A Network Operations Centre, commonly referred to as a Network Management Centre (NMC) or fault management centre, is a centralised area where remote monitoring and management of the network is done. NOC teams are key to the managed engineering and Information Technology (IT) services of, and a big driver of service delivery for, many Managed Services Providers (MSPs).

1.1.1.2 NOC function

NOCs are responsible for monitoring the health of network infrastructure. Teams detect network anomalies, degradation and outages from alarms and make decisions and adjustments to effect repairs. The environment is active for 24 hours a day, 7 days a week and 365 days a year to assure network uptime. All the technical teams comprising of planning, maintenance and field technicians, interact through the NOC.

1.1.2 Network Management Systems

A mobile telecommunications network has a combination of technologies (radio access network, transport network and core network) as per The 3rd Generation Partnership Project (3GPP) standards. All of these technologies have their own infrastructure and trigger unique alarms in the case of network failure. Each of the technologies has its own management system that collects the alarms. The NOC then centralises all the data received from the management systems – this is referred to as the Manager of Managers (MOM). The NOC uses the MOM to pick up network faults and correlate events in order to work towards fault resolution. Each network fault is recorded in an Incident Management System (IMS) for tracking and reporting purposes.

1.1.3 Managing network faults

NOC engineers manually go through large amounts of alarm data to investigate faults from the time alarms are triggered. This process takes long and the investigation quality and speed depend on the skill level of the engineer on duty. The engineer's analysis determines the speed of recovery from the fault (Casey, 2018). The challenge is that business depends fully on this manual process, which usually takes a long time to complete.

1.1.4 Customer impact

Network outages result in customers experiencing degraded or no service on their mobile devices. These customers include individuals and enterprise groups. Communication is a key service, and service unavailability dissatisfies customers; in cases where unavailability is frequent, customers may stop using the services of the mobile operator. This has a direct impact on the quality of service (QoS).

1.1.5 Impact of network failures

Businesses are unable to generate revenue during network outages, which then leads to low or even no profits. Low to no profits lead to cost reductions, which may result in retrenchments. South Africans have limited choices in respect of Mobile Network Operators (MNOs), and network failures on the part of any one of the operators have a significant impact on communications in the country. This problem directly affects Goal 8 of the 2016 United Nations Sustainable Development Goals, namely, *decent work and economic growth* (United Nations, 2016).

1.2 Problem

Extended network faults are caused by NOC engineers working manually to perform fault investigation. The whole process of network fault investigation is reactive, as these investigations are conducted manually.

1.3 Background to the research problem

All network alarms are sent from network elements to the network management systems used by the NOC to monitor the network (Priyadharshini, 2018). Each anomaly or deviation from normal processes is translated into alarms from the network elements. These alarms have different impacts, classified by severity, and which indicate how the perceived capability of the managed object has been affected. The colour of the alert in the event list is controlled by the severity of the event (Das et al., 2018).

The NOC receives a large volume of fault alarms (Niyazmand & Izadi, 2019), which requires the engineers to analyse these faults and take action based on results of this analysis. This process has been followed for the past few years, during which many studies were done (Kim et al., 2011). Analysing data on such a large-scale is challenging to the NOC engineers and NOC systems. As Gebre-Amlak (2018) discovered the duration of a network outage depends on an analysis being done quickly and action being taken almost immediately. The time taken to localise the fault through analysing the cause of the alarm(s) leads to extending the time taken to resolve or repair the fault (Wallin & Landén, 2008).

A number of studies have been undertaken in an attempt to overcome this problem. The solution to this problem was pre-defined alarm correlation (Wang et al., 2017a; Das et al., 2018; Salah et al., 2019). Subject matter experts (SMEs), who are highly skilled NOC engineers, studied alarm history records and identified the sequence of alarms for each fault scenario. The findings were applied to network monitoring systems to ensure correlation is happening place without human intervention for each possible scenario (Maździarz, 2018; Salah et al., 2019). The solution works well but is limited since it is not dynamic. Rapid changes in the networking environment bring changes to alarm data and format, which makes the predefined alarm correlation rules obsolete.

1.4 Statement of research problem

The complexity of Network Management Systems (NMS) has increased over the years and the volume of alarms processed by NOCs has increased (Das et al., 2018); this is also supported by Maździarz (2018). The impact of this challenge is that network outages are being prolonged since network investigations are based on analysing alarms to discover where the fault lies. Extended network outages have a negative impact on MNO subscribers (Salah et al., 2019) due to loss of revenue and brand damage. Research was undertaken in an attempt to close this gap, and predefined alarm correlation was found to be the solution to the problem (Maździarz, 2018). However, this solution has limited value since it can only address the scenarios related to known faults. The gap in research is the lack of data analytics application to simplify network analysis. The existence of this gap is argued by Kibria et al. (2018), who posit that the gap is closed in the Artificial Intelligence (AI) of next-generation networks (NGNs). This argument is, however, not valid because MNOs have hybrid technology implementations (2G, 3G, 4G, and IMS); thus, all the technologies need standard management practised in a uniform manner to ease the investigating complexity for the NOC engineers.

1.5 Research questions

RQ1: What is the relationship of alarm volumes to network monitoring functions with reference to time taken to action?

RQ2: How should a data analytics application for problem detection be implemented so that minimal effort would be required on NOC engineers?

1.6 Hypothesis

Data analytics solutions increase productivity in the NOC by automating the analysis of alarms.

1.7 Aim and Objectives

1.7.1 Aim

The aim of this study was to examine a data analytics solution to enable network engineers to perform optimally during network outages.

1.7.2 Objectives

The objectives of this study were:

i) To propose a data analytics solution that reduces manual network analysis
ii) To propose a near real-time management solution for incident management
iii) To propose an automated smart grouping artefact for alarm patterns

1.8 Research design and methodology

A qualitative pragmatic research approach was used to explore the use of data analytics in network operations. This researcher based the study on the viewpoint that i) a wide range of methods is available for deciphering the world and for undertaking research, ii) that no single perspective can ever give the whole picture, and iii) that there might be numerous real world factors. The ontological position adopted for this research is constructivism, with interpretivism as the epistemological stance. The Design Science Research (DSR) methodology was applied. DSR has proven to be the most effective method for supporting information systems (IS) studies that result in the building of an artefact. Data were collected from users in the form of technical requirements and system information was collected from data extraction systems. The artefact was developed using an open source programming language.

1.9 Limitations of the research

The research focused on the Core Network of one MNO in South Africa. Radio and transport networks were excluded.

1.10 Significance of the research

The research uncovered a clear and systematic way of deploying data analytics in the NOC environment. Standardisation bodies (IEEE, 3GPP, ETSI, ITU-R) can use the output of this study to advance network management standards. The MNO is empowered to redesign NOC departments to include structures that combine the services of network engineers and data scientists.

1.11 Outline of the research

The search is divided into seven chapters that cover the following:

Chapter One: Introduction – This chapter focuses on describing the origin of the research problem, the dynamics around the problem and the reasoning why the topic is research worthy.

Chapter Two: Literature Review – This chapter focuses on studying literature related to the research problem and identifying research gaps.

Chapter Three: Research Methodology – This chapter lays out the approach that was taken in the study.

Chapter Four: Artefact Construction – Problem identification, requirement planning as well as artefact design and development are all covered in this chapter, including a demonstration of the prototype.

Chapter Five: Artefact Evaluation – Assessment of artefact against the identified problem was carried out and is discussed in this chapter.

Chapter Six: Discussion – Interpretation of evaluation results was undertaken and is described in this chapter.

Chapter Seven: Conclusion – This chapter is a summary of the study from the problem, through the research process to the recommendations and proposed final solution. The Chapter ties the study aim, objectives and questions to the findings, thus, confirmation that all that is outlined in Chapter One has been achieved.

1.12 Summary

The information systems (IS) domain is fundamental to all sectors where there is data to be managed and, in this case, the study was undertaken on the NOC component of a mobile telecommunications organisation in South Africa. In this chapter, the researcher outlined the problem, background and objectives of the study. The six remaining chapters build upon this chapter. The findings and recommendations of the study are based on the content of this chapter. Proof of the university's approval for the study to take place is provided in Appendix A.

CHAPTER TWO: LITERATURE REVIEW

The chapter flow is as follows:

- Introduction
- Impact of telecommunications on customers
- Common Caused of Faults
- Network Operations Centre
- Alarm Principles and Technology
- Architecture fault management system
- Software-defined networks and network functions virtualisation
- Gap in NOC domain not addressed
- Data analytics
- Machine learning
- Artificial Intelligence

2.1 Introduction

The theoretical background to support this study as well as solutions applied in prior studies are covered in this chapter. The researcher gathered knowledge from studies related to this research problem and critically evaluated the contents. To ensure the relevance of the identified gap, the researcher predominantly focused on peer-reviewed journals as a means of assuring the quality of the studied content. The foundation concepts of NOCs are covered in this chapter, with particular reference to the South African mobile telecommunications context.

2.1.1 Mobile phones in Africa

The effect of cell phones on economic development in Africa far surpasses their significance on other continents (Teagarden, 2019). The appearance of this innovation opened doors for private ventures in regions where individuals never had landlines; which even brought forth whole enterprises to the continent. A larger number of individuals on this continent have cell phones than they have books, so it is no surprise that Africa has spearheaded the utilisation of cell phones for banking. Mobile network providers such as Safaricom in Kenya and Vodacom have implemented solutions that allow users to make financial transactions through mobile phones without user having back account – the solution is called M-Pesa (Ntara, 2015). Communication technology is a successful business in Africa.

2.1.2 Mobile telecommunications in South Africa

The telecommunications segment is booming, despite a depressed world economy. Firms are investing intensively in infrastructure to satisfy future demand. Financial reports of telecommunications companies undeniably paint a progressively complex picture, one defined by fast innovative advances and unsure income sources. MTN reported revenue growth of 9.7% in the 2019 year (MTN, 2019). Vodacom reported a net profit of R15,562 million in the 2018 year and R15,532 million in 2019 (Vodacom, 2019).

Every generation of wireless communication is an improvement on existing innovations (Lauridsen et al., 2017). The original wireless communication generation was cautiously simple, made for voice only, and has now become redundant. Second generation wireless communication (2G) was an advance on the first, yet remained for the most part suitable for voice and basic data transmission. In 2000, third generation wireless communication (3G) offered higher transmission speeds for mixed media use, and mobile broadband exclusively for cell phones. After 2010, the fourth generation wireless communication (4G) increased the data transmission limit.

South Africa has a highly competitive mobile telecommunications sector (Omonona, 2019). There are multiple operators in this space, presenting a variety of options to customers. Two operators, Vodacom and MTN, have dominated this sector and currently, they still do (ICASA, 2019).

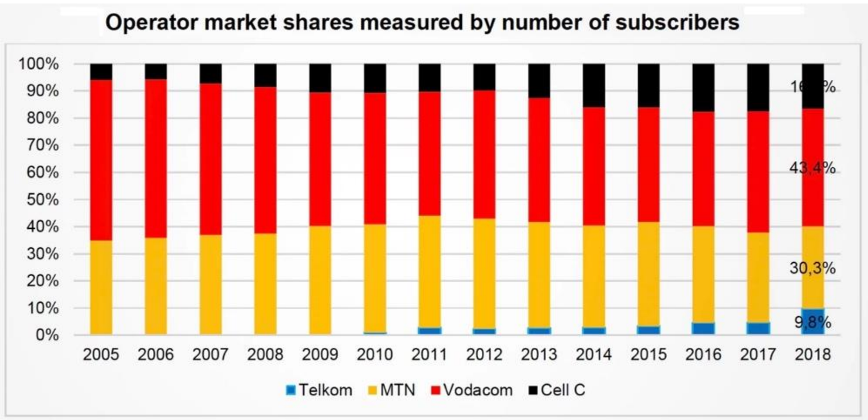


Figure 2.1 South Africa's Mobile Network Operator market share

Vodacom has a market share of 43.4% and MTN has a market share of 30.3%, while the remaining operators share 29% of the market between them (Figure 2.1). MTN and Vodacom enjoy the advantage of being first to market. The country's population is spread across urban and rural areas (Figure 2.2). Urban areas have good infrastructure, a high concentration of people and high levels of employment. Johannesburg, Cape Town and Durban are the biggest cities with the highest number of active subscribers to mobile telecommunication. The rural areas constitute a greater geographical area; however, employment levels are low.

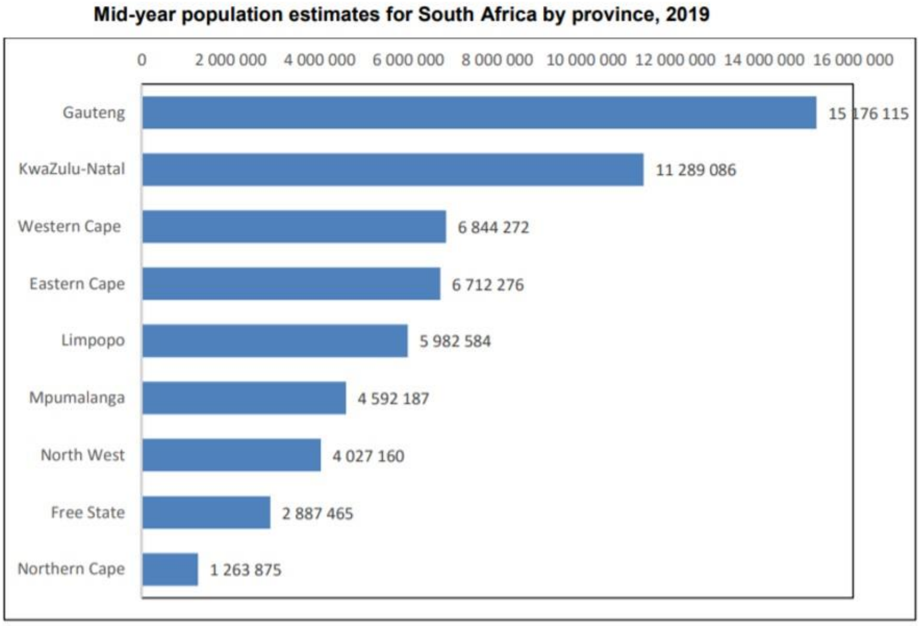


Figure 2.2: South Africa's population by province

Customers have the option of moving from one MNO (Lin et al., 2018) to the other with the principle called Number Porting. Number Porting has been a great feature in driving competition as customers could move to any operators without their Mobile Station International Subscriber Directory Number (MSISDN) changing, meaning people will continue call that customer with the same number. It is common practise to see people changing mobile operators.

Government plays a key role in mobile telecommunications as they allocate the spectrum that the MNOs use to connect customers (ITU, 2014). There are fees and licences attached to the use of the different frequencies of this spectrum that MNOs are subjected to. Customers are protected by government in order to prevent MNOs from exploiting them, be it by pricing or incorrect use of their personal data. The Independent Communications Authority of South Africa (ICASA) plays a role in the regulation of the industry.

2.1.3 Mobile telecommunications technology

Three main domains in mobile networks serve as the foundation of all mobile services. Each of the three domains is a unique speciality, which is best practice globally:

i) Radio Access Network (RAN) is the part of a mobile telecommunications network that carries out data transmission over air interface
ii) Transport (commonly referred to as TX) is the part of a network that connects all network devices through the medium of Ethernet, microwave or fibre
iii) Core is the central part of the network that hosts the subscriber data and executes transaction decisions

Complex engineering principles are applied to enable the sending and receiving of information (Mihailovic, 2018) between devices, such as mobile phones or computers. Information is transported through the air interface, and standardised protocols ensure that data are encrypted. Mobile cellular networks play a huge role in the data analytics space (He et al., 2016) as generators and carriers of massive data. Tasks that were previously performed by skilled engineers can be completed with the use of data analytics and machine learning (ML) (Mulvey et al., 2019).

The telecommunications industry experiences frequent changes, with new hardware and new technologies being introduced, including Software-Defined Networks (SDNs) and Network Functions Virtualisation (NFV) (Cherrared et al., 2019). This view is also held by Nawal and Cherif (2019). Both SDN and NFV aim to simplify the management of the network. Fault management is a critical part of the network maintenance because of the nature of the management of the infrastructure. Network failure can cause significant losses and impact on the business image and credibility of the operator. Cherrared et al. (2019) shared an example of failures within IBM that resulted in a revenue loss of 31 million AUD.

2.2 Impact of telecommunications on customers

2.2.1 People (customers and citizens)

The ability to communicate without limitation, caused by people continuously moving between places, is key in enabling families, friends and communities to communicate with each other. South Africa is unique in terms of many employment opportunities being concentrated in the main cities, which results in people moving from rural areas of the country to the cities (Omonona, 2019). In some cases, people move from one city to the other. Mobile telecommunications is the solution for people to communicate.

Service is the primary attribute that individuals notice about any organisation or brand. The capacity to convey reliable service quicker and, preferably, less expensively than rivals, can separate one business from another (Brady, 2000). A satisfied customer is somebody who has had a fulfilling experience. A loyal customer is somebody who has a relationship with your organisation, which is valued and prevents them from taking their business to rivals.

2.2.2 Economy of the country

Communication in every business, regardless of the sector, is very important (Kumara, 2018). The ability to communicate with the right person as soon as the need arises, leads to business processes flowing efficiently. Telecommunications unlocked the capability of business-to-business as well inter-business communication.

Research by Ozcan (2018) revealed that globalisation has had a big impact on the world, meaning that businesses operating in South Africa, in many cases, are required to communicate with business outside of the country. A lack of good communication technology limits the capacity of businesses, which directly links to economic activity in the country.

2.2.3 Mobile Network Operators

A company providing mobile telecommunications solutions is referred to as a Mobile Network Operator (MNO) as per industry standard (Ali & Alshibani, 2018). MNOs operate in a highly standardised industry (Otani et al., 2017) as solutions provided by one MNO need to be fully aligned with services provided by another MNO for the service to be effectively rendered, i.e. a customer from MNO A must be able to call a customer from MNO B. The two MNOs are required to see that the call goes through, and perceive the call as a revenue opportunity for both. The relationship between MNOs is visible on a global level as well, where customer roaming is at play. MNOs depend greatly on achieving high data rates (Mulvey et al., 2019) because of limited radio spectrum. Rule-based automations have proven to be effective in the past, but with the rate of technology change in the network, smarter and more dynamic rules are needed to meet business requirement (Wang et al., 2016). MNOs in South Africa face unique challenges that affect the services they provide. These challenges are detailed in sections 2.3 and 2.4 below.

2.3 Unstable power grid

There is one electricity producer in South Africa called Eskom. Over the past number of years this company has been experiencing difficulty in maintaining stable electricity supply (Styan, 2019). The generation capacity is limited.

MNOs use Eskom’s electricity supply, and when electricity unavailable, they have to use generators and batteries. Generators require diesel, which is an additional cost to the operators. At big sites there are static generators, while remote sites use mobile generators that are deployed on demand. Electricity supply had a big impact on the area being studied in this research since network elements depend on electrical supply to remain operational. An influx of alarms is received by an NOC when backup power is depleted.

2.4 Elements of crime

Each MNO site has batteries to provide backup electrical power. Batteries are often stolen and in the process of stealing batteries, thieves vandalise the equipment at the MNO site. This has led to an increase in security costs to the MNOs. Vandalised sites often result in down time and the NOC has to manage the incident to restore the site to its operational state.

2.5 Network Operations Centre

Fault management (FM) is the process of locating, analysing, repairing and reporting network problems such as link failures and network overload, as defined by Cherrared et al. (2019). There are multiple teams in the MNO with specific focus areas. Predominant teams are the technology group, the planning team, field operations team and the NOC. These teams work closely together to deliver the service. Teams are spread across the country. The mobile operator that was the focus of this study had field operations teams and planning teams employed South Africa (Reali et al., 2018).

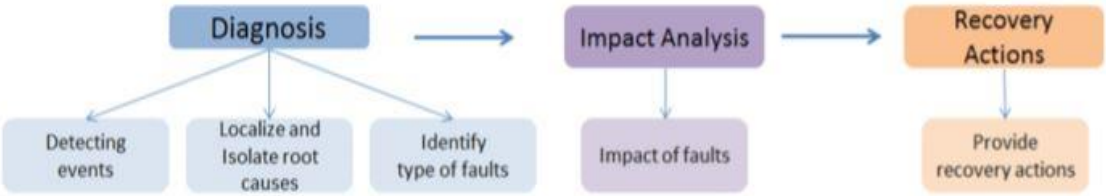


Figure 2.3: Fault management process (adopted from Cherrared et al., 2019:1540)

The NOC of the organisation that was studied is a centralised team operating from Cape Town. The NOC's main role is to ensure that any disruption in the network is detected and resolved as soon as possible. The NOC interacts with various teams and tracks the actions taken towards the resolution of the fault. The NOC is responsible for the entire process from fault detection to fault resolution (Figure 2.3). Major outages lead to the unavailability of services to customers and the process of identifying the root cause of the outage is complicated. Many factors can lead to a network fault, factors ranging from hardware malfunctions, software defects, design mistakes and misconfigured parameters to incorrect actions by the network engineers.

Fault management techniques can be classified into two categories (Mulvey et al., 2019), namely:

a) Logic-based and algorithmic approaches use a set of rules to encode knowledge explicitly about the relationships between fault symptoms and causes.
b) Algorithmic techniques, on the other hand, incorporate expert knowledge implicitly within the software implementing the algorithm.

There are multiple industry standard key performance indicators, which are used to measure the efficiency of recovering from network outages (Gebre-Amlak, 2018). In the case of an outage that affects a wide area (e.g. cable break or power outage), alerts are sent to the NOC team, normally in high volumes. NOC engineers consume the alert data (Goel et al., 2019) as part of the investigation in order to make a decision on what needs to be done. If the fault can be resolved remotely, the NOC will proceed with the resolution procedures. Field technicians and equipment supplied are involved in cases where the NOC is unable to resolve (Gebre-Amlak, 2018).

2.6 Alarm flood

A flood of active alarms has been identified as a detection risk (Meng et al., 2015), especially a flood of false or non-critical alarms. NOC engineers spend a great deal of time on alarms that eventually prove not to need attention the system is thus inefficient. Meng et al. (2015) developed a method of knowledge-based alert verification (KAV) to reduce the number of unwanted alarms. The KAV model is driven by input from the subject-matter expert into the algorithm to provide the desired filtering (Figure 2.4).

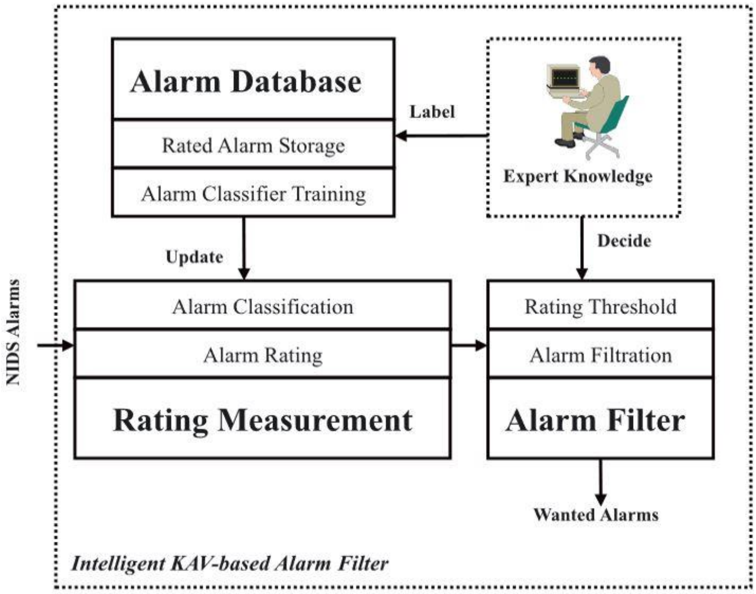


Figure 2.4: KAV-based alarm filter (adopted from Meng et al., 2015:3885)

One major network fault may trigger many alarms because of the relationship of one network element to many other elements. Unplanned shutdowns trigger a flood of alarms (Sanchez-Pi et al., 2015), frequently over 1,000 alarms per minute. This information is beyond what a human being can process. Serious accidents in the last decade, such as the Mildford Haven refinery accident in the United Kingdom (UK) on 24 July 1994, which resulted in a loss of £48 million and two months of non-operation, were related to poor fault detection caused by alarm floods. Operations centres implement performance measures such as MTTR to evaluate the response of alarms. Advances in IS in the industry (Tabia & Leray, 2011) present new challenges, attributed to the interworking of technologies and the increasing complexity of systems. False alarms are one aspect that makes the management of alarms very difficult, as NOC staff first need to identify a valid condition in order to initiate an investigation.

2.7 Technology and processes

Data is harvested from all network elements that form part of the MNO's infrastructure, and the NOC consumes the data. The structure of data from network elements is regulated by the regulation body (IEEE NMC). Alerts that indicate malfunction or degradation of infrastructure are collected across the network (Su et al., 2018). The alerts are made up of alarms, warnings and error logs. The alert types are determined by the equipment supplier and they differ from supplier to supplier. The South African telecommunications sector is dominated by Huawei, Nokia and Ericsson (Omonona, 2019). The mobile operator studied in this research has Huawei and Nokia in their RAN and Core domains.

Goals, objectives and critical success factors of the NOC define why incident management (Salah et al., 2019) is important to information technology's overall vision for delivering and supporting effective and efficient services. The NOC establishes the fundamental goals, objectives and critical success factors underpinning the incident management process. There is a direct relationship between incident and problem management. Incident management focuses on any immediate action taken to restore services, and root cause analysis of the problem is undertaken since this analysis will normally involve the determination of more extensive action to avoid or mitigate risk of similar faults in the future (Figure 2.5). The goals for the incident management processes (Haryadi, 2018) are to:

- Restore normal service operations as quickly as possible
- Minimise the adverse impact of the fault on business operations
- Ensure that agreed levels of service quality are maintained

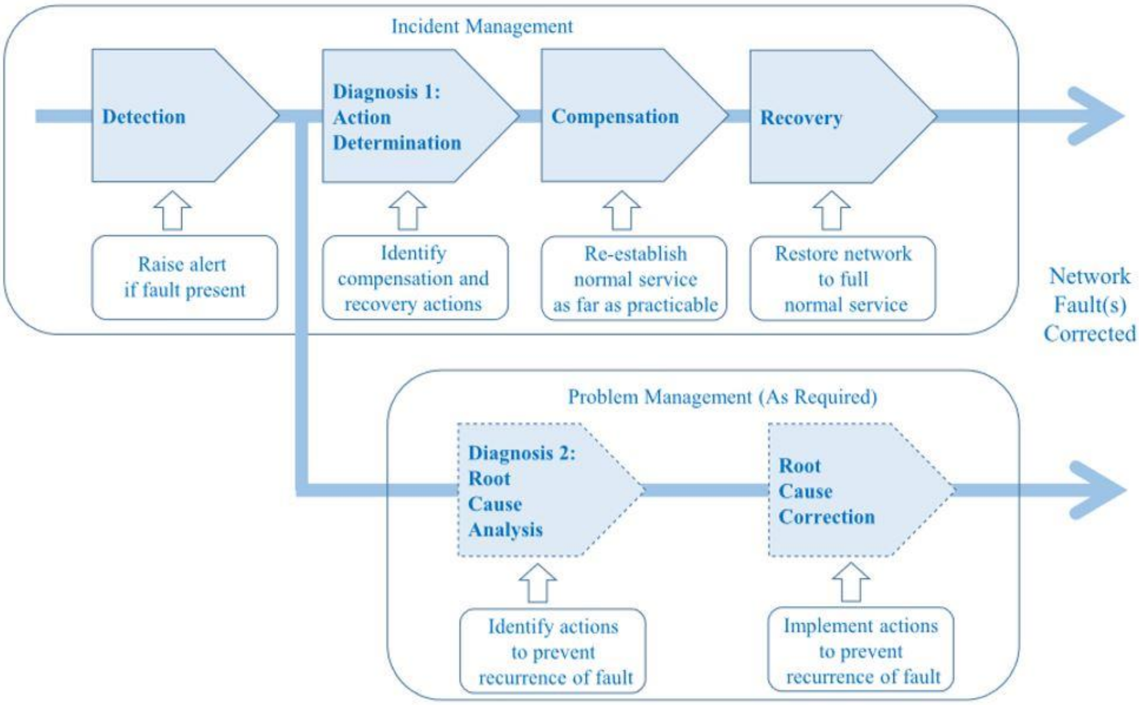


Figure 2.5: Incident and problem management processes (adopted from Mulvey et al., 2019:1245)

To achieve this, the objectives of the operational team's incident management (Das et al., 2018) process are to:

i) Ensure that standardised methods and procedures are used for efficient and prompt response, analysis, documentation, ongoing management and reporting of incidents.
ii) Increase visibility and communication of incidents to business and technical staff.
iii) Enhance business perception of engineering through the use of a professional approach to quickly resolve and communicate incidents when they occur.
iv) Align incident management activities and priorities to activities and priorities of the business.
v) Maintain user satisfaction.

Faults on elements within the MNO network are detected by the NOC teams by means of monitoring systems or are reported to the NOC by customer partners within the MNO – i.e. customer care, regional operations, and enterprise business (Zeydan et al., 2016). Trouble tickets are logged on trouble ticketing systems (Wallin & Landén, 2008). Some faults are investigated and attempts are made to resolve them by means of remote re-configuration, while other faults are assigned to the relevant parties, depending on the nature of the ticket. If a fault cannot be resolved remotely, the NOC assigns priority to the fault and the fault is handed to the responsible support department for action (Wang et al., 2017a). The handover takes place by means of a telephone call, an email or a task assignment on the IMS. Depending on the priority of the fault, the escalation specialist will escalate the fault to the relevant parties or send a notification Short Messaging Service (SMS) to inform the relevant business area. Depending on the nature of the failure (Keim et al., 2006), an emergency conference may be initiated involving all parties that need to contribute towards a resolution. Engineers and management will participate in the conference to ensure that decisions are made as quickly as possible. The support departments may indicate that the fault needs to be assigned to the vendor or supplier, in some cases.

2.8 Architecture fault management system

IBM Netcool is one of the leading NOC solutions for network monitoring and is used by the organisation studied. The implementation of this solution is divided into silos of object servers in a three-tiered architecture. This architecture helps to load balance the overall display as well as alarm processing functionality. The Element Management System (EMS) layer acts as a source of alarms managed by Netcool. Typically, EMS manages the functions and capabilities within each network element (NE) (Tang et al., 2019). The probe layer contains generic probes that continuously fetch raw alerts data from the NE. The collection layer contains the Netcool Omnibus Object Server, which stores all the alerts generated. In this database, alerts are processed automatically by a configurable process called automations. This layer interacts with the Netcool/Impact application for any event enrichment and SMS/email notifications.

The aggregation layer performs aggregation of the data received from the collection layer (Osahenvemwen & Omorogiuwa, 2011). The display layer is presenting network events (alarms) as handled by the collection and aggregation layers to the graphical user interface. In addition, load balancing functionality across the display object servers is provided to ensure controlled management of user sessions for best experience. The Graphical User Interface (GUI) layer contains WebGUI servers. WebGUI is the topmost layer and displays alert information to the user. This enables the user to create a Trouble Ticket (TT) and to update the status of the alert. Impact interacts with external systems (external DB) to append information about the alert or alarm generated.

The network is large, with over 12,000 hardware components which, under normal circumstances, have many alerts being generated. NOC engineers integrate multiple alarm feeds into one central tool, referred to as the umbrella fault management system commonly known as the MOM. This centralised tool is built to manage high volumes of alert data. The higher the volumes of alerts being presented to the NOC engineers, the more complicated it gets for them to correlate the causes of the different faults. The advantage of the MOM is that there is one place for everyone to consume the data.

2.9 Network disruptions

The unavailability of network elements results in customers being unable to access services. MNOs face the challenge of poor net promoter score, unhappy customers, loss of revenue and pressure from the regulatory body, ICASA. Network failures affect all parties in the value chain. The longer the outage, the more severe the impact is on the mobile operator. Customers are unable to access basic emergency services such as hospitals, police stations and fire departments, to mention a few. The quality of life of the operator's customers is heavily affected by disruptions, which is not ideal. There are various causes of network outage or degradations, including: electrical supply, hardware failures, cable breaks, vandalism, traffic patterns (congestions), licensing and human error, among others.

2.9.1 Employees

NOC engineers are expected to work through high volumes of alarm data to ensure that the network is in a healthy state. A study performed by Pavard and Dugdale (2006) confirmed that NOC engineers performed tasks that required cognitive events related to memory, evocation of representations, problem resolution, decision-making and vigilance. A high level of pressure was experienced by employees, as well as fear from committing mistakes in the decision-making process. Over the years, the number of alerts sent to the NOCs has increased, making life very difficult for the engineers. The variety of alerts is vast and changes frequently; the learning rate of the engineers cannot match the rate of change.

The NOC employees operate under high pressure as they deal with challenges that directly impact on the organisation's revenue. The senior leadership team expects that faults are detected as soon as they occur, and that they get resolved immediately. This expectation is channelled to the NOC staff. The process of monitoring incoming alarms and performing assessments to determine which of the alerts need to be actioned on, consumes much of the engineers' time. In major faults, there is a storm of alarms, which is impossible for a human being to work through. Customer care departments receive high volumes of complaints during major failures and hand them over to the NOC for resolution. Enterprise customers contact the NOC through business-to-business processes when they are affected. The NOC engineer is the single point of contact (SPOC) during outages. Each party contacting the NOC is working according to agreed service level agreements (SLAs) and, depending on the complexity in processing alerts, the details of the NOC engineers who failed to respond and resolve alarms timeously are escalated to MNO management.

2.9.2 Impact of Operations outputs on Mobile Network Operators

A fully functional NOC enables the organisation to avoid outages where possible and minimise outage time in cases where outages could not be avoided. The MNO manages to reach high full term for a Net Promoter Score (NPS), maintain revenue streams, and meet government operating license requirements if the NOC is efficient, while the opposite is true if the NOC is inefficient.

2.9.3 Modern solutions addressing challenges

Alert automation solutions, including delay and correlation (Salah et al., 2019), are commonly implemented by an MNO as a way of managing the volumes of alerts dealt with by the NOC employees. These solutions work very well; however, they are dependent on the SME knowing the pattern of alerts in advance. The algorithms are built on these known facts. To improve on time spent on manual efforts to create incident records, this can be solved by the automatic incident logging solution. Furthermore, there is a workflow solution for the automatic assignment of alarms that do not need the NOC's attention, for example power failures at a site. These solutions are rule-based and do not have intelligence to learn changes in the network. Highly skilled engineers who are SMEs in the domain spend a great deal of time assessing alert logs to learn how the defined rules perform and to identify new cases that were not considered initially.

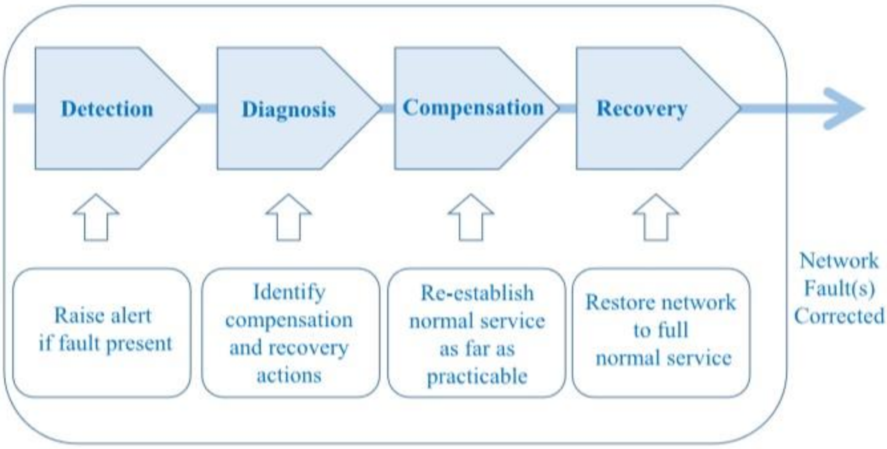


Figure 2.6: Self-healing solution (adopted from Mulvey et al., 2019:124516)

Equipment suppliers in the telecoms industry have been working on a solution that detects and automatically corrects faults. The objective of this initiative is to reduce the effort expected of network engineers to a minimum. Self-healing represents the autonomic ability for a system to recover from failures (Cherrared et al., 2019). A controlled and well-sequenced process is essential for effective self-healing, and the function of each sub-process is well defined in the use case, ranging from the triggered alarm to the restoration (Figure 2.6).

2.10 Rule set based algorithms

An intelligent alarm correlation system was developed by Bandini et al. (2006), taking into consideration the dynamic nature of network fault conditions. The system was based on the use of domain knowledge. Data mining is valuable in the discovery of new insights and in enhancing our understanding of existing knowledge (Bandini et al., 2006). According to Tabia and Leray (2011), alert correlation is the task of analysing the alerts triggered by one or multiple network elements in order to provide a synthetic and high-level view of the interesting malicious events targeting the information system. Most fault detection solutions have features that allow users to configure them to their own requirements (Tabia & Leray, 2011).

Alarm delay is another mechanism applied to minimise the number of alarms displayed on the views of the NOC users. The process holds alarms back for a specified duration before presenting them to the users' network monitoring view (Zeng et al., 2017). Once the set delay time has expired and alarm is still active, then alarm will be presented to the NOC users, a rule favoured by previous studies on alarm reduction. Threshold is another concept applied to manage frequent occurrences of the same alarm, and such alarms are active for short durations. Threshold assess whether the number of alarm occurrences meets a set count within a specified time and, if so, one synthetic alert is created, indicating to the user that there is a problem (Salah et al., 2019). The synthetic alarm masks the actual alarms. Lastly, there is a suppression rule that completely removes predefined alarms from the view of the user. Most importantly, one clearly sees a pattern of key information being removed from the NOC user in order to reduce alarms presented to the user on the screen for short-term gain. Alarms hidden from users are referred as nuisance alarms (Wang et al., 2017a). The practice of hiding data deprives the organisation of a comprehensive view of the network status, which is the fundamental principle of this study. The alarm colouring method is common practice in the industry and used by leading solutions providers such IBM Netcool to aid NOC users with the detection process (Das et al., 2018). Considering the alarm volumes, the colour of alarms is pointless, as the monitoring application will present too many events, thus pushing colours considered high impact to the bottom of the list where a user can no longer see it. The literature supports the alarm flooding condition detailed in section 2.6; however, the sustainability of rule set based algorithms is not practical, as will be further discussed in section 6.3.1.

2.11 Software-defined networks and network functions virtualisation

Software-Defined Networking (SDN) is defined by Yu et al. (2019) as an emerging network paradigm that promises to simplify network management and enable innovations in networking. Implementing open flow for automating network management activities poses the risk of conflicting rules being active since multiple vendor systems will be managed on the same layer. Network engineers need reskilling to be able to integrate the SDN architecture. Cherrared et al. (2019) define SDN as an emerging network architecture that separates the network control, management and forwarding functions, thereby enabling the network control to be directly programmable and the underlying infrastructure to be abstracted for applications and network services. Network troubleshooting is a difficult exercise to undertake, especially when there are multi-tier architecture and complex network states in SDN.

Cherrared et al. (2019) define Network Functions Virtualisation (NFV) as the concept of transforming pure hardware appliances hosting network functions (NFs), for example, NAT, firewalls, intrusion detection and DNS, into software functions hosted on commodity hardware servers. According to Mulvey et al. (2019), *self-organising network solutions* are becoming more advanced, and they are best suited for improving operational efficiency in fault management.

2.12 Gap in domain not addressed

Network architecture changes frequently and new network elements are introduced regularly. These changes give little time to the NOC engineers to fully understand the alerts related to, and the impact of, faults when they occur. Rule-based algorithms as discussed in section 2.10 do not perform well in a changing environment. The disadvantage of rule-based automation is that the experts spend a great deal of time on the analysis of the data in order to be able to recommend the deployment of new rules. Human-free approaches can be effective; however, the users do not have insight into how the rule is constructed (Mulvey et al., 2019). There is no standard defined by IEEE and/or ITU-R that will enable MNOs or equipment suppliers to deal with this challenge. The fast rate of change exposes NOCs to miss the detection of service faults, which affects fault repairs, thereby impacting on the MNO. The labour cost of engineers makes it difficult for an MNO to increase the number of engineers and still be profitable. Addressing the workload demand with the addition of more employees is not sustainable. With more meaningful data available, an intelligent and fast solution is needed to process the data with limited to no human intervention. The solution needs to manage the dynamic nature of the environment.

2.13 Data analytics

Given that advanced organisations have a larger amount of information than can be made sense of, these organisations need a better approach to sort through the current heap of data so that their administrators can evaluate the pertinence and potential effect of this data (Loshin, 2003). Data analytics is the field that achieves this goal. Analytics is any information-driven procedure that provides an understanding. Advanced forms of analytics use mathematics and statistics to arrive at solutions. Business analytics explicitly use all types of analytics to accomplish business results. However, business analytics must yield precise ends and provide procedures which organisations can use (Schoenherr & Speier-Pero, 2015).

Organisations have multiple ISs to support their business; the norm is that data are stored and only a subset of that data is being consumed for the benefit of the organisation (Kibria et al., 2018). Underutilisation of data deprives businesses their competitive advantage and requires modern technological skills for leveraging the business. Telecommunications operations have multiple data components that are managed in silos. These components are in alignment with the ISO Telecommunications Management Network model. The framework divides the silos as follows; fault, configuration, accounting, performance and security. This approach limits the operators from gathering end-to-end insights from the available data, and this is the opportunity big data analytics presents (Lin et al., 2018). The growth of data is exponential in the information age, and is of great value for gaining competitive advantage through the efficient use of this data (Maździarz, 2018). Data analytics is complex as it involves large amounts of data from different sources in different formats. Expert knowledge and superior technology are needed in order to manage this data. This field has grown over the recent years and is referred to as the data science domain (Russom 2011).

According to Choi, Wallace and Wang (2018) use the vast amount of business data now available, companies need a way to process and understand information. Data analytics makes sense of all types of data for strategic planning purposes. Businesses can use analytics to learn how to make their firms stand out among their rivals. The purpose of a data analytics team is to deliver value to the firm by supplying information that spurs action. Analytics is valuable to construct a business case in which managers weigh the benefits of various activities, based on the analytical data for each alternative. The measured results of a data-based action plan show the value of analytics. To make the best use of the knowledge generated by data analytics, it needs to be communicated in ways everyone can understand. Cost to compute is a serious factor that needs to be considered in high alarm volume automation cases (Dorton & Liu, 2016) since model performance will depend on used resources. Business will reap more value if mined data are visualised correctly, thus meeting user expectations since the aim is for the end user to make sense of the information. Bandini et al. (2006) indicated that they hope future studies will deliver more realistic case studies.

Good quality decision-making has been the main drive behind data analytics over the past few years (Mikalef et al., 2018). The competitive advantage business is directly related to quality decision making at all levels of a company. The need to automate fault management is also common outside the telecommunications sector. Distribution Automation (DA) is deployed to reduce outages (Wang et al., 2018) and to rapidly reconnect customers following network faults. In the electricity sector, Pole Mounted Auto-Recloser (PMAR) was explored by automating the analysis of PMAR data in order to diagnose PMAR faults. The PMAR model (Figure 2.7) is well structured; however, the algorithm is highly dependent on SMEs defining what needs to happen (Wang et al., 2018).

As reported by Hubballi et al. (2011), there is a trade-off between model accuracy and detection rate. Business needs to choose which priority gets the response from a model fast or whether responses should take a little longer to ensure accuracy of the response. This principle is driven by the fact that computation resources are limited and the more the data to are analysed, the more the resources are required.

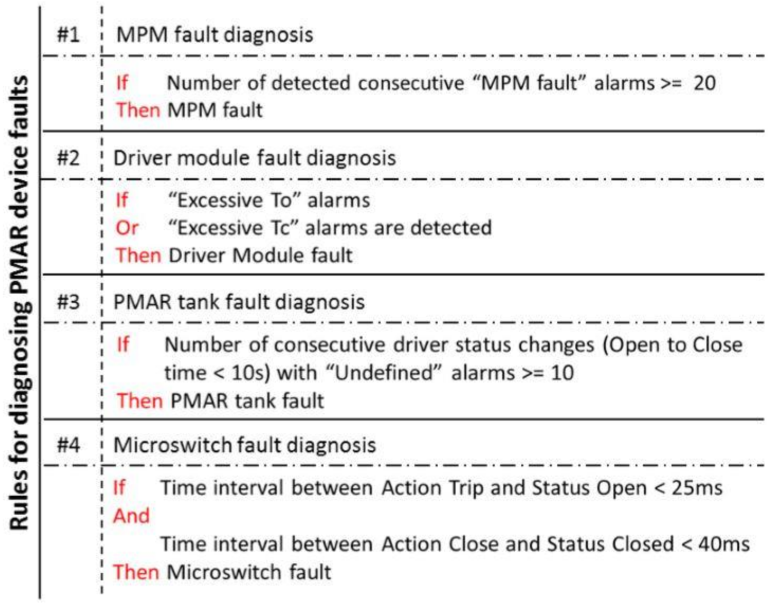


Figure 2.7: Rule-based PMAR solution (adopted from Wang et al., 2018:6267)

The study undertaken by Mikalef et al*.* (2018:558) informs us of three main categories of data analytics solutions:

i) Tangible resources (e.g. infrastructure, IS, and data)
ii) Intangible resources (e.g. data-driven culture, governance and social IT/business alignment)
iii) Human skills and knowledge (e.g. data analytics knowledge and managerial skills)

There is evidence that managers are positive about investing in data analytics; however, the same managers experience difficulty with accepting the insights provided by the analytics tool. These managers still believe that their intuition is more accurate than an analysis performed on large sets of data (Silva, Diyan and Han, 2019).

There are different classes of data analytics according to Swaminathan (2018:1698):

i) Descriptive analytics, describing the present
ii) Prescriptive analytics, prescribing solution(s); commonly used in dispatching or scheduling activities when relating to fault management tasks
iii) Predictive analytics, predicting the future state; commonly used in forecasting of supply planning

He et al*.* (2016:5) suggest that the application of big data analytics in MNOs can be divided into two categories:

i) Internal business supporting applications
ii) External innovative business model development

The internal business supporting applications mainly focus on efficiency and customer experience. Operations management (OM) as per Choi et al. (2018) is commonly known as the discipline which employs scientifically sound analytical methods to help organisations make optimal decisions. Big data analytics is a highly specialised and scientific field and involves processing different formats of data from different sources. Choi et al. (2018) highlight a gap in linking the available technology to reveal more insights into the application of big data analytics. In the past few years, the academic and business communities have experienced an increase in interest in big data (Troisi et al*.*, 2018).

High quality decision making, which improves efficiency across teams, is made possible by data analytics (Hamister et al., 2018). Many organisations have large volumes data stored in databases which add little to no value to the business. A database (DB) in simplistic terms (Langdon & Langdon, 2019) is an organised collection of data. There are different types of data originally stored in different forms. Combining all the data into one platform is a challenging task. Structured data are organised in a simple and constant format, such as data in a structured query language (SQL) database. Semi-structured data, on the other hand, is organised to some extend as one can figure out consistency when the data are carefully studied, for example, Man-Machine Language (MML) command output from a network device. Unstructured data are data in a recognisable structure, such as text messages, social media data, audio/ video material or location information.

The data analytics solution built on the foundation of expert workflows (Aye et al., 2017) is successful in retaining knowledge. Retained knowledge is more valuable as it can be used by any employee with any skill level. The primary use of a database is for retrieving, storing and managing information. Telecommunications engineers spend a great deal time running queries on systems in order to localise network failures (Kipf et al., 2019).

Zhang et al. (2018) completed a study in the manufacturing industry, which confirmed that an increase in the amount of manufacturing information available means big data can be collected with appropriate deep analysis. According to Feng et al. (2019), big data analytics (BDA) is a systematic approach to analysing data and identifying different patterns, relations and trends within a large volume of data. The growth and maturity of data analytics is evident across multiple industries over recent years (Feng et al., 2019). The need for a generic analytics system was identified and the requirement was driven by the need to reduce operational costs. The greater part of data analytics techniques in the manufacturing sector to date has been aimed at specific scenarios. Zhang et al. (2018) identified constraints in the ability of the R language to handle big datasets.

2.14 Machine Learning

Computer scientists developed algorithms with the ability to comprehend the world without requiring essential directions (Boutaba *et al.*, 2018). These algorithms are currently accessible in standard tool compartments that scientists who do not regularly use AI can undoubtedly utilise to distinguish designs in information. In the processing framework Mathematica, for example, only a couple of basic directions do the trick to harness AI’s special capacities (Hutson, 2019). In order to prepare a machine, scientists should initially characterise the parameters and objectives of the task in the Machine Learning (ML) algorithm. Next, a researcher should ‘feed’ the machine with information from which the machine can learn. One approach to this is the ‘learning student’ approach in which ML frameworks learn by helping, watching and copying humans. Computer-based intelligence is a 'universally useful innovation' with an extensive scope of conceivable applications (Brynjolfsson & Mitchell, 2017).

Implementation of algorithms to improve fault detections (Ran et al. 2018) has proven to be inefficient because of inaccuracies caused by false negatives. There is a trend towards introducing AI and ML techniques into SDN to fulfil network management automation requirements (Yu et al., 2019). The growth of networks results in large quantities of data which have to be analysed (Hubballi et al., 2013). Two categories of ML techniques are used for anomaly detection:

i) Supervised learning, using a labelled dataset
ii) Unsupervised learning, using an unlabelled dataset

Supervised learning techniques are commonly used to build a predictive model for normal versus anomaly classes. Unsupervised learning techniques are mostly used for grouping similar instances into clusters and labelling them. The research of Hubballi et al. (2013) revealed that support vector machines (SVMs) have a history of good performance in classification algorithm use cases. The only disadvantage of SVMs is the long time it takes to get the model trained. Clustering is a method for anomaly detection, one of the unsupervised techniques. ML approaches are more beneficial in automation of mundane tasks and can be used on large volumes of fault data.

Mulvey et al. (2019) considers self-healing as automated fault detection, classification and repair in the context of base stations.

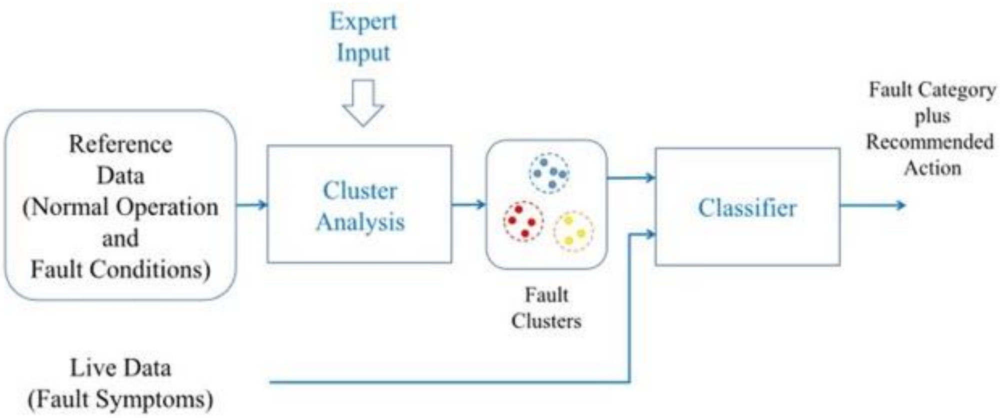


Figure 2.8: Clustering principle on alarms (adopted from Mulvey et al., 2019:124527)

Clustering techniques that were applied used a set of unlabelled alarms; this method did not need an SME. Bruno and Abrahão (2012) suggests that classification techniques assume a set of labelled alarms available for training a model. Clustering-based alarm mining techniques require more computation time since the dataset is scanned multiple times. Mulvey *et al.* (2019) demonstrated the application of clustering on alarms was applied, the model was trained, and live data were fed to deliver fault categories (Figure 2.8).

ML is ready to change the economy in principal ways by enabling machine learning engineers and data scientists to draw insights from organizations data, results shall unlock data driven decisions. Prior to ML, machines and programming supplanted human labourers performing centre aptitude and routine undertakings, for example, office assistants and assembly line labourers. ML, notwithstanding, will influence a completely different arrangement of employment classifications by supplementing labour force and improving productivity, crafted by profoundly gifted specialists (Brynjolfsson & Mitchell, 2017). For instance, ML may help legal advisors to find significant cases in law databases while human legal counsellors deal with meeting clients and drafting persuading legal arguments. With everything taken into account, ML is probably going to reduce the general interest in work for majority of the worker that enjoyed solving complex technical problem since the self learning ability of the algorithm will do that work better than people. However, specialists differ on how much job losses ML will cause, taking into account the fact that ML-upgraded human movement will probably prompt the development of new items and administrations that will create new sorts of assignments and occupations (Brynjolfsson & Mitchell, 2017).

People will keep on beating machines at a wide scope of complex tasks. ML exceeds expectations in organisations that harvest meaning data and apply systematic approach in identification of use cases for ML consideration (Boutaba *et al.*, 2018). At the point when people settle on complex choices, they draw on a wide range of factors and past encounters – data that a prepared ML framework might not have access to, or will not perceive the significance of. In addition, computers, despite everything, come up short in the human capacity of working out ‘good judgement’. While ML frameworks may become capable at associating information, they do not proceed to the clarification of causal connections. For instance, computers can identify particular kinds of malignant growths with incredible exactness, yet it takes a trained and experienced doctor to clarify the results reached by the algorithm. In fields where the realities on the ground change rapidly, ML is less appropriate since it will not have the option to adjust to the new conditions sufficient quickly to deliver a precise outcome. ML is inclined to propagate imperfections and inclinations in the preparation information. For however long there are (human) blunders in the preparation information, ML algorithms will not have the ability to perform tasks with immaculate precision (Hutson, 2019).

2.15 Artificial Intelligence

AI applies computer systems to tasks that once required human intelligence. A long-standing debate within the AI community revolves around whether AI should augment the human mind or replace the work done by the human mind. Either way, AI and automation will fundamentally reshape the workforce. Firms profit by developing cognitive strategies for converting business problems to intelligent solutions through use os AI. Decomposing raw data is regularly the focal point of AI to the point that makes sense for a user to understand what the data entails (Davenport, 2018). There are different tools and techniques applied in the AI domain. Think about internal and external objectives of a company, for example, serving clients better and making progressively well-educated 'inner choices'. Some strategies can be developed that become AI-driven plans of action that place AI at the centre of future improvements.

The abilities of AI can be developed through supervised or unsupervised learning. In supervised learning, which is more common, people train AI systems using data and guide the system to make distinctions, for example, to distinguish between pictures that show dogs and pictures that do not show dogs. In unsupervised learning, systems start with data that mean nothing to them and they identify patterns on their own (Agrawal, Gans and Goldfarb, 2019).

AI is not a hypothetical development that might appear sometime in the future. Businesses utilise AI today, and AI transform how they work. Many consumers experience AI today in the form of virtual helpers like Siri (Apple's virtual assistant with speech capability). An AI framework has multiple capabilities. The capabilities in the first set are speech and recognition, image recognition, search and clustering. Image recognition involves tagging images and making distinctions between them. When machines capture information, they convert unstructured data (big data) into structured data. This requires fast processors and a great deal of training. Certain capabilities make AI immediately useful. For example, speech recognition lets people give machines direct commands.

In the real world some organisations use AI in order to improve customer service, for example, via chatbots. Simple chatbots can answer only yes/no or multiple-choice questions. However, chatbots that receive extensive training through thousands of human-to-human chat conversations can answer questions and help customers place orders (Blinnikova and Ying, 2020).

A recommendation AI engine such as Amazon and Netflix, use data from customer purchases to suggest future purchases. AI processes quickly improve functions that the customers will never see. Business leaders who want to work with AI should identify the challenges their companies face and ask how AI could help overcome these challenges. Leaders should consider AI and automation together and then decide what they want such systems to accomplish. They can try a solution or application on a small scale, test it, and then apply it more broadly. Businesses should align their AI strategies to their overall strategies (Gotthardt *et al.*, 2020).

2.16 Summary

Critical assessment of prior work by examining reliable studies helped the researcher to understand the key concepts and their relevance to the problem under investigation. Literature was reviewed by applying a funnel approach, starting with the South African telecommunications market, then literature dealing with Network Operations Centres, before narrowing it down to the key concepts of the NOC. The gap in the existing knowledge was established. To be more specific, the main take away from this chapter is that rule-based algorithms are only effective for short-term benefit at the cost of depriving an organisation of data that could be converted into useful information suitable to gain a competitive advantage.

CHAPTER THREE: RESEARCH METHODOLOGY

The chapter flow:

- Introduction
- Research philosophy
- Research approach
- Research methodology
- Research design
- Data collection
- Data analysis
- Data validity and reliability
- Reflection
- Ethical considerations
- Summary

3.1 Introduction

In this chapter, the researcher lays out the research approach followed in the study. Only academically sound methods were explored. In this chapter, the researcher deals with the research methodology suited for this research with reference to the problem statement. The methodology is in alignment with the aim and objectives of the study. Data collection, analysis and governing aspects assuring quality are also covered in this chapter. The researcher studied the available literature on research methodology and this played a very important role, considering that this study must comply with the academic requirements of the university.

3.2 Research philosophy

A qualitative-pragmatic research approach was followed to explore the use of data analytics in network operation. A wide range of methods are available for deciphering the world and undertaking research, that no single perspective can ever give the whole picture, and there might be numerous factors influencing the study. The ontological position taken in this study was constructivism, and the epistemological position was interpretivism.

Qualitative research is the methodology used in research related to the social constructivist worldview, as a rule. This methodology stresses the socially developed nature of the real world (Caldas, 2009; Pandey, 2016). The methodology is tied in with the recording, breaking down of and endeavouring to reveal the more profound importance and essentialness of human conduct and experience, including opposing convictions, practices and feelings. Qualitative researchers are keen to build on a rich and complex comprehension of individuals' understanding, rather than getting data which can be collected and aggregated from larger focus groups (Yannis & Nikolaos, 2018). Qualitative researchers do not formulate their exploration with respect to pre-decided theories. Overall, they obviously recognise an issue or theme that needs to be investigated and possibly guided by a hypothetical focal point, which can be viewed as a larger hypothesis that gives structure to their examination.

For qualitative research, the way to deal with an assortment of information and conducting the investigation is precise, yet this technique is much more adaptable in terms of the information that is taken into account than quantitative research. Information is gathered in literary structure based on the perceptions of, and collaboration with, the respondents through, for example, determining respondent perceptions, and hosting top to bottom meetings and focus group meetings (Pandey, 2016).

Interpretivism and pragmatism are two potential and significant ideal research models for qualitative research in IS. Interpretivism is more applicable to an understanding of information, while pragmatism is more applicable to acquiring constructive knowledge that is useful in actions (Goldkuhl, 2012b). The practical nature of this study has led to the pragmatic approach being adopted.

Pragmatism is applied to explore reasoning, acknowledging that ideas are significant in the event that they bolster activity. Pragmatists perceive that there is a wide range of methods for deciphering the world when undertaking research, that no single perspective can ever give the complete picture and that there might be numerous real-world factors to consider (Da Silva et al., 2018). Pragmatists can consolidate both positivist and interpretivist positions inside the extent of solitary research, as indicated by the idea of an explorative question (Goldkuhl, 2012b). As a research paradigm, pragmatism depends on the recommendation that researchers should utilise the philosophical as well as the methodological approach that works best for the specific research topic being explored (Goldkuhl, 2012a). The pragmatic IS researcher generally has a good chance of taking an interpretive stance since social practices are important in a study. Pragmatism strongly decrees that action and the practical aspects of the study are key points of scientific research (Feilzer, 2010; Smith et al., 2011). This study is an ideal case in point since a group was approached and 'pain points' (section 3.5) were determined, which led to the research problem and the researcher exploring various concepts that led to the design and development of an artefact.

The key concept that drives constructive research is the development of a solution by using current knowledge in novel ways. The development continues through formulating a belief that can be realised in future hypotheses and fills theoretical and other knowledge gaps (Zupan & Medhi, 2003). Constructivism is an epistemology applied to describe how people know what they know. A person's experience of solving problems and discovering new insights is incrementally building his/her understanding (Mills, Bonner & Francis, 2006).

This information systems study creates knowledge that can be applied to a MNO, specifically in network management. DSR explicitly deals with the application of existing knowledge in order to address fascinating and important problems and has consistently been applied in information systems research. Be that as it may, design science is not a design process, but the science covering the journey from problem identification to assessing the effectiveness of the built solution (Piirainen & Gonzalez, 2013).

This study considered two paradigms that portray a significant part of the research in the Information Systems (IS) discipline, namely behavioural science and design science. The behavioural science worldview tries to create and check hypotheses that clarify or anticipate human or hierarchical conduct. The design science worldview tries to expand the limits of human and authoritative capacities by making new and inventive artefacts (Hevner et al., 2004). This study followed the Design Science Research (DSR) approach. DSR has staked its legitimate claim as a significant and genuine (IS) research paradigm (Gregor & Hevner, 2013). DSR focuses on the creation of artefacts.

3.3 Research approach

There are two types of approaches, the deductive approach and the inductive approach. The deductive approach aims to test a theory and the inductive approach, also known as inductive reasoning, focuses on generating new knowledge. Studies have related the deductive approach to quantitative research, while the inductive approach is associated with qualitative research (Maxwell, 1992). This study adopted a combined approach that tends more towards the inductive approach. The problem did not originate from theory; it originated from observation (engagement with users). The general inductive approach provides for a handily utilised and methodical arrangement of systems for breaking down qualitative information that can deliver dependable and substantial discoveries. It is important to understand that the inductive approach does not infer dismissing theories when defining research questions and objectives.

3.4 Research methodology

3.4.1 Understanding Design Science Research Methodology

DSR involves identifying a pertinent and significant problem in an area of study, making or building an artefact that takes care of the problem, and assessing the value of the made artefact (Peffers et al., 2007).

A DSR venture starts with the distinctive proof of a difficult problem or opportunity that is academically worthy. The prerequisites for the research are characterised alongside the acknowledgment criteria for the evaluation of the integrity of a design solution. The research group explores current knowledge on the basis of enlightening and prescriptive knowledge to understand what has been done to date on the problem, whereupon new knowledge will be developed in the venture (Baskerville et al., 2018). In information systems, DSR includes the development of a wide scope of socio-technical artefacts, for example, decision support functions, displaying devices, administration systems, techniques for IS assessment, and IS change interventions (Gregor & Hevner, 2013).

3.4.2 Design Science Research history

According to Weber (2010), DSR originated in the engineering discipline and then also evolved to the IS domain. A few researchers spearheaded DS research in information systems, yet, over the course of recent years, little DS research has been done to understand which control measures can be applied to quantify the quality of the study. As of late, a few researchers have prevailed with bringing configuration research into the IS research network, effectively presenting a defence for the legitimacy of design science (DS) as an IS research paradigm (Peffers et al., 2007).

Studies dating back as early as 1969 were based on design principles and then in 1992, a clearly defined DSR emerged (Table 3.1). DSR has become well known and much used by postgraduate students, mostly in the IS domain. An Australian study that involved 40 DSR doctoral theses undertaken between 2006 and 2017 confirmed the increase in the use of DSR (Cater-Steel et al., 2019). Senior researchers have strongly emphasised the possibility of the incorrect application of DSR by junior researchers because of the complexity of the methodology, which can be avoided by studying available material on the subject.

Table 3.1 History of DSR (adopted from Weber, 2010:5)

Reference	Topic	Description
Simon (1969)	Design	Provides a discussion on the design of the artificial.
Mantei and Teorey (1989)	IS Development	Describes the IS developmental life cycle on the example of a user interface for a DBMS.
Walls et al. (1992)	DSR basic literature	Depicts a milestone of DSR and provides a first DSR framework. Moreover, this article contains basic definitions of DSR and of IT artifacts
March and Smith (1995)	DSR basic literature	Provides basic definitions of DSR and a framework for creating an IT artifact It is the first article that tries to place DSR research in context to other disciplines and paradigms.
Orlikowski and Iacono (2001)	DSR basic literature	A detailed description of an IT artifact.
Hevner et al. (2004)	DSR basic literature	Depicts a milestone of DSR. Basic definitions of DSR and IT artifacts are provided in this article. Moreover, it conceptualizes a new framework for creating an IT artifact (7 Guidelines for DSR in Information Systems Research). Additionally, it evaluates the guidelines on the basis of three different Design Research (DR) papers.
Carlsson (2006)	DSR general literature	Provides a definition of DSR derived from Hevner et al. (2004) and Walls et al. (1992). Additionally, it classifies DSR as critical realism.
Baskerville (2008)	DSR general literature	A short introduction of DSR and IT artifacts.
Winter (2008)	DSR general literature	A short introduction of Design Science (DS) and Design Research (DR).
Kuechler and Vaishnavi (2008)	DSR general literature	Describes kernel theories and suggests a framework for DSR.
March and Storey (2008)	DSR general literature	Provides a definition of DSR and 5 examples of literature that uses the DSR approach.
Markus et al. (2002)	DR	Applies the DSR approach to the example of TOP Modeler (an IT artifact that supports emergent knowledge processes).
Aalst and Kumar (2003)	DR	Describes the creation of a new IT artifact (XML-based schema for inter-organizational workflow).
Weedman (2008)	DR	Describes the creation of a new IT artifact (Sequoia 2000; the IT artifact brought together prestigious scientists from computer science and earth science to get in collaboration).
Umapathy et al. (2008)	DR	Describes the creation of a new IT artifact (embedding a mechanism that helps designers develop integration solutions based on recurring solutions captured as patterns).
Gregg et al. (2001)	DSR vs. behavioral science	Classifies DSR as a separated paradigm called socio-technologist or developmentalist paradigm.
Holmström et al. (2009)	DSR vs. behavioral science	Describes the difference of DSR and behavioral science. Thereby, it explores 4 phases which are derived from the different approaches.

3.4.3 Methodology

DSR has been a significant paradigm of IS research since the origin of the field, and its general acknowledgment as an authentic way to deal with IS research is expanding. The DSR approach consolidates the benefits of various paradigms that are applied in traditional research outside the IS domain.

DSR is a well-established academic tool that has been accepted by multiple universities as research methodology. Gregor and Hevner (2013) outlined the below schema for DSR research (Table 3.2).

Table 3.2: Publication schema for DSR study (adopted from Gregor & Hevner, 2013:348)

Table 3. Publication Schema for a Design Science Research Study	
Section	**Contents**
1. Introduction	*Problem definition, problem significance/motivation, introduction to key concepts, research questions/objectives, scope of study, overview of methods and findings, theoretical and practical significance, structure of remainder of paper.* For DSR, the contents are similar, but the problem definition and research objectives should specify the ***goals*** that are required of the artifact to be developed.
2. Literature Review	*Prior work that is relevant to the study, including theories, empirical research studies and findings/reports from practice.* For DSR work, the prior literature surveyed should include any prior design theory/knowledge relating to the class of problems to be addressed, including artifacts that have already been developed to solve similar problems.
3. Method	*The research approach that was employed.* For DSR work, the specific DSR approach adopted should be explained with reference to existing authorities.
4. Artifact Description	A concise description of the artifact at the appropriate level of abstraction to make a new contribution to the knowledge base. This section (or sections) should occupy the major part of the paper. The format is likely to be variable but should include at least the description of the designed artifact and, perhaps, the design search process.
5. Evaluation	Evidence that the artifact is useful. The artifact is evaluated to demonstrate its worth with evidence addressing criteria such as validity, utility, quality, and efficacy.
6. Discussion	*Interpretation of the results: what the results mean and how they relate back to the objectives stated in the Introduction section. Can include: summary of what was learned, comparison with prior work, limitations, theoretical significance, practical significance,* and *areas requiring further work.* Research contributions are highlighted and the broad implications of the paper's results to research and practice are discussed.
7. Conclusions	*Concluding paragraphs that restate the important findings of the work.* Restates the main ideas in the contribution and why they are important.

DSR can be directed in a formative way by concentrating on the building process as applied in Chapter Four of this thesis or in an interpretive way by producing theory out of the developed IT artefact and its utilisation (Weber, 2010). Experience shows that numerous developers, analysts and editors battle to display and decipher DSR function admirably due to a limited understanding of the methodology. The challenges here likely arose from a blend of issues, which include the IS information innovation disciplines of young people, and the late acknowledgment of DSR as a distinct, yet real, research paradigm.

The DSR method involves:

i) Identifying the problem
ii) Defining solution objectives
iii) Design and development
iv) Demonstration
v) Evaluation
vi) Communication (Gregor & Hevner, 2013)

A developing inquiry seeks to determine whether DSR delineates an isolated paradigm or whether DSR is an approach that can be joined to other approaches. In the light of this, a few endeavours have been made to consolidate both the first underlying foundations of DSR and the improvement of hypothetical commitments of the approach (Weber, 2010).

Gregor and Hevner (2013) emphasised the importance of identifying a research-worthy problem that will lead to artefact design and development as a minimum requirement for a researcher to consider applying DSR. This view was well presented in their model (Figure 3.1).

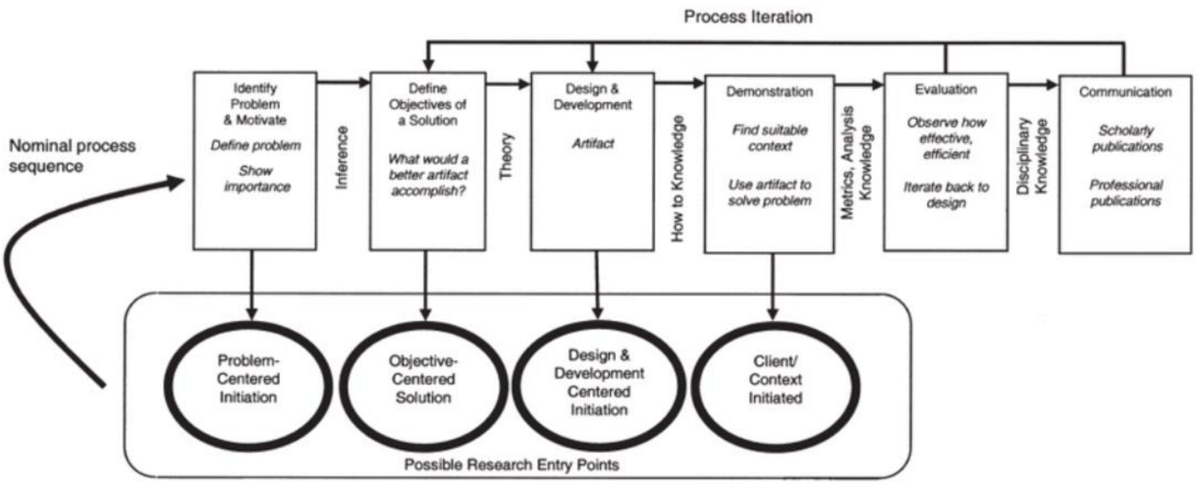
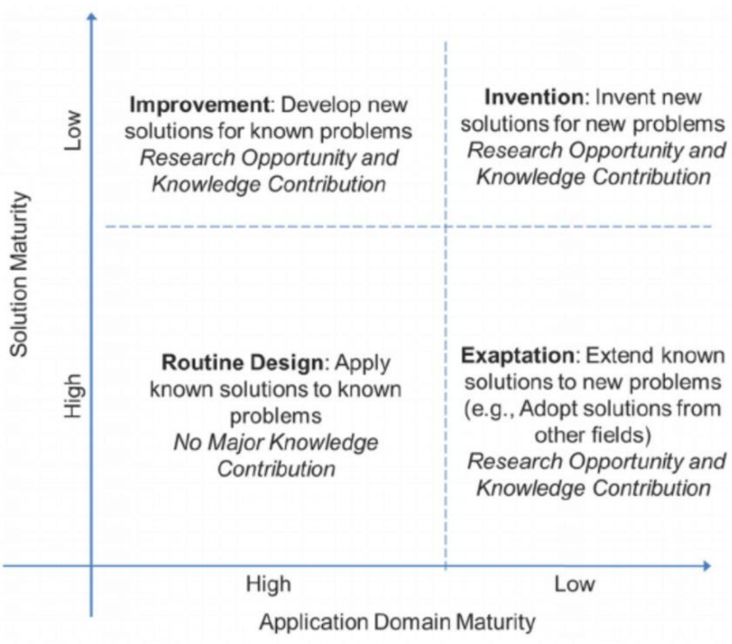


Figure 3.1: DSRM process model (adopted from Gregor & Hevner, 2013:54)

This particular study is classified the 'new solutions for known problems' study (Gregor & Hevner, 2013) using the four-quadrant model (Figure 3.2).



C2 General

Figure 3.2: DSR knowledge contribution (adopted from Gregor & Hevner, 2013:345)

Many years of technological changes affecting industries have changed human social orders; design is vital to these developments (Baskerville et al., 2018). The significance of the practical problem must be advocated to allow industries to keep up with the rate of solving problems. DSR has two main objectives, namely: i) to identify the right problems that need to be solved; and ii) information creation.

Design science artefacts can be arranged as constructs, models, methods and instantiations. Constructs provide another dialect to portraying an issue at hand, models provide conceptual definitions of the issue, and methods provide procedures for taking care of the issue regularly in formal terms, while instantiations provide helpful usage of constructs, models and methods (Kogan et al., 2019). A variety of assessment approaches can be used in DSR, including case and field studies, explanatory investigation, controlled trials and reproductions, and practical and auxiliary testing, such as educated arguments and situation examinations.

DSR is not part of the predominant information systems research culture. Design science enables researchers and organisations to efficiently identify gaps that are affecting operational efficiency and apply solutions that consider available techniques. The most significant aspect is that the research must deliver a unique solution that is addressing the problem and this must be confirmed through thorough evaluation of the artefact with reference to the problem statement. Baskerville et al. (2018) highlight the importance of quality in DSR, identifying the five pillars described in Table 3.3.

Table 3.3: What it means to perform quality DSR projects (adopted from Baskerville et al., 2018:359)

DSR Objective	Positioning Statement	Publication Guidance
Technology and science evolutions	DSR projects must be correctly positioned in the cycles of science technology evolution for the chosen application domain.	• Assess the maturity of the scientific (descriptive) and the technology (prescriptive) knowledge bases for the DSR project domain • Identify the appropriate starting points for the DSR project to demonstrate clearly new knowledge contributions made by the project
Design artefacts	A novel IT artefact must be built and evaluated in a DSR project. The design artefact most often comes before the development of nascent and mid-range design theories.	• Represent the IT artefact clearly • Demonstrate its novelty and its practical improvements, first, then move to reflecting on design theory contributions
Design theories	The distinctive value and origins of design knowledge, as represented in both artefacts and theories, should be recognised as both inputs and outputs of a DSR project.	• Traditional kernel or reference theory to support DSR is important but is not always feasible, particularly with research that involves creativity • Some design theorising is expected for DSR as a reflection on the advance in design knowledge • There is no requirement that non-design theory contributions should be made in DSR
DSR processes	Reference processes are available for conducting DSR and these should be followed, or contributed to, through exemplar research process descriptions.	• Use a reference process for guidance in performing the research • Reflect on the reference process when reporting research and propose process improvements as appropriate
DSR impacts	Design artefacts should impact on practice and design theories should impact on research.	• DSR research must report the kinds of impact planned and achieved by the resulting IT artefact • The impacts of the artefact and the design theory are cumulative, cohesive, and inseparable

3.4.4 Relevance to the study

The nature of the study is practical, and it involved establishing the problem through studying participants and presenting a working solution (artifact) to address the problem. The study was done as a university research project, exposing the researcher to the academic way of performing research. The researcher has 13 years of experience in telecommunications engineering, from an operational to a management level. This experience contributed to finding clarity in terms of the research context. DSR has proven to be best suited for this type of research. The important aspect to consider is the fact that the research was done with the objective of meeting the requirements for the Master of Technology degree in Business

Information Systems. The literature reviewed identified the strong link between design science research and information systems. The specific area being studied (NOCs) has rich data that were not explored to the fullest potential for business. DSR is agile in nature and has been established itself in academia in recent years. The project stages will be highlighted based on based on Figure 3.1.

3.5 Data collection process

The first DSR session took place where the researcher was introduced to the team (Appendix J) and the study details were shared. The objective of the session was to align the expectations of the participants in the research with those of the researcher. In this session, the structure of the team and how the department fits into the business was explained to the participants. Individuals were identified as study leads and it was agreed that they were to be the single point of contact (SPOC) to the researcher because the nature of the work required teams to work in shifts. A pilot interview was conducted with the SPOCs, with the objective of testing participation readiness for group interviews. The pilot session indicated that an improvement scheduling of the contact sessions was required. The set time of 08h00 in the morning was not conducive as the shift team was still concluding shift handover activities.

The first group interview was conducted, and in this interview, the researcher enquired about operational pain points the team experienced. Team members described their working environment and the challenges they encountered. The inputs were recorded by the researcher. Grouping of similar pain points was done and items on the list were prioritised. The researcher collected the inputs from the participants, consolidated them into one document, and scheduled a follow up session with the team in order to confirm whether the details were captured correctly. The team agreed that the content of the document was a true reflection of what they had said and their input was adopted as final requirements for the researcher to investigate.

The researcher's access to access the database that stored the raw data obtained from the alarms was granted. Data were explored and domain knowledge was sourced from the senior network engineer. The data studied were extracted from a Structured Query Language (SQL) database. A combination of explored and unexplored data led the researcher to the conceptualising phase as detailed in Chapter 4. The design and development of the artefact followed.

A testing session was scheduled with the study participants. In this session, the researcher demonstrated an artefact that addressed the top three priorities on the list obtained from the participants, which were agreed by the organisation under study to be part of Iteration 1. The users confirmed the benefits brought to the table by Artefact 1.

Additional adjustments were requested:

i) Add a feature to categorise data insight by network device type
ii) Build a feature to simplify incident management analysis

Iteration 2 efforts started with using the feedback provided by the users (participants). The researcher consolidated and documented the user requirements, and then shared these requirements during a validation session. The validation process was undertaken so that the researcher could confirm that he understood the problem communicated by the users and that the users agreed with the documented requirements. The 'categorising by network element type' feature was a medium effort as the task involved intermediate data preparations skills, which was done using the data that had already been studied by the researcher. The specification of an incident management artefact required a major effort on the part of the researcher, which involved learning new data sources from the incident management database. Domain knowledge was sourced from the senior network manager, which gave the researcher a solid understanding of the data. The design and development phase followed. Users were invited to a demonstration/testing to specifically test the artefact against the agreed upon requirement. The session went well, during which the requirement to add automated alarm categorisation was raised.

Iteration 3 then commenced. Experience from the first two iterations was used to good advantage by the researcher in Iteration 3, as the alarm data and teamwork were well understood. The requirement proved to be more complicated since the users needed the categorisation of faults to happen without human intervention. The researcher studied ML algorithms and assessed the algorithm most applicable to the problem. The learning of the ML algorithms involved both theory and practical aspects. The artefact concept was drafted, the design was planned and the solution was developed. A session was held with users to demonstrate the proposed ML artefact. The results were not well received and it was agreed that no new requirements would be given to the researcher, but rather that time would be given to the researcher to fine tune the artefact. Because of its complexity, development of the artefact took longer than the expected; however, the development was eventually concluded. A session was scheduled with the users where the artefact was well received. Because the researcher had limited time (a two-year Master's study), the process of taking on additional user requirements was carefully controlled.

3.6 Data analysis

The data analysis was done in conjunction with the data collection because this is what the DSR methodology requires. The researcher worked in iterations: collecting data, analysing data, and developing the artefact.

3.7 Validity and reliability

The outcomes of this study will be of benefit to the Network Operations Centre of mobile network operators in South Africa because using a data analytics artefact will help reduce manual analysis undertaken by NOC engineers when assessing the state of a network from alarm data. The senior engineers that have been in the telecommunications industry for over a decade, working at a leading mobile network operator, were interviewed by the researcher to gather high quality information. The identified problems were further confirmed by the senior manager in the NOC department. Participants were contacted to validate the recorded requirements (operational pain points) to ensure that there was no misalignment.

System data collected in the exploratory data analysis was recent (less than 12 months old) and extensive, ensuring that the analysis was not based on insufficient input data. The actual value to business is *improved customer service* as a result of human-free analysis. This study contributes to the body of knowledge through introducing a well-structured and simplified mechanism of applying ML to data analysis in the NOC.

3.8 Ethical considerations

Authorisation to collect data was granted in writing by the organisation (APPENDIX B: ETHICS FORM) through:

i) Interviewing employees
ii) Extracting system data

Each participant was made aware that their participation in the study was voluntary and they had to option to opt out of the study at any time. The meeting with the team took place on the employer's premises, in compliance with basic conditions of employment, especially from a health and safety aspect.

The ethics procedure of the university (Cape Peninsula University of Technology) was followed under the supervision of Dr Andre de la Harpe, and the study was granted permission to proceed. A Memorandum of Understanding (APPENDIX A: MoU) was issued by the university and signed by the institution’s research leadership. The Cape Peninsula University of Technology is in the Republic of South Africa. The researcher ensured that no activity of the study violated the constitution of the country

(Constitution of the Republic of South Africa Act 108 of 1996) (Department of Justice and Constitutional Development, 1996).

3.9 Summary

In this chapter, the researcher confirmed DSR as the optimal method for this study because the intention of the researcher was to address a problem identified by the study, with the expected output being an artefact. The reviewed literature informed the researcher that DSR is applied by other researchers in the IS discipline. Details of the research approach and the research methodology in relation to the problem as well as in relation to data collection and analysis were provided by the researcher.

CHAPTER FOUR ARTEFACT CONSTRUCTION

The chapter flow:

- Introduction
- Alarm management
- Incident management
- Problem identification
- Requirement planning
- Design and development
- Prototype demonstration
- Summary

4.1 Introduction

In this chapter the researcher focused on the problem identification, conceptualisation, design and development of the prototype. Understanding the context of the researched area is important, as it relates to a discussion of the problem investigated and to the solution that forms the first part of this chapter.

4.1.1 The Problem Context

In order to provide context, the case used will now be discussed. The study area served a central function, meaning that the work done by the team was not performed elsewhere in the organisation. The primary function of the team is to manage network faults. Network elements generate alarms in the case of network failure or degradation. All alarms are forwarded to a standardised network monitoring tool that the NOC uses for fault detection. Based on alarm events received, the members of the NOC perform investigative actions to identify the root cause of the fault and to take the fault through the incident management process. The organisation at large is dependent on work carried out in the NOC; internal departments or business areas listed in the customer segment of the business have an immediate and huge impact on the NOC. It was important for the researcher to understand the context of the NOC, and the business layout below (Figure 4.1) simplified the process. Not all network faults can be avoided; therefore, the need to manage faults efficiently has and will continue to exist so long as there is network infrastructure to be maintained. Network outages have far-reaching consequences on revenue loss and customer experience, as outlined in the revenue streams part of the layout. It is clear that the NOC plays a critical role in service assurance. The ability to detect faults and take action quickly, especially when dealing with service-impacting faults, is of utmost importance. Tasks on a granular level that provide value to the organisation are listed in the value proposition.

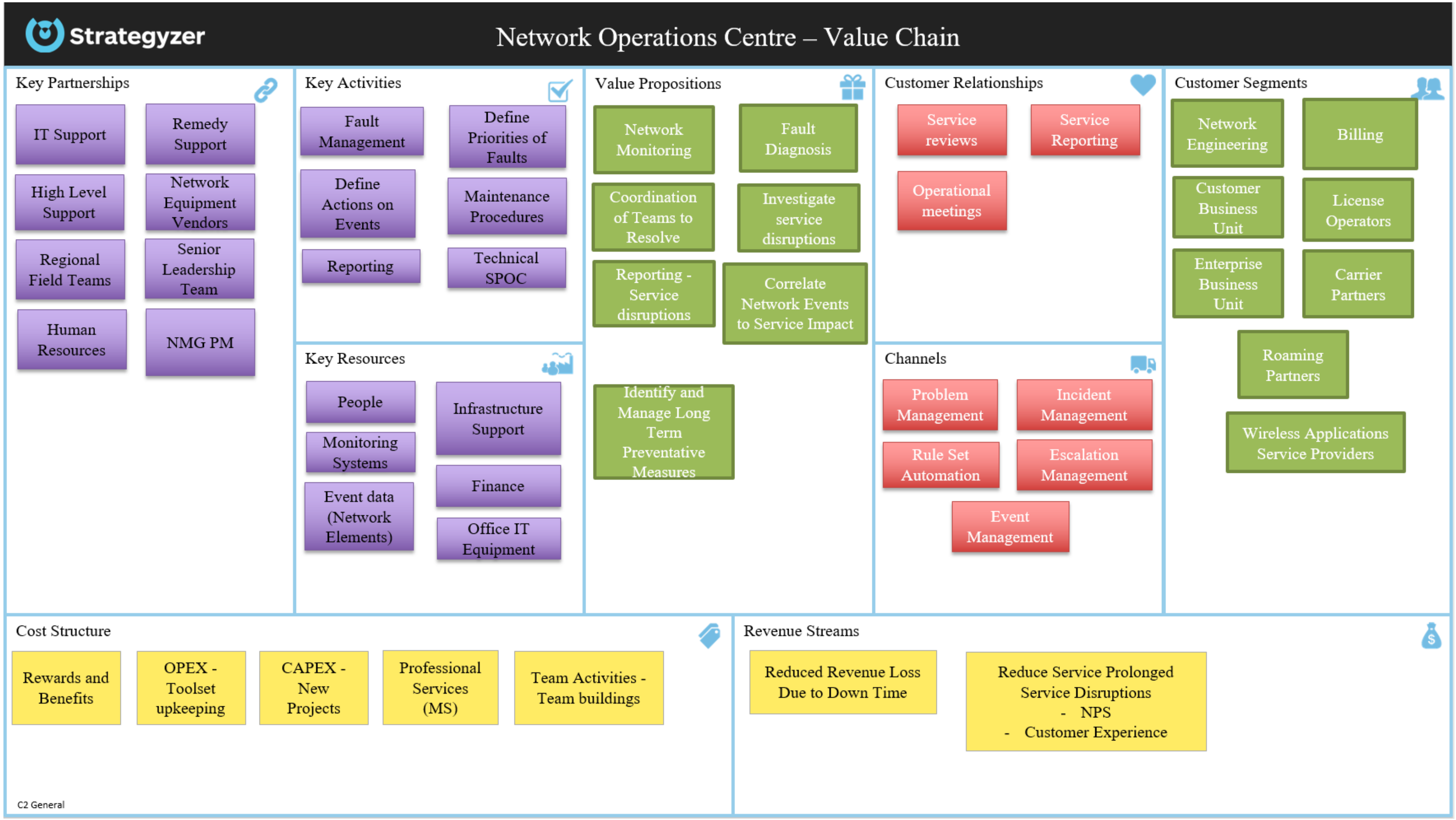


Figure 4.1: NOC value chain

The NOC serves multiple areas and inter-department relationships are managed through service reviews, operational reports and meetings. Network monitoring and incident management is fulfilled by three distinct areas (in green blocks), as per organisation structure below (Figure 4.2).

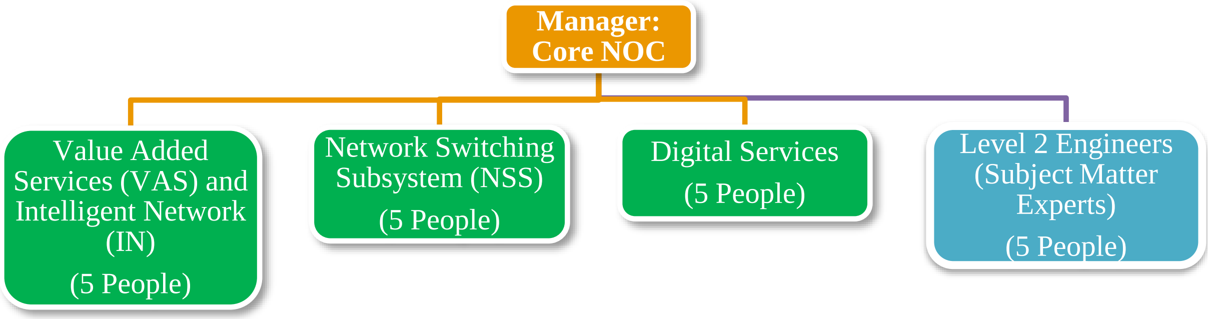


Figure 4.2: Organisation structure

The Level 1 NOC team (in green blocks) works 24 hours a day, 7 days a week rotating shift cycle, including weekends and public holidays. The Level 2 team is made up of highly skilled engineers who support Level 1 by handling complicated faults and deploying rule set based automations. The deployment of rule set automations is based on the Level 2 engineers' experience.

4.1.2 Alarm management

Large amounts of alarms are received, giving very limited time to the Level 1 team to do quality analysis. Rule set based automations did provide relief to some extent; however, it was clear that a smart, automated solution was needed as detailed in the problem identification section (4.2.1). The alarm presentation mechanism was clearly effective in the early years of telecommunication, where variations in alarm details were minimal and networks were smaller.

Figure 4.3 is the network monitoring view. Critical events are indicated in red, while orange indicates major events, yellow indicates minor events, blue indicates warnings and purple indicated indeterminate. Node is the network element reporting alarms, and applicable policies indicate the deployed rule-based automation applied. Correlation and delay serve the purpose of reducing alarms, and AutoTT is for automated incident creation. The summary field contains detailed information that enables the engineer to understand the nature of the problem (note that it is not in a format that is readable by humans).

LastOccurrence	TechType	Node	Alert Key	ApplicablePolicies	TTRef	TTStatus	Summary	AMO
11/05/2020, 21:41	MSS	MSPRL	SGS_SGSAP_LINK_FAILURE	7=AutoTT,5=Correlation	INC000017540052	In Progress	00000001	ManagedElement=MSC-373024
11/05/2020, 21:41	MSS	MSPRL	SGS_SCTP_ASSOCIATION_LOST	7=AutoTT,5=Correlation			00000001 00000001	ManagedElement=MSC-373024
11/05/2020, 21:41	MSS	MSPSO	SGS_SCTP_ASSOCIATION_LOST	7=AutoTT,5=Correlation			00000001 00000001	ManagedElement=MSC-372984
11/05/2020, 21:41	MSS	MSPCE	SGS_SCTP_ASSOCIATION_LOST	7=AutoTT,5=Correlation			00000001 00000002	ManagedElement=MSC-404655
06/05/2020, 21:06	MSS	MSLKK01	ROUTE_SET_UNAVAILABLE	5=Correlation			00000501 08 04	ManagedElement=MSC-326915
30/04/2020, 07:41	MSS	MSMSU01	ROUTE_SET_UNAVAILABLE	5=Correlation			00000501 08 04	ManagedElement=MSC-396741
30/04/2020, 12:17	MSS	MSMTR	OVERFLOW_IN_HANDLING_ALARM_EVENTS	2=Delay			00002241 058A0000 00	ManagedElement=MSC-396216
30/04/2020, 12:18	MSS	MSMTR	ROUTE_SET_UNAVAILABLE	5=Correlation			0000300B 08 00	ManagedElement=MSC-396216
29/04/2020, 18:50	MSS	MSMSU01	ROUTE_SET_UNAVAILABLE	5=Correlation			00003331 00 00	ManagedElement=MSC-396741
29/04/2020, 22:21	MSS	MSCTN	ROUTE_SET_UNAVAILABLE	5=Correlation			00003835 08 00	ManagedElement=MSC-373001
07/05/2020, 21:57	MSS	MSPFS	ROUTE_SET_UNAVAILABLE	5=Correlation			00003835 08 00	ManagedElement=MSC-372803
30/04/2020, 12:18	MSS	MSMTR	ROUTE_SET_UNAVAILABLE	5=Correlation			000038E3 08 00	ManagedElement=MSC-396216
30/04/2020, 12:18	MSS	MSMTR	ROUTE_SET_UNAVAILABLE	5=Correlation			000038EB 08 00	ManagedElement=MSC-396216
30/04/2020, 12:18	MSS	MSMTR	ROUTE_SET_UNAVAILABLE	5=Correlation			000038F3 08 00	ManagedElement=MSC-396216
12/05/2020, 09:07	MSS	MSMTH	FAILURE_IN_D-CHANNEL_ACTIVATION_OR_RESTORATION	2=Delay			0020 00 00 65535d	ManagedElement=MSC-397825
12/05/2020, 09:16	MSS	MSJMR	ANNOUNCEMENT_ID_ANALYSIS_ERROR	2=Delay			0030 07 09 00 00 30 00 00 00 00 00 00 00 00	ManagedElement=MSC-913028
06/05/2020, 14:55	MSS	MSCFO	ANNOUNCEMENT_ID_ANALYSIS_ERROR	2=Delay			0030 07 09 00 00 30 00 00 00 00 00 00 00 00	ManagedElement=MSC-373003
02/05/2020, 09:44	MSS	MSDNR	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	2=Delay,5=Correlation			0034 00 00 05 0B 27 82 98 99 81 60 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-372857
01/05/2020, 22:16	MSS	MSBEO	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	2=Delay,5=Correlation			0034 00 00 05 0B 27 82 98 99 82 80 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-913031
11/05/2020, 18:36	MSS	MSPPE	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	2=Delay,5=Correlation			0034 00 00 05 0B 27 82 98 99 86 20 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-373025
11/04/2020, 17:51	MSS	MSDTJ	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	2=Delay,5=Correlation			0034 00 00 05 0B 27 82 98 99 86 50 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-404656
11/05/2020, 12:53	MSS	MSPCE	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	5=Correlation,2=Delay			0034 00 00 05 0B 27 82 98 99 87 50 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-404655
09/05/2020, 15:25	MSS	MSPRL	ANALYSIS_FOR_NETWORK_GENERATED_NBR_OR_FOR_CHA_CASE_IS_MISSING	5=Correlation,2=Delay			0034 00 00 05 0B 27 82 98 99 89 20 00 00 00 00 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-373024
19/04/2020, 11:42	MSS	MSMTR	CALL_CONTROL_ANALYSIS_MISSING	2=Delay,3=Threshold			02 0001 00 0D 04 58 22 75 93 73 D0 00 00 00 00	ManagedElement=MSC-396216
11/05/2020, 14:06	MSS	MSPRL	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 00 2D 2F AA 00 00 00	ManagedElement=MSC-373024
12/05/2020, 09:56	MSS	MSPFS	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 00 31 27 A7 00 00 00	ManagedElement=MSC-372803
11/05/2020, 13:52	MSS	MSDNR	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 00 36 29 91 00 00 00	ManagedElement=MSC-372857
11/05/2020, 13:53	MSS	MSPCE	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 00 54 4A FF 00 00 00	ManagedElement=MSC-404655
11/05/2020, 19:51	MSS	MSDME	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 01 38 81 D1 00 00 00	ManagedElement=MSC-913029
11/05/2020, 13:44	MSS	MSBEO	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 01 A5 8B 4F 00 00 00	ManagedElement=MSC-913031
11/05/2020, 13:05	MSS	MSPSO	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 02 EC CE 41 00 00 00	ManagedElement=MSC-372984
12/05/2020, 06:52	MSS	MSCTN	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 07 DA 7A 64 00 00 00	ManagedElement=MSC-373001
11/05/2020, 15:54	MSS	MSDTJ	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 0C 4F AE 07 00 00 00	ManagedElement=MSC-404656
11/05/2020, 13:48	MSS	MSPPE	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 17 7F E2 01 00 00 00	ManagedElement=MSC-373025
12/05/2020, 10:08	MSS	MSJFR	CALL_ESTABLISHMENT_FAILURE	2=Delay,3=Threshold			02 02 00 02 05 A1 00 00 1F AF C0 3D 00 00 00	ManagedElement=MSC-397824
29/04/2020, 18:49	MSS	MSLKK01	CALL_ESTABLISHMENT_FAILURE_IN_USER_PLANE_CONTROL	3=Threshold			02 03 00 00 00 01 00 00 00 00 00 00 00 00 00	ManagedElement=MSC-326915

1	5	28	153	176	268	All Events (631)

Figure 4.3: Network monitoring view

The expectation was that the network monitoring team was able to translate this information into readable format using various tools, for example, initiating man-machine-language (MML) session commands. Extracting unique information required a specific alarm (in all cases in the figure, rule set automation is in play). The switching between tools to perform such a basic task was time consuming and the translation of data was a complex process, which explained the time taken to execute being long (section 5.3.1.2) as mentioned in the artefact evaluation chapter. The organisation did not have the capacity to establish alarm patterns in a way that would assist the user to understand the relation of events.

In the context of the study, the analysis performed as part of the investigation using alarm data has led to the following objectives as stated in Chapter One:

i) To propose a data analytics solution for reducing manual network analysis
ii) To propose a near real-time management solution for incident management
iii) To propose an automated smart grouping artefact for alarm patterns

4.1.3 Incident management

Each fault that has an impact on the service needs to be recorded, assigned for investigation and followed up until resolution, which is done through incident management. At any given point, there are multiple open incidents with different priority levels that need to be managed. Figure 4.4 shows as example 486 open incidents. The entries are presented in a list that is not quantified in a way that empowers the monitoring team to understand the overall state of the network clearly.

DSR was applied in the study and demonstrated that the trigger point was the problem identification, unlike traditional research methods that start with the researcher presenting the problem. The users were key stakeholders in identifying the problem, and the artefact that solved the problem is described in detail in this chapter.

Incidents

Create | View | Print | Process Overview | Service Context | Quick Actions | Bulk Close

1-100 | Showing 1 - 100 of 486

ID	Submit Date	CI	Priority	Service	CI Short Descr...	Tx Hub Site	Summary	Status
INC000017542624	5/12/2020 8:42:44 AM	prsesxia009	High	Service (Network)	n/a		TEMPERATURE_SUPPLY_AIR_AHU_1_DXX_RM_1	In Progress
INC000017542514	5/12/2020 8:34:46 AM	mtpss01001-nvi	High	Service (Network)	mtpss01001-nvi		HOST_IN_A_CLUSTER_THAT_DOES_NOT_HAVE_FU	In Progress
INC000017542608	5/12/2020 8:23:13 AM	ISD.C3D	Medium		ISD C3D Service		CONNECTIVITYADS TECHNICALCANNOT CONNECT.	In Progress
INC000017542442	5/12/2020 8:19:35 AM	MTA_MOne_I2000	High	Service (Network)	No Description sp		SERVICE_PROCESS_STATUS_ABNORMAL	In Progress
INC000017542491	5/12/2020 8:08:07 AM	pgiar27zamrcc	Critical	Service (Network)	pgiar27zamrcc.vo		SERVICE_STOPPED	In Progress
INC000017542419	5/12/2020 7:55:34 AM	MTA_MOne_I2000	High	Service (Network)	No Description sp		SHARE_DIRECTORY_UNUSABLE	In Progress
INC000017542417	5/12/2020 7:54:15 AM	PRS.NFV-2.C2960XR-1	High	Service (Network)	PRS.NFV-2.C296		CPU_UTILIZATION	In Progress
INC000017542411	5/12/2020 7:51:26 AM	MTA_MOne_I2000	Critical	Service (Network)	No Description sp		HIGH_USAGE_OF_CPU	In Progress
INC000017542271	5/12/2020 7:37:56 AM	ER	Low	ER	n/a		VC_ER_BUSINESS_TRANSACTION_HEALTH_BROKE	In Progress
INC000017542469	5/12/2020 7:36:47 AM	GGDN01	High	Service (Network)	GGDN01		CPU_UTILIZATION	In Progress
INC000017542362	5/12/2020 7:28:16 AM	MDP.NFV-1.C2960XR-1	High	Service (Network)	MDP.NFV-1.C296		CPU_UTILIZATION	In Progress
INC000017542209	5/12/2020 7:20:15 AM	MDP.NFV-2.C2960XR-2	High	Service (Network)	MDP.NFV-2.C296		CPU_UTILIZATION	In Progress
INC000017542148	5/12/2020 7:18:15 AM	PRS	High	Service (Network)	PRS (Rosslyn1 M		CPU_UTILIZATION	In Progress
INC000017542144	5/12/2020 7:14:15 AM	MDP.NFV-2.C2960XR-1	High	Service (Network)	MDP.NFV-2.C296		CPU_UTILIZATION	In Progress
INC000017542200	5/12/2020 7:10:45 AM	MDP.NFV-1.C2960XR-2	High	Service (Network)	MDP.NFV-1.C296		CPU_UTILIZATION	In Progress
INC000017542201	5/12/2020 7:10:56 AM	MDP	High	Service (Network)	MDP Midrand Dat		CPU_UTILIZATION	In Progress
INC000017542137	5/12/2020 7:08:15 AM	PRS.NFV-2.C2960XR-2	High	Service (Network)	PRS.NFV-2.C296		CPU_UTILIZATION	In Progress
INC000017542198	5/12/2020 7:08:05 AM	PRS	High	Service (Network)	PRS (Rosslyn1 M		CPU_UTILIZATION	In Progress
INC000017542001	5/12/2020 6:22:06 AM	NJG	Critical	Service (Network)	Johannesburg Ge		SCTPASSOCIATIONFAILURE	In Progress
INC000017542162	5/12/2020 5:42:54 AM	ONENDS_ALL	Critical	Service (Network)	n/a		SOFTWARE_FAULT	In Progress
INC000017541652	5/12/2020 5:14:46 AM	dnefscrx300pim01	Critical	Service (Network)	dnefscrx300pim01		FILESYSTEM_HAS_BEEN_UNMOUNTED	In Progress
INC000017541808	5/12/2020 5:07:25 AM	mtpvoam01	High	Service (Network)	mtpvoam01.ims.v		9250570_FAULT_IN_TSP_(141-9994)	In Progress
INC000017541799	5/12/2020 4:57:25 AM	CAT1_MGW_MGJMS_E(	Critical	Service (Network)	No Description sp		RECTB_SINGLE_MOD	In Progress
INC000017541449	5/12/2020 4:54:15 AM	MGMDN	High	Service (Network)	MGMDN		SDH_PROTECTION_SWITCHING_FAILED	In Progress
INC000017540335	5/12/2020 4:38:48 AM	Automation CI	Low	Automation Servic	To be used for aut		FW: Summer error	In Progress

Figure 4.4: Incident management view

4.1.4 Conclusion

At any given point, multiple events are happening on the network; some are planned and others are unplanned, and these events generate many alarms for the NOC to deal with. Rule-based automation was in place and did not result in the efficiency needed to manage events. Furthermore, the benefits brought by rule set automations did not have capability to unpack alarm parameters to the most granular level. The network monitoring team was inundated with repetitive manual tasks in the analysis of the network, and because of workload, in many cases the monitoring team did not execute them consistently. The incident management view did not quantify information in order to present information in a manner easily consumable by a human user. The study objectives, as aligned to the research problem, were to implement an information system that automates manual functions, and this was successfully done.

The flow of the chapter is illustrated in Figure 4.5. It important to understand that the problem identification was done within the research process, which is what makes design science research unique. An advanced level of machine learning forms part of the artefact. The researcher involved users extensively in the design process to ensure alignment of the problem and solution.

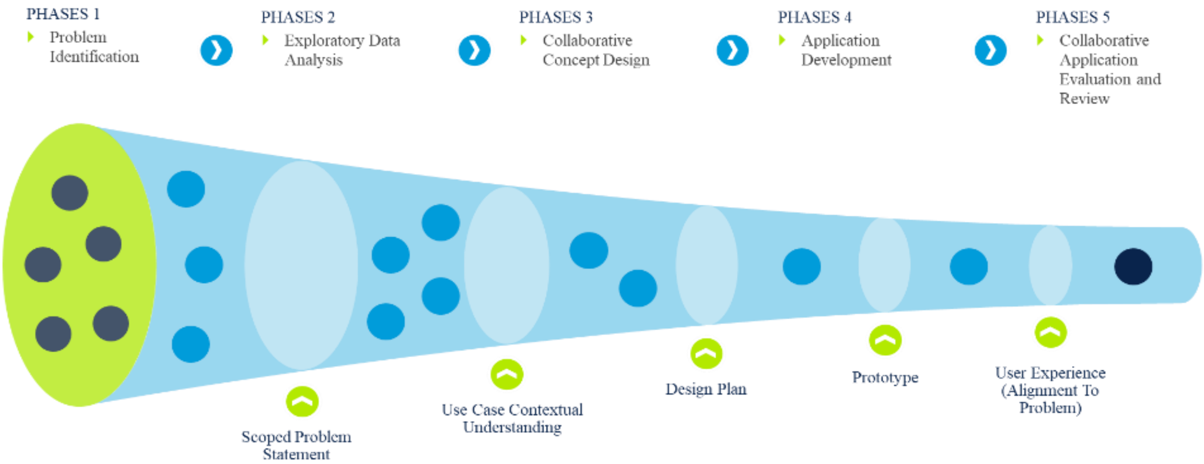


Figure 4.5: Chapter flow

The artefact construction phase posed a high level of complexity to the researcher as it involved engagement with people (users), learning the environment, preparing a software development environment, and also the design and development of the artefact. The research area is highly specialised, which led the researcher to conducting numerous interactions with the users in order to understand the context. The high volumes of data were analysed and domain knowledge was sourced. Data science practices were applied, forming a good IS basis.

The problem was identified and divided into three categories:

i) Too much effort on the part of users in manual network analysis
ii) Too much effort on the part of users in performing near real-time analysis of incident management data
iii) Lack of automated functionality for establishing alarm patterns

4.2 Requirements planning

4.2.1 Problem identification

The researcher held a session with the users, conveying the purpose of the interaction with the team. It was explained that the researcher was conducting an academic study towards a Master's degree in Business Information Systems, with the objective of identifying a practical problem and working towards solving an IS problem. The requirements planning process is illustrated in Figure 4.6. The users were from an electrical engineering and information technology background, which made it easier for them to understand the IS field. A period of two weeks was dedicated by the researcher to understand how users performed their functions. A follow up session was held in the form of a workshop, where users shared their operational pain points. The first level of problem identification was not limited to any specific category. The grouping of pain points was done in the workshop with the input of subject matter experts (SMEs).

The groups were as follows:

i) **Human resources:** Shortage of employees as well as a gap in skills development
ii) **Process gaps:** The lack of operational level agreements with some technical support teams
iii) **Technological gaps:** Lack of information system to assist with consumption of high data volumes generated by the network

The technological gaps identified presented a good opportunity for the researcher, for the following reasons:

i) Adoption of a good IS solution has the potential to solve possible human resourcing and process challenges since the application could reduce manual work done by people.
ii) The study has relevance to IS – working on any problem that was not IS relevant would have resulted in the study being disqualified by the university's information Systems department.

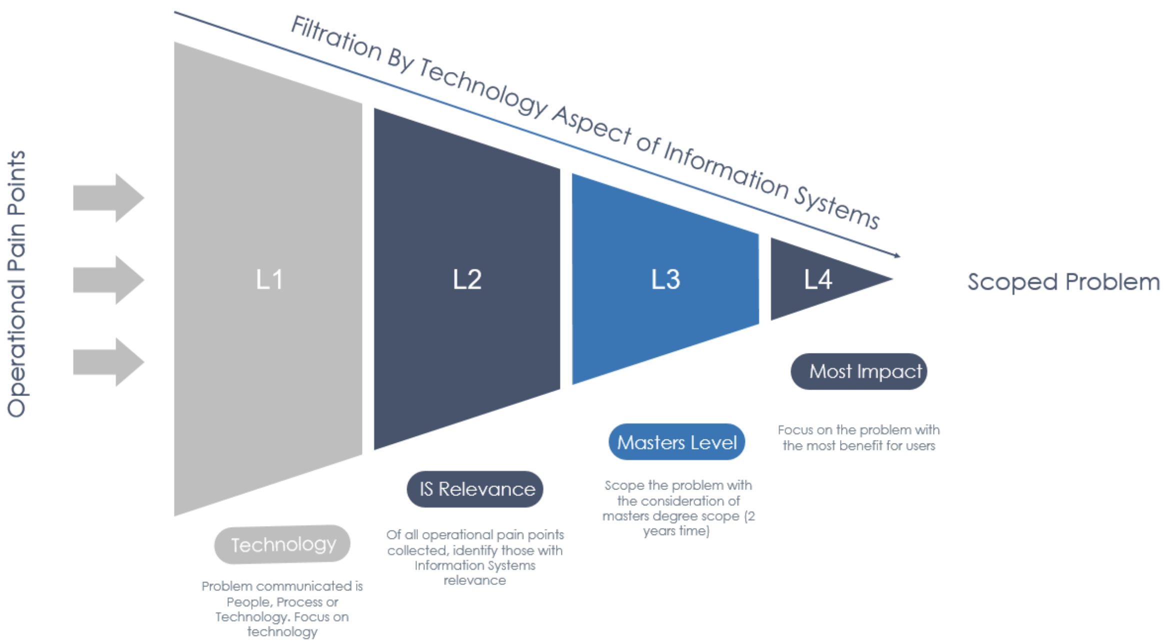


Figure 4.6: Requirements planning

The translation of the inputs from users using the fishbone tool (Figure 4.7) enabled the researcher to understand who or what the main contributors to the reduced efficiency in the NOC were. Event and incident management components are closely related to each other, as alarms indicate an event, and the incident management is used to track the fault. Figure 4.7 f**Error! Reference source not found.**ocuses on the refined problem statement.

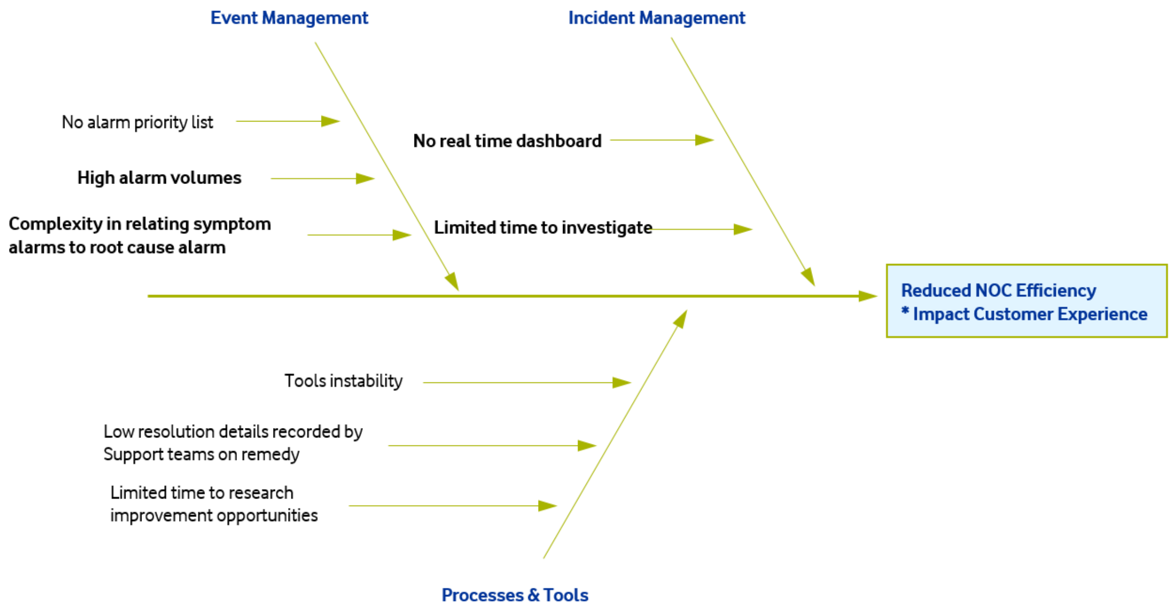


Figure 4.7: Problem contributors

It was established that extended network faults are caused by NOC engineers working manually to perform an investigation. The whole operation is reactive since investigations are done manually. It is not practical to analyse the pattern of alarms manually. The impact of this challenge is the prolonging of network outages since network investigations are based on analysing alarms to determine what is faulty on the network.

4.2.2 Proposed solution

The artefact was created with the consumption of the event data and incident data with the use of data analytics. The solution included highlighting the manual, inconsistent methods that were used to analyse the patterns in the incident, and event data. The artefact's main focus is indicated as improving efficiency in the NOC. The solution (Figure 4.8) should enable users to logon to the application and view information in a simplified manner that gives insights into currently active and cleared network faults with a few clicks.

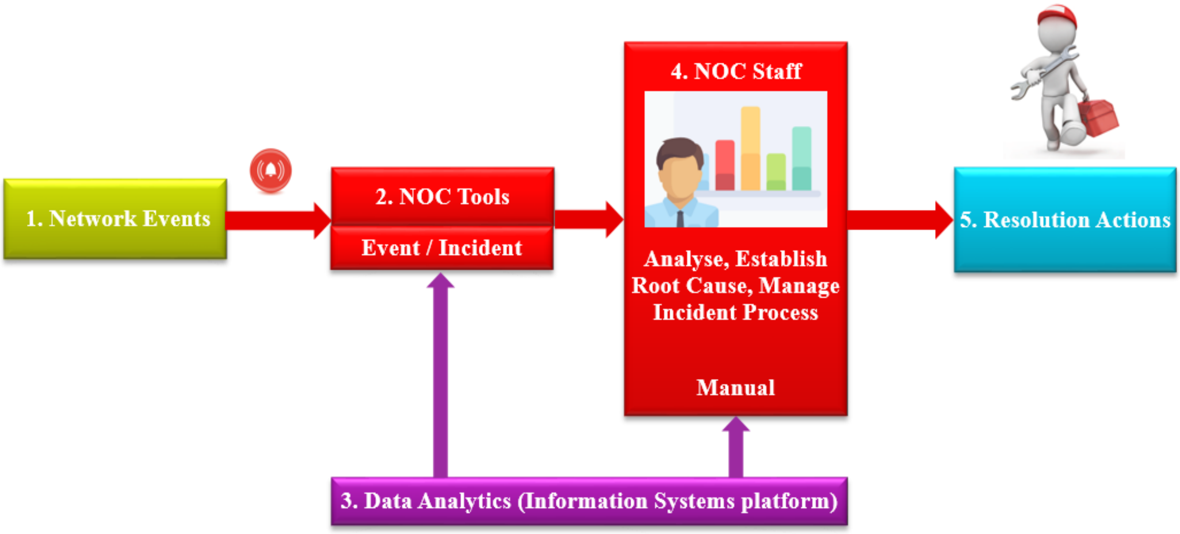


Figure 4.8: Proposed solution

It is important that the solution works seamlessly together with existing processes, and that the model (Figure 4.9) presents a holistic view of main processes in the area being studied. Components that are part of the solution are marked with an asterisk (*).

Support activities are the underlying resources that enable the service to be managed. A monitoring toolset involves event and incident management tools. The *Management Information Systems – Data Analytics* (Figure 4.9) is the layer introduced by this study to unlocking the potential to draw insights from the monitoring data. The human resources and business strategies cover the availability of people to get the work done in alignment with the goals of the business. The lack of any of the support activities will result in a low-quality service.

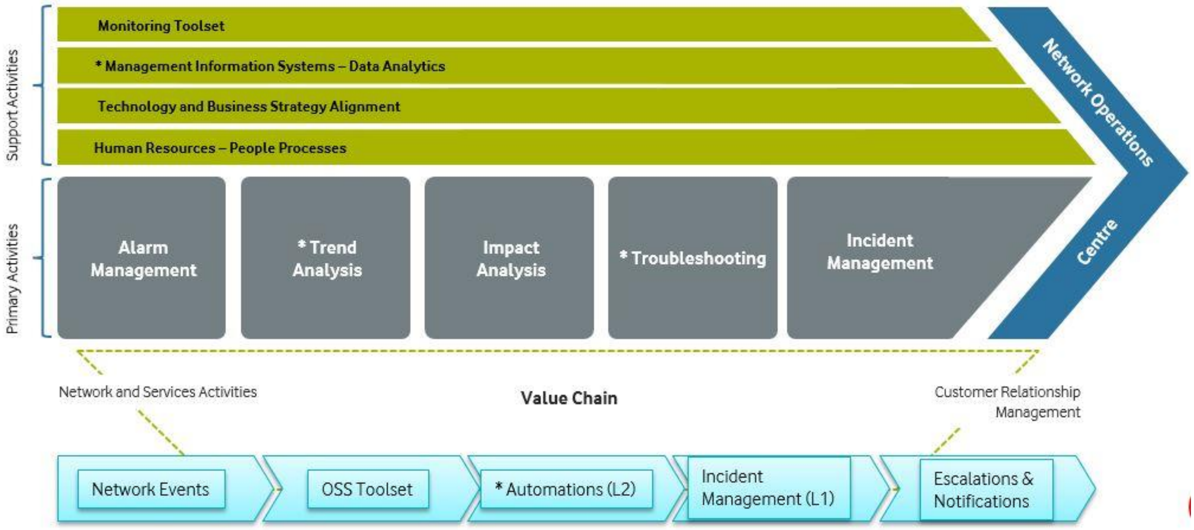


Figure 4.9: Framework aligning solution to operation

Primary activities (Figure 4.9) are a speciality of the NOC. As much as preventative measures help to avoid network faults, it is not the case that all the faults can be avoided and, once network faults are active, the NOC executes these primary activities.

The proposed solution covers the following:

i) Data analytics solution for reducing manual network analysis
ii) Near real-time management solution for incident management
iii) Automated smart grouping artefact for alarm patterns with the use of machine learning

4.2.2.1 Programming language

There are many programming languages available and the researcher chose to use Python. One of the most important skills in a data scientist's toolbox is the understanding of a general-purpose programming language that empowers a data scientist to perform data cleaning, data control and data analysis. These requirements call for programming languages that are sufficiently simple to learn and yet powerful enough to achieve complex coding tasks. Two such programming languages for data science utilised in industry and the scholarly community are Python and R (Taori & Dasararaju, 2019).

Python has numerous scientific algorithms, with more than 600 extraordinary code contributors, a large number of packages, more than 100,000 repositories and a great many downloads every year (Virtanen et al., 2020). Python is suitable for programming on a small to large scale, which makes it relevant to programmers at different levels (Menczer et al., 2020).

Computing, in terms of doing mathematical calculations, is an expertise that humankind created more than a thousand years ago. However, programming is in its earliest stages, with a history that traverses a few decades. The two concepts are immensely far reaching and typically taught as discrete subjects in educational institutions around the world, particularly at the undergraduate level (Linge & Langtangen, 2020). Most universities and colleges expect undergraduate students to spend a significant amount of time on computer science on the off chance that they need to gain proficiency in the specialty of programming since other study disciplines, for the most part, do not offer programming to the extent demanded for obtaining quality skills.

4.2.2.2 Integrated Development Environment

Jupyter Notebook was used in the development of the artefact. Jupyter Notebook supports more than 40 programming languages, and notebooks are intended to naturally develop and run codes (Le et al., 2020). P'erez and Granger (2018) define Jupyter Notebook as a web application that allows one to create and share documents that contain live code, equations, visualisations and explanatory text. The Jupyter interface is straightforward to use. When one is acquainted with the functionality of the interface, one can begin coding inside a note pad, using an internet browser. Jupyter Notebook is ideal for programmers who pride themselves in working efficiently (P'erez & Granger, 2018). A notebook comprises a progression of cells that each contains lines of code. Every cell can be run or rerun freely, provided that the rationale of the programming is regarded. The code can be effectively altered; for example, changing an algorithm in a cell without affecting the remainder of the clients (Le et al., 2020).

4.2.2.3 Unit testing

The code was divided into Jupyter Notebook cells with a print of the expected code output. Each piece of code (individual cell) had built-in testing to enable the developer to track how each of the pieces of code functioned. Failure to execute at each level was easily identified. This method of programming is very useful when software is handed to a different developer. The handing over of program to a different developer is common practice, as there is normally one team involved in program development and a different team in operations (maintenance).

4.2.3 Software artefact project plan

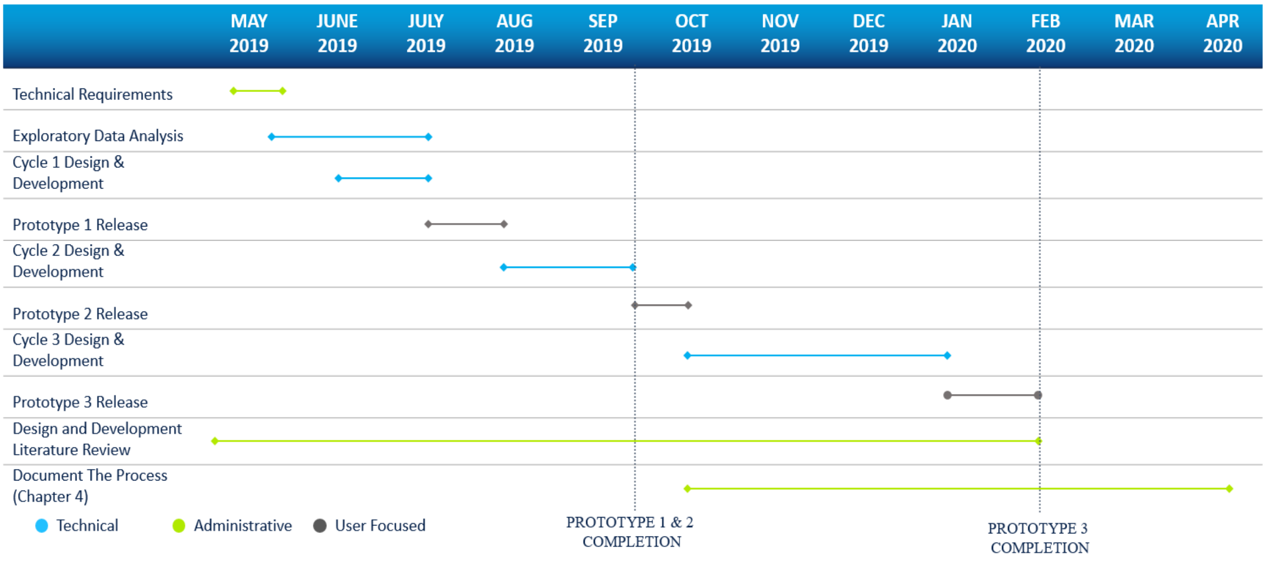


Figure 4.10a: Project plan

The software artefact project plan commenced on May 2019 and ended in April 2020, with an incremental model of delivery. Users enjoyed the value of Prototype 1 and Prototype 2 from October 2019, and Prototype 3 from March 2020 (Figure 4.10a). The development was structured in three distinct cycles (Figure 4.10b)

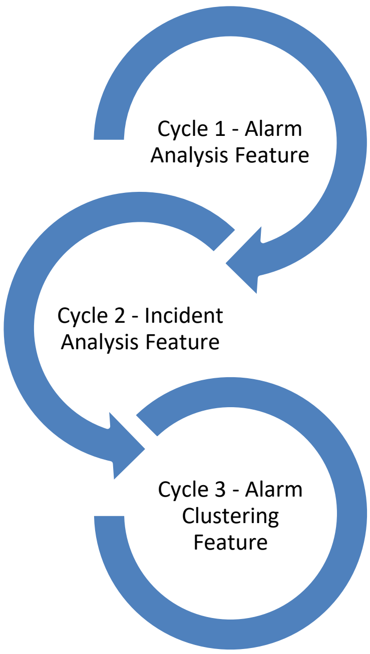


Figure 4.10b: Development Cycles

4.2.4 Reflection and revision

The problem identification process did prove to be complicated since, at the initial stage, different team members shared input with an individual understanding that each of their experiences was most relevant and meaningful. The collective work in grouping shared problems helped, as team members showed positives signs of being included.

4.3 Cycle 1 – Iteration 0

Figure 4.10c indicates the current stage which Cycle 1.

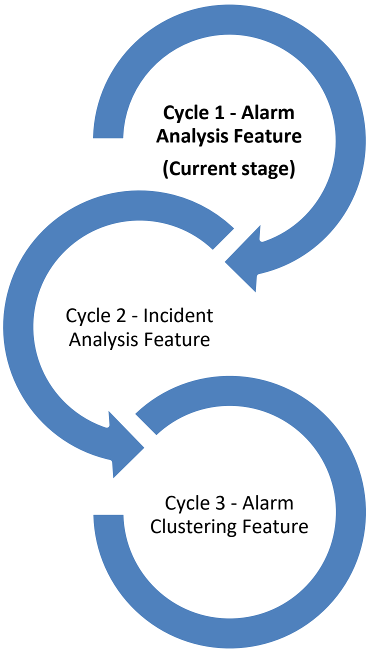


Figure 4.10c: Development Cycle 1

4.3.1 Iteration objective

The users conveyed an operational gap, namely, the difficulty to assess the pattern of alarm flows in order to enable them to pinpoint areas of concern on the network that requires investigation. The way network engineers work is by assessing alarms as they flow in, and attend to them accordingly. The only way to get a clear understanding of the alarm patterns is by manually exporting alarms from the front-end application of the Netcool monitoring tool into an Excel spreadsheet and then group information to create plots. That takes a long time and does not provide any perspective of alarms that occurred prior to the data export. The exercise was lengthy and it was not practical to perform on a frequent basis during a shift. This pain point led to the network engineers neglecting to perform an overall network status analysis.

Two network malfunction conditions were specified. These two conditions occur frequently and are caused for a variety of reasons. The network engineers had to manually login to the network elements generating these alarms, extract the failure reason code in numerical format and then translate this code into the description. This process took very long and, as a result, these conditions were ignored.

4.3.2 Data exploration

The user requirements' dependence on alarm data led the researcher to study the available data. It was established that alarm data were stored in a SQL server. The reliability and quality of the data were at very high levels, as the organisation had a regulatory requirement for maintaining this data. The database was confirmed to be one of the critical parts of the NOC and access was provided to a limited number of engineers. The researcher was granted access to the database and started to access the data schema, an exercise that took a month to complete. The table hosting information that related to user requirements was identified. The format of alarm data was consistent; however, the table had 290 columns and domain knowledge was sourced. A Level 2 (L2) NOC engineer was consulted and he assisted in the process of eliminating unnecessary columns. The L2 engineer further assisted by describing information contained in each of the columns. The researcher wrote SQL queries to extract the data using Oracle SQL. The query outputs were satisfactory, meaning the researcher had a good understanding of the data.

4.3.3 Collaborative concept design

A high-level concept was put in place. This concept was meant to formulate a structure (a kind of mind map) for the designer. This concept is very important as it maps the organisation requirement to the expected product in order to serve the customer's needs. It had already been established from discussions with the users that the artefact design and development requirements were dependent on the alarm data. The expected results, from the user's point of view, were a tool able to perform the unnecessary manual exercise of data extraction, cleaning and visualisation. The concept (Figure 4.11Figure 4.11) refers to the design concept, which includes the alarms, a web application and a regular refresh.

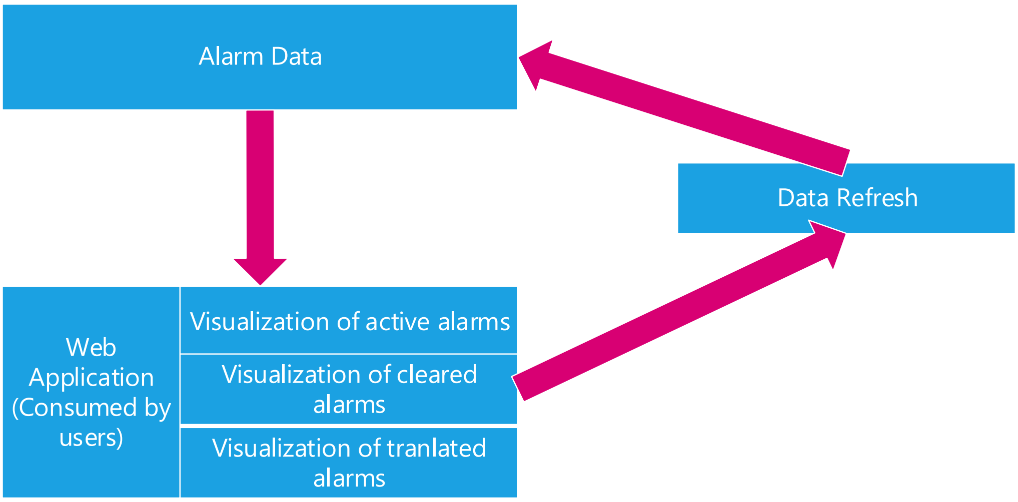


Figure 4.11: Concept design

4.3.4 Design

The user requirements were understood, the concept was in place, and the data were understood by the researcher, leading to the next action, which was creating a detailed plan of how the artefact would be created. The process involved training the available resources in the organisation that could be used with little to no cost implications, as the researcher did not intend to invest much funding into the project. Using what is already available is essential in software engineering since it prevents organisations from buying tools or licenses for new systems when there are tools available to accomplish the same or better results. The following resources were available:

- Data source (SQL server)
- Windows 16 server
- Connectivity via the Intranet of the organisation
- Access to open source development tools

4.3.4.1 High Level Design

Data were sourced from the alarm database and transferred to the Windows server that hosted the artefact. The original state of the data was not conducive to be processed, and for that, a data preparation excise was performed (Figure 4.12Figure 4.12). The preparation ensured that only the data needed to address the identified problems were processed. The prepared data were transformed into user consumable information. The user interface was created, enabling access though a web interface.

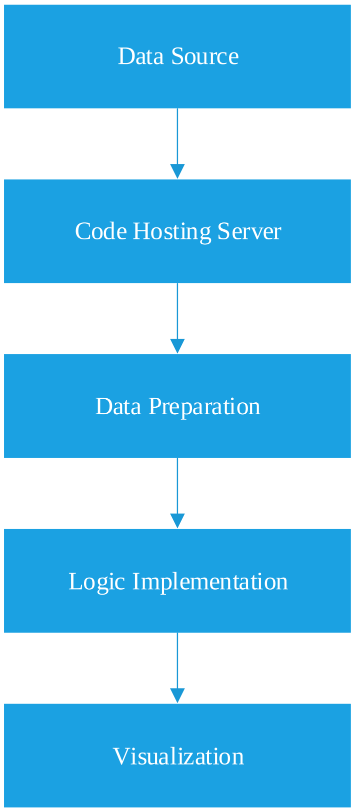


Figure 4.12: High level design

4.3.4.2 Low Level Design

A more detailed version (Figure 4.13) of what the researcher did in planning for the artefact creation is set out as follows:

i) Connection from the code hosting server to the alarm DB was established. This was required to ensure that alarm data could be accessed in the server where the development was done.

ii) A data extraction query through SQL was written to only source the data relevant to the problem being addressed. This step was very important for the performance of the application as sourcing high volumes of unnecessary data had a big impact in terms of the poor performance of the application.

iii) In order to keep the data available to users updated, an update mechanism was put in place. This was a routine code to trigger the SQL query.

iv) A data frame was created from data sourced through a data extraction query.

v) The data frame was further refined to drop some bits of data that were not needed. This was essential for performance improvement.

vi) A module was created for network configuration lookup and the translation to readable names.

vii) A module was created for dictionaries to transform numerical data to readable text.

viii) The data from all modules were aggregated.

ix) Visuals (aligned to the outlined requirements by users) were created. The importance of this step was that it had to be evident that the artefact addressed the communicated pain points identified by users.

x) All created visuals were inserted into the application (user interface) and the UI was published.

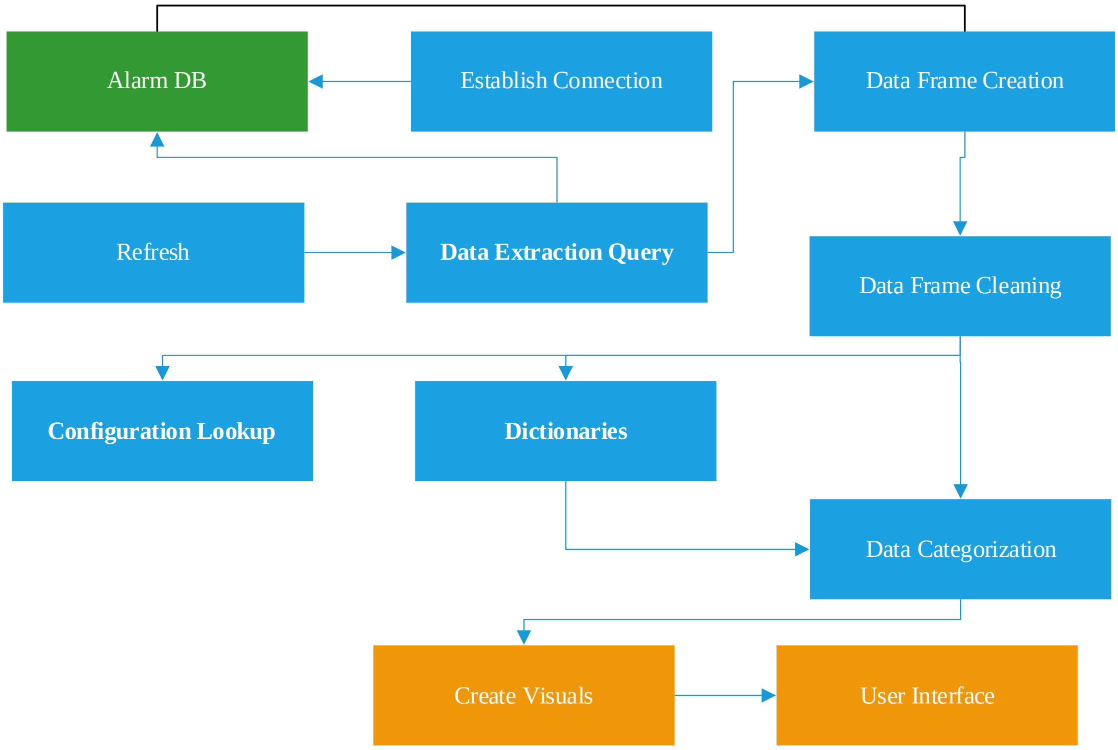


Figure 4.13: Low level design

4.3.5 Application development

A Cx_Oracle module was used to connect to the data source (Figure 4.14). Logon credentials were managed in a separate file (called cred) for security reasons.

Login to the server

```
ip = "10.xx.xx.xx" #xx added to mask actual IP address
port = 1521
SID = "oss"
dsn_tns = cx_Oracle.makedsn(ip, port, SID)

f= open("cred")
# print(f.readline(),f.readline())
db = cx_Oracle.connect(f.readline().strip(),f.readline().strip(), dsn_tns)
f.close()
#cursor = db.cursor()
```

Figure 4.14: Connection establishment

This code established connection between the code-hosting server and the alarm database. Two **select** statements were compiled to extract the alarm data. The collected data were divided into two network element types. Two data frames (omgwraw & mssraw) were created from the extracted data.

```
def shorten(x):
    """Shortens Supp_info names for plotting"""
    if x == None or x == 'None':
        return "/none"
    import re
    dontinclude = ['fsClusterId','fshaRecoveryUnitName','fsipHostName','fsFragmentId',
    l = re.split('=|,',x)
    l = [el.strip() for el in l if el not in dontinclude]
#     print(l)
    return ";".join(l)
```

Figure 4.15: Data preparation

Some text in columns (Figure 4.15) to be consumed later had long strings that had to be cut as domain knowledge sourced from the SME confirmed that the benefits were not to be affected. Failure to shorten the text impacted on data visualisation.

```
mssalarmsactive = mssraw[mssraw["ALARM_STATUS"] == 1]
mssalarmsclear = mssraw[mssraw["ALARM_STATUS"] == 0]
omgwalarmsactive = omgwraw[omgwraw["ALARM_STATUS"] == 1]
omgwalarmsclear = omgwraw[omgwraw["ALARM_STATUS"] == 0]
cncedf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2745]
cefiupdf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 3263]
cefdf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2668]
aiaedf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2194]
rsudf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2064]

# omgwalarmsclear.SUPPLEMENTARY_INFO = str(omgwalarmsclear.SUPPLEMENTARY_INFO)
omgwalarmsclear.SUPPLEMENTARY_INFO = omgwalarmsclear.SUPPLEMENTARY_INFO.apply(shorten)
omgwalarmsactive.SUPPLEMENTARY_INFO = omgwalarmsclear.SUPPLEMENTARY_INFO.apply(shorten)
```

Figure 4.16: Data preparation

Data frames (DFs) were further refined for performance reasons (Figure 4.16). The idea was that each of the DFs was consumed multiple times within the program, ensuring that the separation allowed parallel executions. Furthermore, in cases where the data structure changed for one of the cases, not all figures would be impacted. Thus, the debugging has been simplified.

```
def extract(s):
    return s.split()

def extract_code_from_list_as_string(l, index):
    if index >= len(l):
        return 'NONE'
    else:
        return l[index].strip()

def first_col_dict(x):
    dictionary = {'03' : "Cell Identity (CI)",
                  '04' : "Location Area Code (LAC)",
                  '05' : "Location Area Code and Cell Identity (LAC + CI)",
                  '06' : "Service Information Octet and Signalling Point Code (SIO+SPC)",
                  '08': 'Cell global Identity (CGI), (includes: MCC,MNC,LAC,CI)',
                  '09': 'Location Area Identity (LAI), (includes: MCC,MNC,LAC)',
                '0A': 'Service Area Identity (SAI), (includes: MCC,MNC,LAC,SAC)',
                '0B': 'Global RNC Identity, (includes: MCC,MNC,RNC_ID)',
                '0C': 'RNC identity',
                '0D': 'Service Area Code (SAC)',
                '0E': 'Multimedia gateway (MGW(Rel99))'
    }
    return dictionary[x]

def seventh_col_dict(x):
    dictionary = {  '00' :  "unknown",
                    '01' :  "supplementary service",
                    '02' :  "mobile-originated call",
                    '03' :  "IMSI detach",
                    '04' :  "location update",
                    '05' :  "mobile-originated short message",
                    '06' :  "mobile-terminated short message",
                    '07' :  "mobile-terminated call",
                    '08' :  "emergency call",
                    '09' :  "handover",
                    '0A' :  "re-establishment of the connection",
                    '0B' :  "SCCP subsystem state change",
                    '0C' :  "SDCCH handover",
                    '0D' :  "handover in MSC-B",
                    '0E' :  "resource indication",
                    '0F' :  "handover candidate",
                    '10' :  "load announcement"

    }
    return dictionary[x]


def translate_cols(df):
    functions = {"Supplementary_0" : first_col_dict,
                 "Supplementary_7" : seventh_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = cncedf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    cncedf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))
cncedf = translate_cols(cncedf)
```

Figure 4.17: CNCEDF dictionary

A dictionary was created to translate hexadecimal codes to readable text (for the cellular network configuration error case). This function fulfilled the tasks that users were doing manually on an individual alarm basis, with the advantage that very high alarm volumes can be processed.

```
def extract(s):
    return s.split()

def extract_code_from_list_as_string(l, index):
    if index >= len(l):
        return 'NONE'
    else:
        return l[index].strip()

def second_col_dict(x):
    dictionary = {  '00' :  "international network 0 (IN0)",
                    '04' :  "international network 1 (IN1)",
                    '08' :  "national network 0 (NA0)",
                    '0C' :  "national network 1 (NA1)"

    }
    return dictionary[x]

def third_col_dict(x):
    dictionary = {  '00' :  "ordinary signalling point",
                    '03' :  " group of signalling points",
                    '04' :  "adjacent signalling point"

    }
    return dictionary[x]


def translate_cols(df):
    functions = {"Supplementary_1" : second_col_dict,
                 "Supplementary_2" : third_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = rsudf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    rsudf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))

rsudf = translate_cols(rsudf)
rsudf
```

Figure 4.18: RSUDF dictionary

A Dictionary was created (Figure 4.18) for the *Route Set Unavailable* case. Users received these alarms and had to interpret the causes from the hex codes. This dictionary solved the problem of the user having to figure out the cause and type of the signalling route fault.

A **select** query was written and embedded in the Python code to collect network configuration data. The challenge the researcher was solving was that the *Route Set Unavailable* case had a meaningful numerical address that was indicating the peer network element that the route failed towards. The users and the SME explained that this particular fault scenario had a big impact on the service and required the identification of impacted elements to be outlined as soon as possible.

```
pc = mssalarmsactive[mssalarmsactive['TEXT'] == 'ROUTE SET UNAVAILABLE'].sort_values('SUPPLEMENTARY_INFO')
pc["Point_code"] = pc['SUPPLEMENTARY_INFO'].apply(lambda x : x[4:8])
pc.sort_values('Point_code')
pc.drop('SUPPLEMENTARY_INFO',axis=1,inplace=True)
#alarms.set_index("TEXT",inplace = True)
pc
```

Figure 4.19: Signalling point code preparation

The code in Figure 4.19 above extracts the four characters that need a lookup against the configuration data.

```
alarm=pc.merge(RAN,left_on=["Point_code"],right_on=["DSP_SIG_PNT_CODE"]).drop(["DSP_SIG_PNT_CODE"],axis=1)
alarm
```

Figure 4.20: Lookup signalling point code names from configurations data

The code in Figure 4.20 fetches the four characters and performs a lookup from the confirmation data (sourced through the RAN data frame (DF) from the network configuration data). The human-readable name of the network element that triggered the alarm was appended to the alarm DF. This DF is an end product to be consumed for visualisation (for the *Route Set Unavailable Case*).

```
from bokeh.models.widgets import DataTable, TableColumn

def get_ROUTE_SET_UNAVAILABLE_ACTIVE(df):
    df2 = alarm.head(10)
    """Get a table with the latest Pointcode"""
    data = {"ALARM_TIME" : df2.ALARM_TIME,"TEXT" : df2.TEXT,
            "NODE" : df2.NODE,"Point_code" : df2.Point_code,
            "DSP_SIG_PNT_NAME" : df2.DSP_SIG_PNT_NAME}
    src = ColumnDataSource(data)

    columns = [
            TableColumn(field="ALARM_TIME", title="Alarm time"),
            TableColumn(field="TEXT", title="Text"),
            TableColumn(field="NODE", title="Node"),
            TableColumn(field="Point_code", title="Point code id"),
            TableColumn(field="DSP_SIG_PNT_NAME", title="Point code name")
        ]
    data_table = DataTable(source=src, columns=columns, width=300, height=500)
    panel = Panel(child=data_table,title="Unavailable Active")
    return panel

# show(get_ROUTE_SET_UNAVAILABLE_ACTIVE(alarm))
```

Figure 4.21: Table for RSU visualisation

The code above (Figure 4.21) is the Bokeh implementation for *Route Set Unavailable Case* visualisation.

The other case that proved to be time consuming for the users was the *Call Establishment Failure Scenario*. This fault is caused for a number of reasons, and translation, similar to the cellular network configuration error translation, was required and implemented as indicated in Figure 4.22.

```python
def first_col_dict(x):
    dictionary = {  '01' :  "opening of the file failed",
                    '02' :  "initialization of file pointer failed",
                    '03' :  "creation of hand failed",
                    '04' :  "setting of TNSDL-timer failed",
                    '05' :  "wrong answer received for a service request",
                    '06' :  "use of parameter file failed",
                    '07' :  "no free handover number",
                    '08' :  "no free roaming number",
                    '09' :  "the table of IMSI numbers is full",
                    '10' :  "the table of TMSI numbers is full"

    }
    return dictionary[x]


def second_col_dict(x):
    dictionary = {  '00' :  "undefined",
                    '01' :  "all",
                    '02' :  "MS",
                    '03' :  "PBX",
                    '04' :  "PSTN",
                    '05' :  "handover",
                    '06' :  "call forwarding",
                    '07' :  "call drop-back",
                    '08' :  "call transfer",
                    '09' :  "location update"

    }
    return dictionary[x]


def third_col_dict(x):
    dictionary = {  '00' :  "undefined",
                    '01' :  "all",
                    '02' :  "MS",
                    '03' :  "PBX",
                    '04' :  "PSTN",
                    '05' :  "handover",
                    '06' :  "call forwarding",
                    '07' :  "call drop-back",
                    '08' :  "call transfer",
                    '09' :  "location update"

    }
    return dictionary[x]
```

```python
def fourth_col_dict(x):
    dictionary = {  '01' :  "partly",
                    '02' :  "the whole computer unit in question",
                    '03' :  "the whole network element in question"

    }
    return dictionary[x]

def translate_cols(df):
    functions = {"Supplementary_0" : first_col_dict,
                 "Supplementary_1" : second_col_dict,
                 "Supplementary_2" : third_col_dict,
                 "Supplementary_3" : fourth_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = cefdf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    cefdf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))

cefdf = translate_cols(cefdf)
cefdf
```

Figure 4.22: CEFDF dictionary

The users asked the researcher to resolve a pain point by creating a solution that would allow them to assess alarm volumes in a graphical way. The main objective, from the users' perspective, was to be able to quantify records flowing to the network monitoring screen. The visualisation for this aspect was built from the well-constructed

DFs we started with. A function (Figure 4.23) to be consumed for multiple visualisations was created to avoid having to write a visualisation code for each of the alarm volume-based conditions.

```
def get_plot(df,title):
    """Generate and returns a plot"""

    temp={"Columns":df.index,
         "values": df.values}
    src=ColumnDataSource(temp)

    plot = figure(x_range=df.index.values, title=title,plot_height=800,plot_width=700,background_fill_color='white', l
    plot.xaxis.major_label_orientation = "vertical"
    plot.vbar(x="Columns", top = "values", width=0.8, source=src, fill_color='red', line_color='black')


    #Anotations
    labels = LabelSet(x="Columns", y="values", text="values",
                          x_offset=-8, y_offset=0, source=src,text_font_size="9pt", render_mode='canvas')
    plot.add_layout(labels)

    plot.xgrid.grid_line_color = None
    plot.y_range.start = 0
    return plot
```

Figure 4.23: Plotting function

At this stage, all the figures that needed to be visualised, had been prepared. The next step was to build an application that would allow the user to consume the information, and the solution had to be easy to navigate. First, tabs were created (Figure 4.23) and the panels were structured with the code below. The application structure required the figures to be inserted (Figure 4.24). At this point, the final product to be consumed by the user was in place, thereby enabling users to consume the data from a web interface.

```
# Application Structure
def make_tab(fig1,fig2,fig3,fig4,title):
    grid = gridplot([[fig1,fig2],[fig3,fig4]])
    tab = Panel(child=grid,title=title)
    return tab

def make_to_figs(fig1,fig2,title):
    grid = gridplot([[fig1,fig2]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab2(fig1,fig2,title):
    grid = gridplot([[fig1,fig2]])
    tab = Panel(child=grid,title=title)
    return tab
```

Figure 4.24: Application tabs structure

```
p1 = get_plot(mssalarmsactive["NODE"].value_counts().head(20),"MSS-Plot Top 20 Count Per MSS (Active)")
p2 = get_plot(mssalarmsactive["TEXT"].value_counts().head(20),"MSS-Plot Top 20 Count Per ALARM  (Active)")
p3 = get_plot(mssalarmsactive["SUPPLEMENTARY_INFO"].value_counts().head(20),"MSS-Plot Top 20 Count Per Supp Info (Activ
p4 = get_plot(mssalarmsactive["SEVERITY"].value_counts().head(20),"MSS-Plot Alarm Count Per SEVERITY (Active)")
tab1 = make_tab(p1,p2,p3,p4,"Active MSS")


p1 = get_plot(mssalarmsclear["NODE"].value_counts().head(20),"MSS-Plot Top 20 Count Per MSS (Clear)")
p2 = get_plot(mssalarmsclear["TEXT"].value_counts().head(20),"MSS-Plot Top 20 Count Per ALARM  (Clear)")
p3 = get_plot(mssalarmsclear["SUPPLEMENTARY_INFO"].value_counts().head(20),"MSS-Plot Top 20 Count Per Supp Info (Clear)
p4 = get_plot(mssalarmsclear["SEVERITY"].value_counts().head(20),"MSS-Plot Alarm Count Per SEVERITY (Clear)")
tab2 = make_tab(p1,p2,p3,p4,"Cleared MSS")

p1 = get_plot(omgwalarmsactive["NODE"].value_counts().head(5),"OMGW-Plot Top 5 Count Per OMGW (Active)")
p2 = get_plot(omgwalarmsactive["TEXT"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Alarm (Active)")
p3 = get_plot(omgwalarmsactive["SUPPLEMENTARY_INFO"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Supp Info (Activ
p4 = get_plot(omgwalarmsactive["SEVERITY"].value_counts().head(5),"OMGW-Plot Top Count Per Severity (Active)")
tab3 = make_tab(p1,p2,p3,p4,"Active OMGW")

p1 = get_plot(omgwalarmsclear["NODE"].value_counts().head(5),"OMGW-Plot Top 5 Alarm Count Per OMGW (Clear)")
p2 = get_plot(omgwalarmsclear["TEXT"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Per Alarm (Clear)")
p3 = get_plot(omgwalarmsclear["SUPPLEMENTARY_INFO"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Per Supp Info (Cl
p4 = get_plot(omgwalarmsclear["SEVERITY"].value_counts().head(5),"OMGW-Plot Top Count Per Severity  (Clear)")
tab4 = make_tab(p1,p2,p3,p4,"Cleared OMGW")


p5 = get_ROUTE_SET_UNAVAILABLE_CLEAR(alarm2)
p6 = get_ROUTE_SET_UNAVAILABLE_ACTIVE(alarm)
# tab5 =  make_to_figs(p5,p6,"Route set unavailable")
t5 = Tabs(tabs=[p5,p6])
tab5 = Panel(child=t5,title="Route Sets")


p7 = get_plot(cncedf["Supplementary_0"].value_counts().head(20),"CNCE Reasons")
p8 = get_plot(cncedf["Supplementary_7"].value_counts().head(20),"CNCE Transaction Type")
#t6 = Tabs(tabs=p7,p8)
tab6 = make_tab2(p7,p8,"CNCE")

dash = Tabs(tabs=([tab1,tab2,tab3,tab4,tab5,tab6]))

show(dash)
```

Figure 4.25: Application visualisation

4.3.6 Cycle 1 Prototype

An application with seven tabs populating a variety of visuals was developed, addressing user requirements. The tabs are based on focus areas as detailed below.

Case 1

Case 1 is a simplified alarm analysis for network faults (alarm) per network element, with the number of most frequent alarms in a descending format (Figure 4.35). This user interface display required no action from the user in terms of sourcing, sorting and visualising the data. The user just had to login to the application and all the key information needed by the NOC engineer to action was presented. The bottom two panels in Figure 4.26Figure 4.26 provide a breakdown of faults per severity and codes of unique causes (or supplementary information). This view was replicated for both the cleared faults and active faults for the two types of network elements (Figure 4.27), namely the Mobile Switching Server (MSS) and the Open Media Gateway (OMGW). Tabs one and two have been reserved for the MSS information, and tabs three and four for the OMGW information. This approach ensured that the user was not presented with combined data from two element types, as this would lead to complicated interpretations.

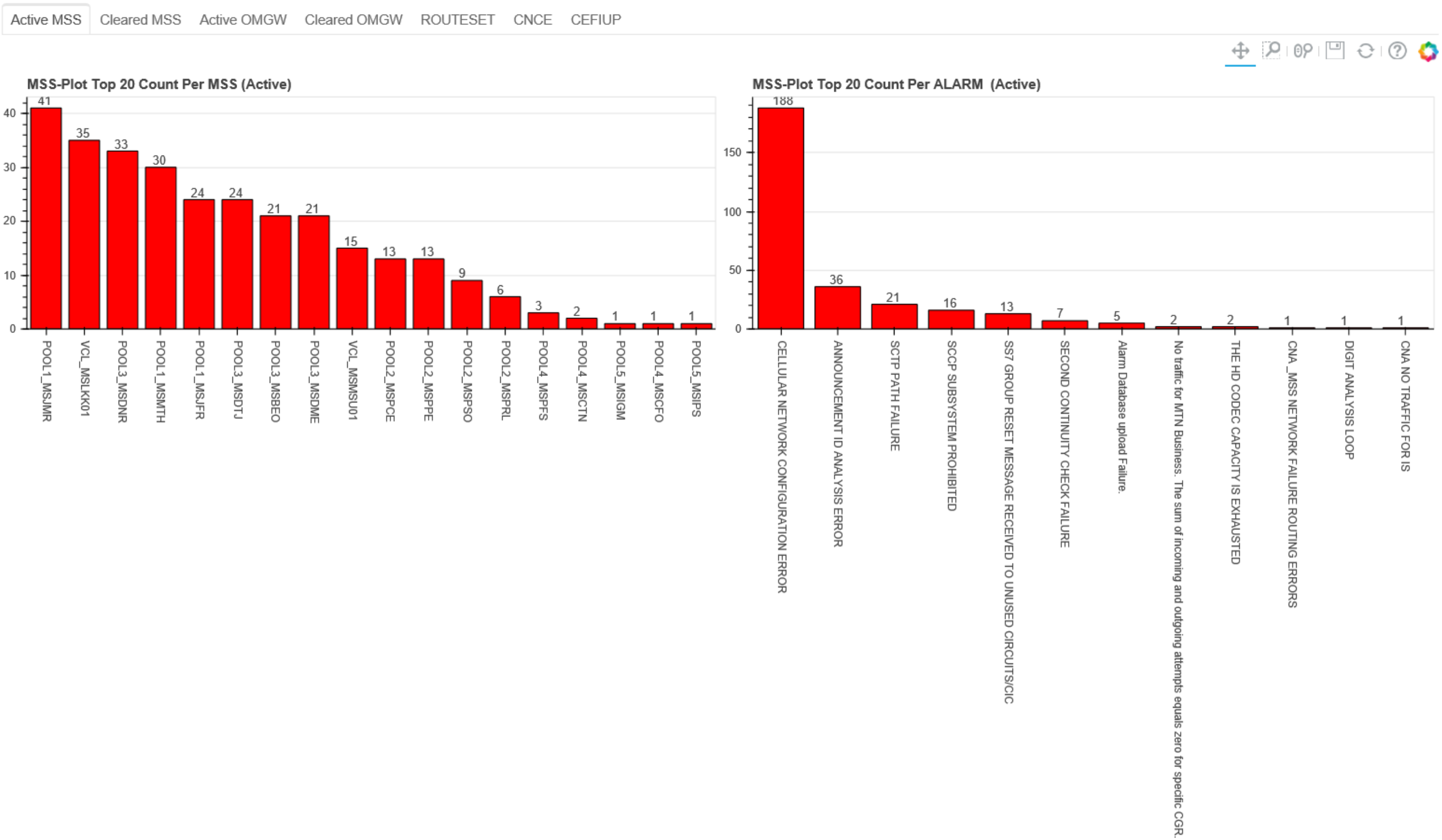


Figure 4.26: Cycle 1 – Prototype A

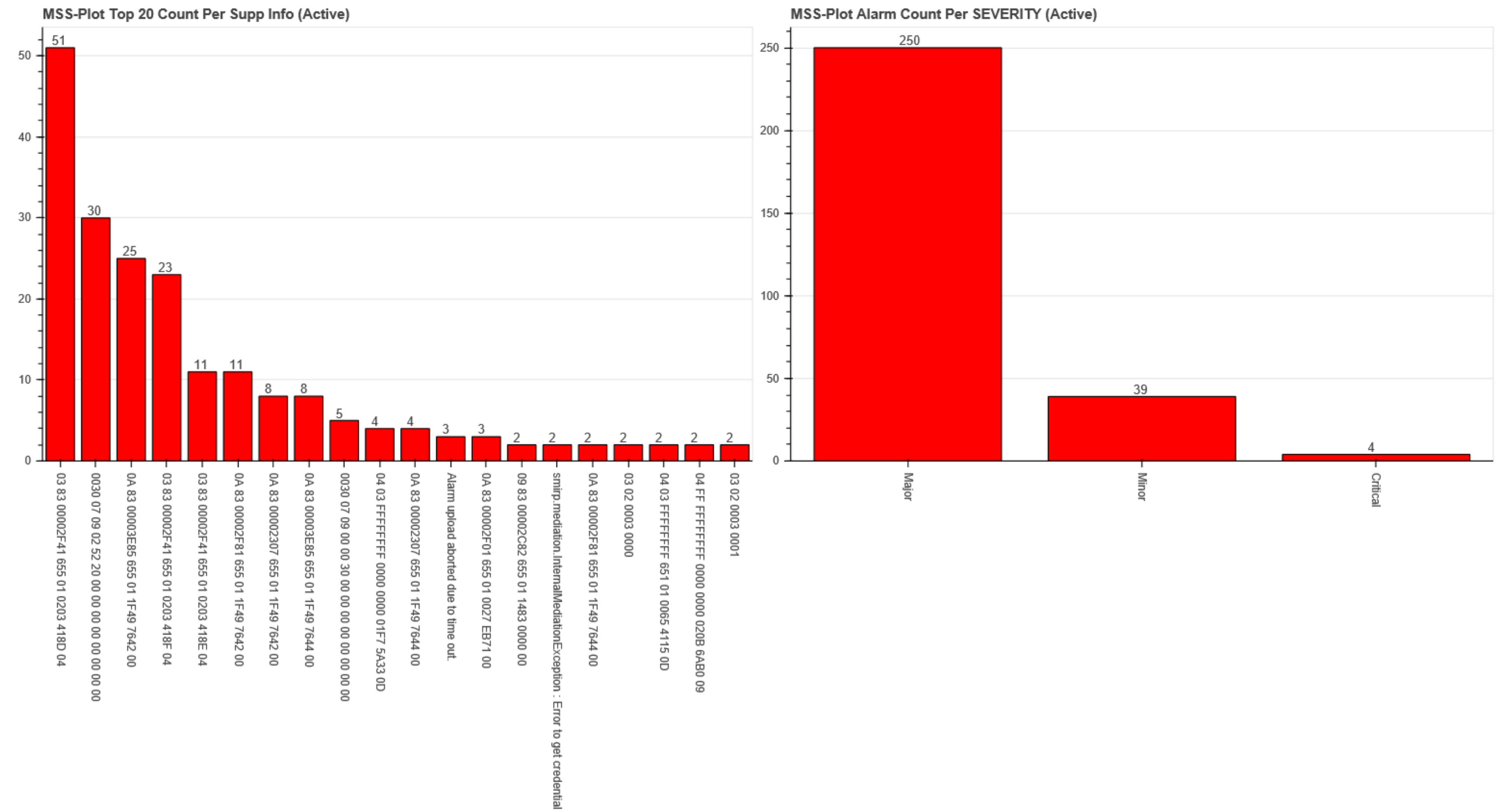


Figure 4.27: Cycle 1 – Prototype B

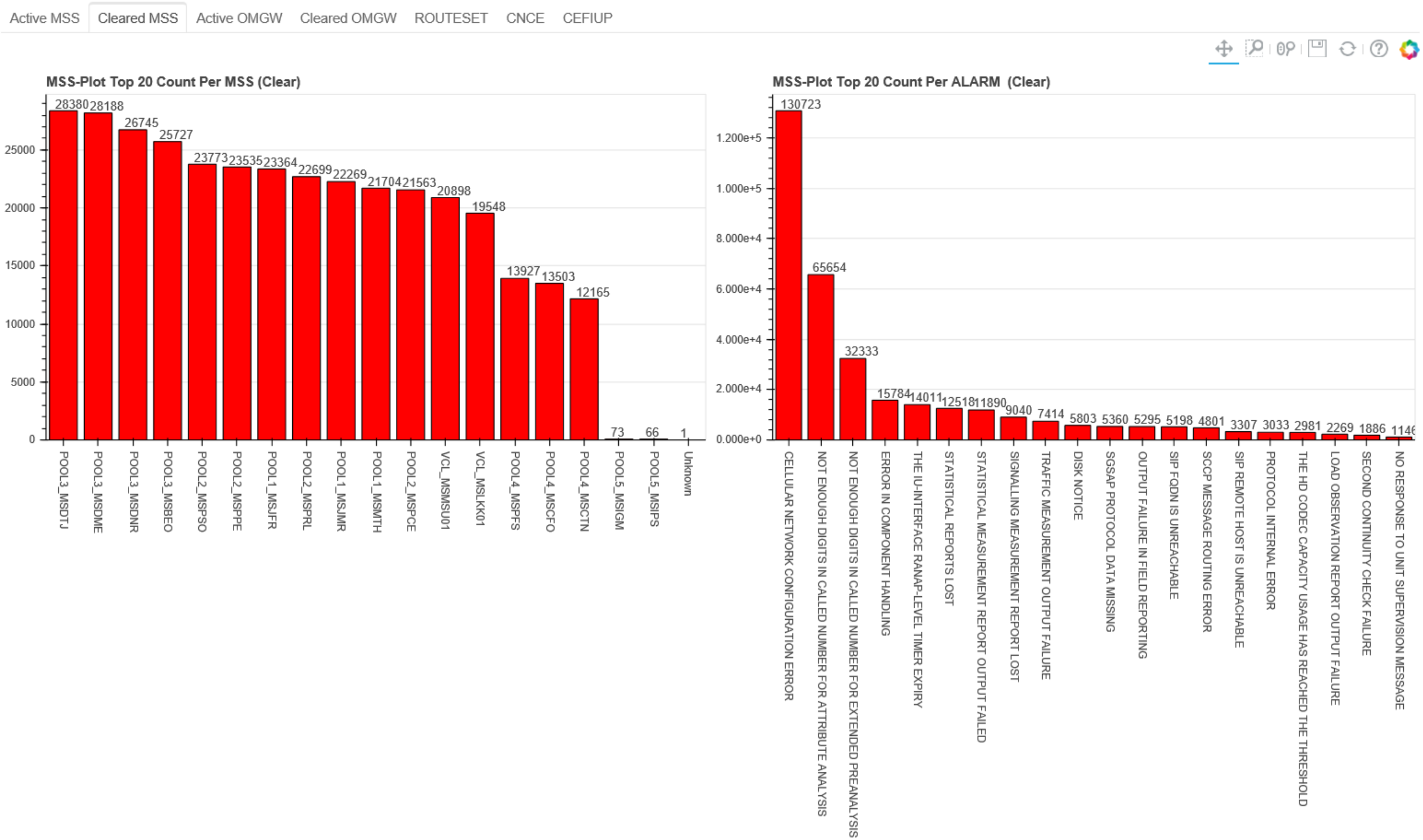


Figure 4.28: Cycle 1 – Prototype C

The representation of the cleared alarm view as per Figure 4.28 gives a better appreciation of the volumes of alarms being analysed, which was over 250,000 alarm transactions for MSSs in only a 24 hour period, as quantified in the exploratory data analysis. The displayed content changed based on the status of the network. As the shift progressed the content was refreshed, providing a very clear perspective of network patterns. Some conditions triggered for a very short period of time (less than 5 seconds) and the NOC engineer would not have been aware of them since they cleared before being presented on the network monitoring screen. This application enabled the users to detect these conditions and act accordingly. The application demonstrates a classic case of an information system adding direct value to a business by converting data into insights used for decision making.

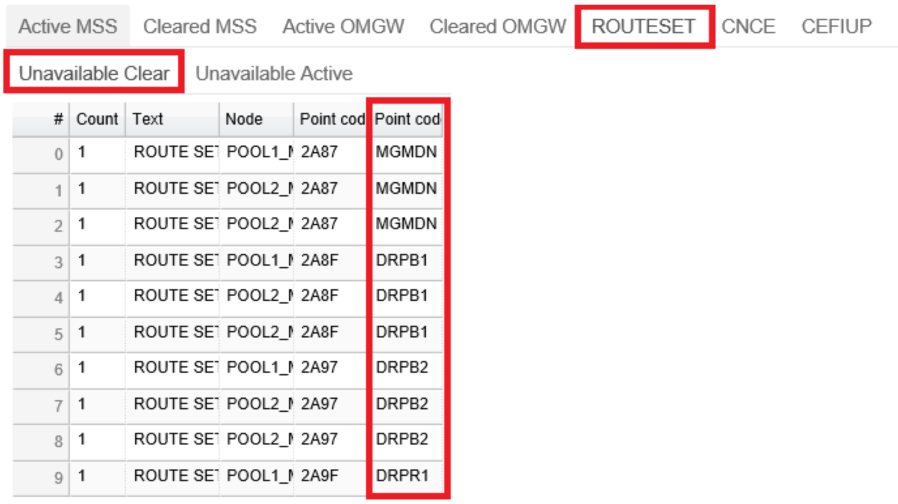
Different network conditions affect service to customers and this application unlocked the untapped potential of the NOC in a way that customer experience could be influenced.

Case 2

Route Set Unavailable is a condition that presented an alarm with a code of the peer network element impacted by the failure. Users had to take manual steps using secure shell (SSH) interactions with the element to translate the value to a name. Occurrences of these events are not very frequent; however, when they do happen, the impact on the network is very high and the translation needs to be done as quickly as possible. The manual translation posed business risk, as it was dependent on the skills level of the engineer on shift and the scale of the fault. If the fault occurs on a large scale, multiple translations need to be done at a time, and this becomes challenging, even for the most skilled engineers, as it is a time-consuming exercise.

This part of the artefact presented the information without manual effort needed from the network engineer. The feature enabled the user to view the current status of the faulty route set (Figure 4.29), presenting a name, code, and conditions that have cleared.

In the case of a major fault, this feature enabled network engineers to get an accurate impact analysis and communicate this to the management teams as soon as they accessed the application. The earlier an impact is understood by the different teams, the better the prioritisation of the fault will be done.



#	Count	Text	Node	Point cod	Point cod
0	1	ROUTE SE	POOL1_I	2A87	MGMDN
1	1	ROUTE SE	POOL2_I	2A87	MGMDN
2	1	ROUTE SE	POOL2_I	2A87	MGMDN
3	1	ROUTE SE	POOL1_I	2A8F	DRPB1
4	1	ROUTE SE	POOL2_I	2A8F	DRPB1
5	1	ROUTE SE	POOL2_I	2A8F	DRPB1
6	1	ROUTE SE	POOL1_I	2A97	DRPB2
7	1	ROUTE SE	POOL2_I	2A97	DRPB2
8	1	ROUTE SE	POOL2_I	2A97	DRPB2
9	1	ROUTE SE	POOL1_I	2A9F	DRPR1

Figure 4.29: Cycle 1 – Prototype D

Case 3

The Cellular Network Configuration Error (CNCE) (Figure 4.30) occurs very frequently on the network and it is caused for different reasons. As in the other cases where users had to use manual intervention to understand the causes and the affected transaction types because of the large volumes, it was practically impossible for NOC engineers to unpack each of the cases and there was no visibility, of which causes and transaction types were the highest contributors to the fault. This feature Figure 4.30addressed the challenge by going through the CNCE data and presenting user-consumable visuals to the user.

Immediate value was enjoyed by the business when the feature was released since the core NOC engineers informed the researcher that the causes of this condition were now well understood and the evidence clearly pointed to misconfigurations between the RAN and the core network. This simplified presentation of information was well received by the users. The business had the data; however, no information was derived from the data prior to this feature.

Case 4

Call establishment failure in user-plane control (CEFIUP) is the last feature (Figure 4.31) of Iteration 0. Similar to the CNCE case, this feature addressed simplifying an operational process. The users had an interest in these cases and needed a feature to assist with the analysis of the condition.

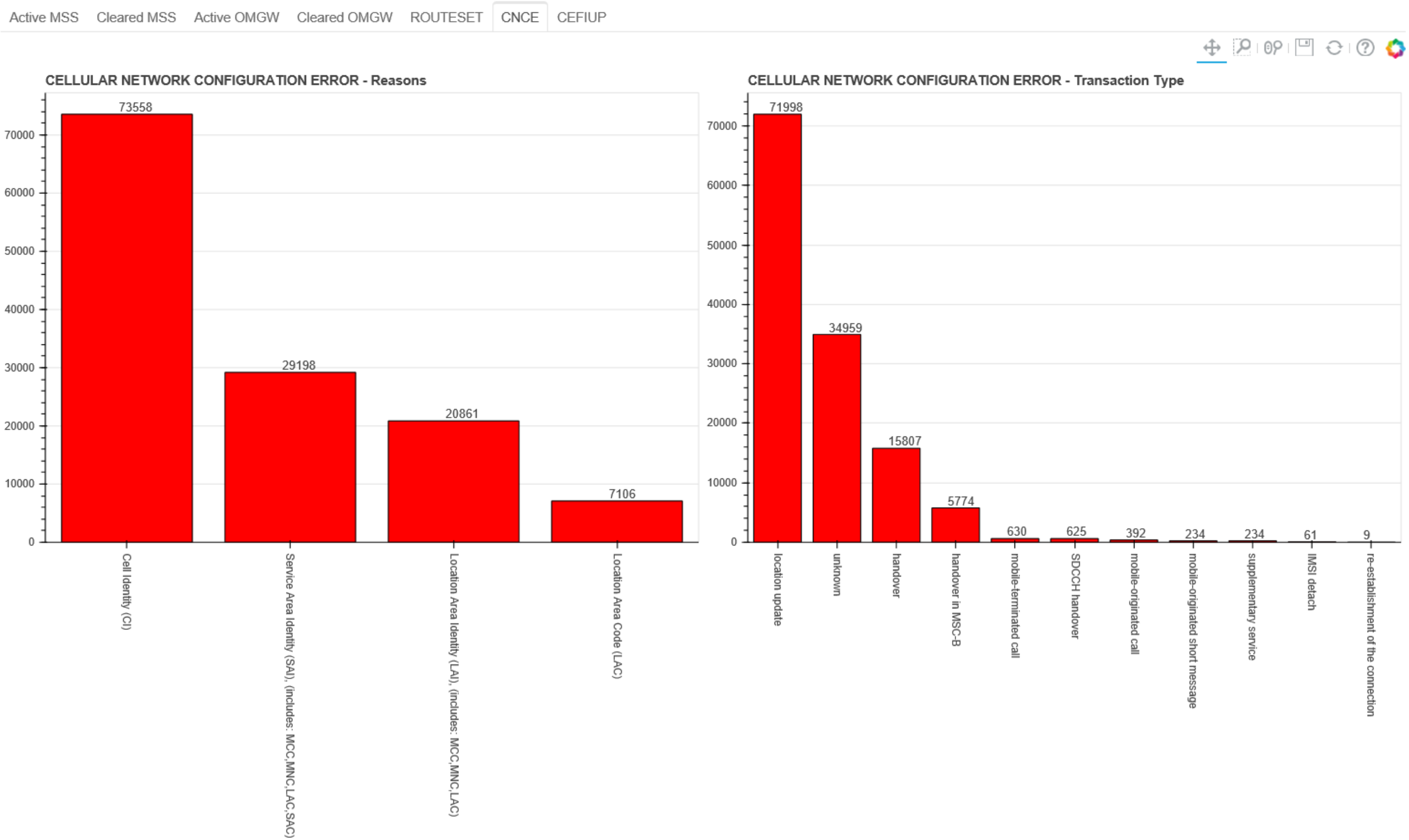


Figure 4.30: Cycle 1 – Prototype E

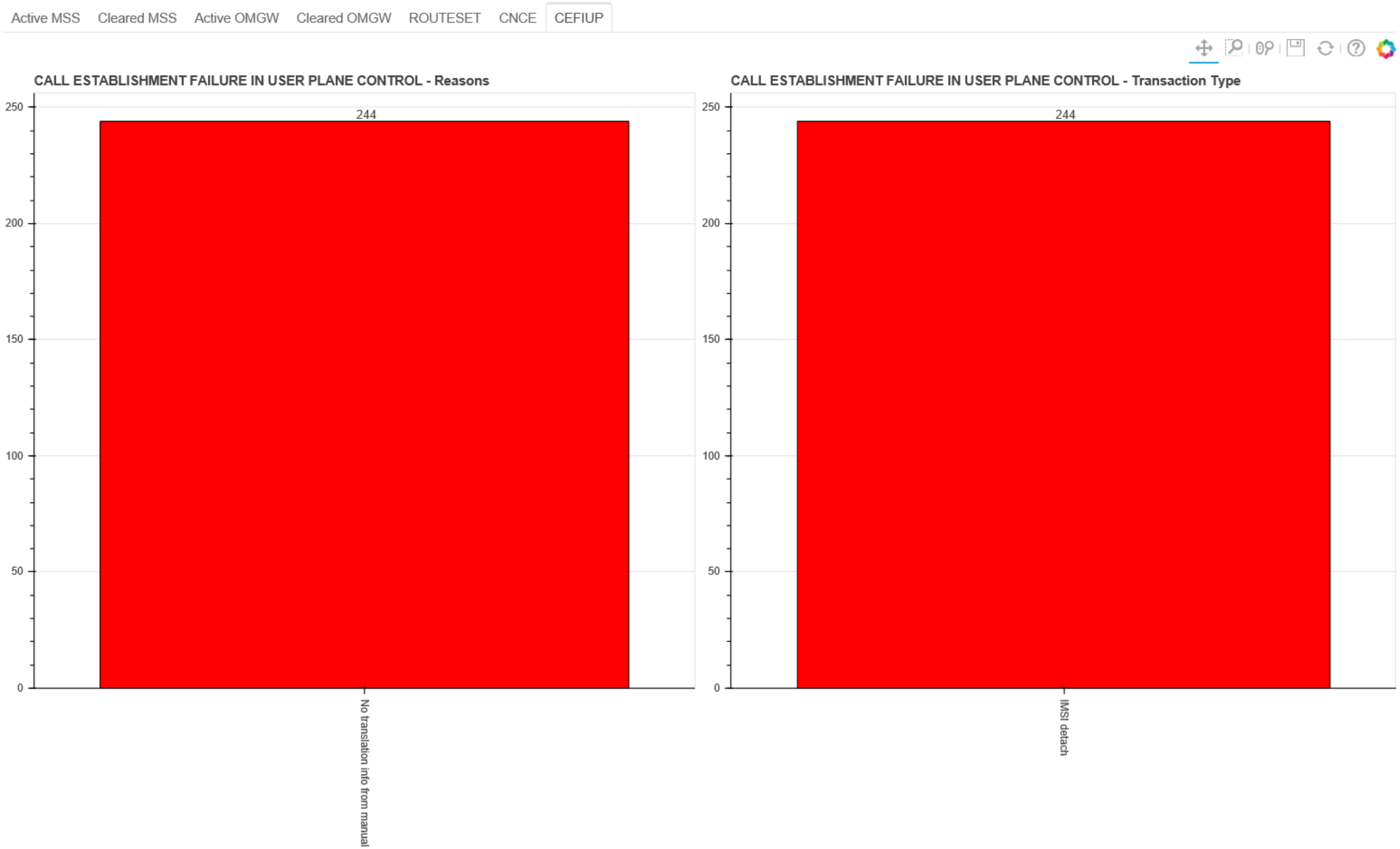


Figure 4.31: Cycle 1 – Prototype F

4.3.7 Collaborative application evaluation

The users were happy with the application since it met the communicated requirements (Appendix D); however, the success of the iteration led to more requirements being tabled by the participants. Iteration 1 addressed the alarm data, while the incident data were not put into perspective as this was not a requirement. Users made a strong case for gaining operational efficiency if incident data were put into perspective.

The exact requirement was that all faults under investigation should be logged and tracked through the incident management tool, which is a different tool from the alarm monitoring tool. The two tools, from the NOC's point of view, work closely together. Users requested the following:

i) The ability to quantify the number of open and closed incidents per team and per person
ii) A breakdown of incidents by network priority

4.3.8 Reflection and revision

Accessing and understanding the data was challenging because of the researcher's lack of experience in this area. Domain knowledge was a key enabler in the development work. Setting up Anaconda Navigator (development graphical user interface) and installing libraries required the IT security department to open firewall ports. The actual development went well and the results were satisfactory to the developer. The focus was on building a functional artefact; very limited effort was put into the cosmetics since the main objective of the artefact was not to create the best user interface (UI), but rather to create a functional tool. The design section (4.3.4.2) put in place prior to the coding proved to be extremely useful, as this helped the developer to remain focused. The alarm data volume handled by the application proved to be very high, at over 200,000 events per day), further justifying why it become challenging for people to manage the network.

New requirements addressing incident management data analytics were tabled and the researcher considered them for Cycle 2 in Iteration 1. Overall, the users were happy with the artefact and confirmed that it addressed the issues they raised.

4.4 Cycle 2 – Iteration 1

Figure 4.10d indicates the current stage which Cycle 2.

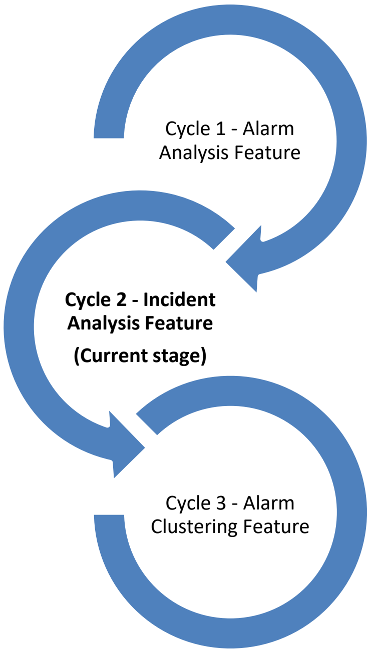


Figure 4.10d: Development Cycle 2

4.4.1 Iteration objective

The management tracking of network faults is performed through an incident management system (IMS). Each network fault attended by engineers is recorded on the IMS. In the system, the data describe the incident, the team attending to the incident, and the specific individuals investigating the faults. The users are able to view raw entries of all incidents. The issue with this system communicated by the users was that there is no way other than manually to quantify the workload distribution between the teams and the cases handled by each engineer, or cases not handled at all. The lack of this information meant that the allocation of incidents was done in way that was not based on the statistics available. Further to this, the main challenge was that each engineer was unable to track their own performance as the shift (or week) progressed. Inability to track performance in near real-time posed a challenge in that the goals set by business were missed, which had an impact on how the business performed, as well as on the appraisal of individuals. The manner in which the users communicated this requirement gave the researcher a clear picture that business and individual performance was taken seriously by the business and that the users needed a solution which would enable them at both a team and an individual level.

4.4.2 Data exploration

Incident management data are stored in a relational database. The database is different from the one used in Iteration 0, which made the situation complicated for the researcher once again, since it took a great deal of effort on the part of the researcher to understand the database schema. The assistance of the NOC L2 engineer was requested once again to help the researcher identify the data obtained from relational database, which was of value to the team. The columns of importance were identified and an exploratory data analysis was conducted using SQL Developer. The incident management application owner did not allow the researcher to establish an open connection or to embed a SQL data extraction query into the Python code hosting server, citing reasons that the researcher might run frequent queries that would impact on the performance of the incident management application. It was agreed that the researcher could compile a query, running on an agreed upon scheduled interval, on the incident management application to push the file directory to the incident management server, from where the researcher's code-hosting server could collect the file. The arrangement was honoured and both parties, the researcher and the incident management application owner, were comfortable with the arrangement.

4.4.3 Collaborative concept design

The incident management data source is different from that of the alarm data source. Figure 4.32 is an illustration of the design, detailing the process that was followed, representing the data sources and content visualisation for the user.

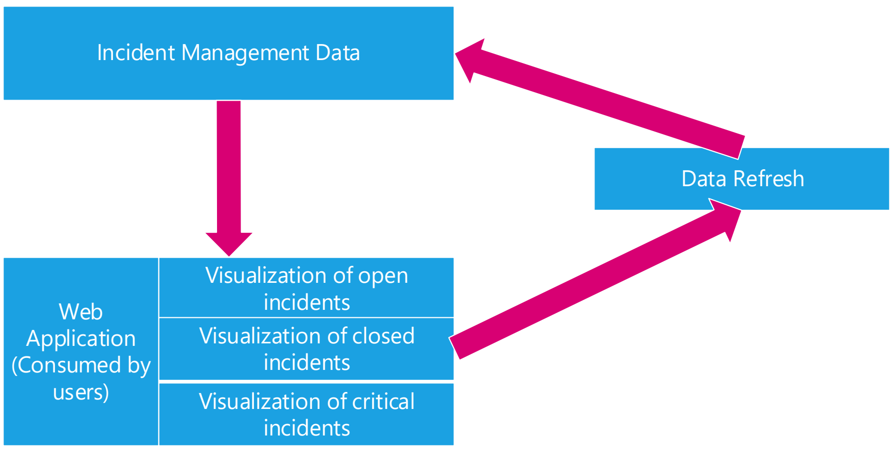
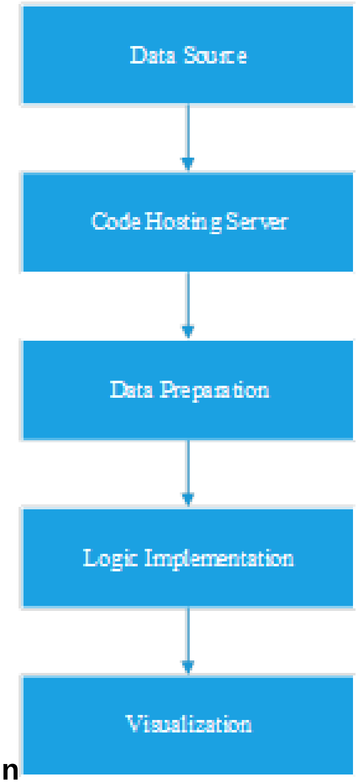


Figure 4.32: Cycle 2 – Concept Design

4.4.4 Design



4.4.4.1 High Level Design

Figure **4.33**) is similar to that of the Iteration 0 section (Figure 4.12) with the difference in the data source. Maintaining a standard in the design was deliberate from the researcher's point of view, with the objective of simplified maintenance when the application is handed over to the operations team. The benefits of the standardised design will be enjoyed by the next researcher undertaking future studies.

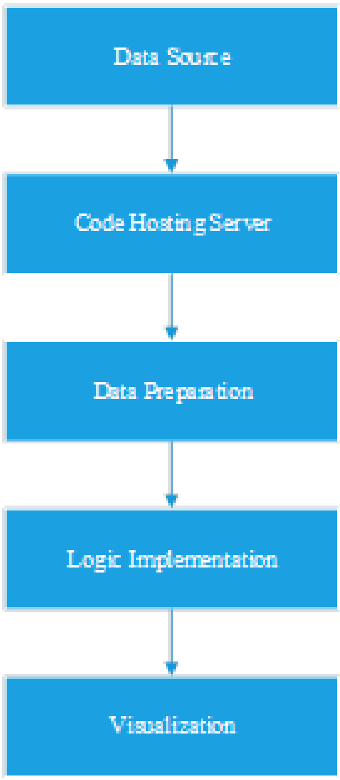


Figure 4.33: Cycle 2 – High Level Design

4.4.4.2 Low Level Design

A more detailed version (Figure 4.34) of what the researcher did in planning for artefact creation is outlined as follows:

i) Compiled data extraction SQL on the incident management database. Set the query on schedule to output the file to a specified directory.

ii) SQL was used for data extraction query and only data that was needed was extracted for performance reasons.

iii) A data frame was created from the data sourced through the data extraction query.

iv) Categorised data frames were created from the original data frame.

v) Visuals aligned to the outlined requirements by users, were created. The importance of this step is that there must be evidence that the artefact is addressing the pain points communicated by users.

vi) All created visuals were inserted into the application (user interface) and the UI was published.

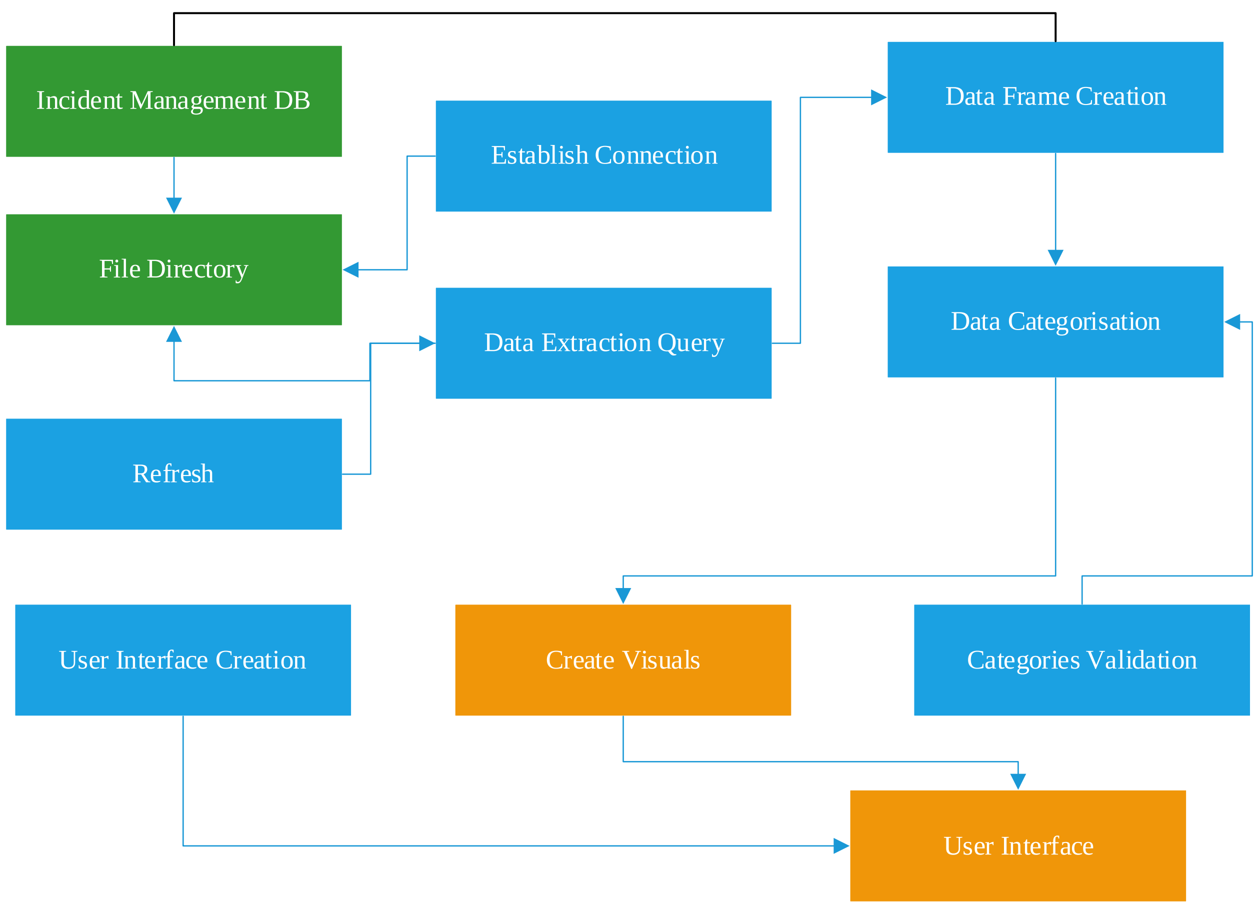


Figure 4.34: Cycle 2 – Low Level Design

4.4.5 Application development

The SQL incident management data extraction query (Figure 4.35) for the extraction of incident data for the previous 24 hours and provided to the department teams studied, is presented below.

SQL Help

```
select INCIDENT_ID,INC_STATUS,VOD_INCIDENT_ID,INC_COMPANY_NAME,INC_CI_SITE_NAME,INC_PRIORITY,INCIDENT_TYPE,REG_CREATED
        REG_MODIFIED,INC_EVENT_START,INC_EVENT_FINISH,INC_OWNER_NAME,INC_ASSIGNED_GROUP,INC_DESCRIPTION
FROM  REMEDY_MODULE.INCIDENT_DETAILS
WHERE INC_ASSIGNED_GROUP in ('CORE:FAULT MANAGEMENT SWITCHING:NMG','CORE:FAULT MANAGEMENT
VAS/IN:NMG','NMG:FM:CORE:EMAILS','CORE:NMG Digital Services')
and REG_CREATED >= trunc(sysdate)-1
```

Copy Errors to Clipboard | Validate | OK | Cancel

Figure 4.35: Cycle 2 – Data Collection

The output of the above query was then consumed in the Windows server hosting the Python code (Figure 4.36).

```
df_all = pd.read_excel("\\\\INCMANHOST\\Shared_Export\\file\\CoreNOC\\VSA_Core_Incidents_Daily-en-gb.xlsx")
```

Figure 4.36: Data frame

A data frame called **df_all** was created. At this point, data cleaning proceeded and application development was secured.

A column in the data indicated whether the incident entry was open or closed. The code presented in Figure 4.37 was used to group all the incidents that were not in a closed state since they required focus in their own right. There are various statuses that can be set to indicate whether an incident is closed, namely:

i) Assigned
ii) In progress
iii) Resolved
iv) Pending

```
df_open = df_all[df_all['INC_STATUS'] != 'Closed']
```

Figure 4.37: Exclude closed incidents

The code presented in Figure 4.38Figure 4.38 created a data frame for an **in-progress** incident with a priority that was critical.

```
df_inprogress = df_open[(df_open['INC_STATUS'] == 'In Progress') & (df_open['INC_PRIORITY'] == 'Critical')]
df_inprogress['date_time'] = pd.to_datetime('now')
df_inprogress['Last_Updated'] = df_inprogress['date_time'] - df_inprogress['REG_MODIFIED']
df_inprogress['Last_Updated']=df_inprogress['Last_Updated']/np.timedelta64(1,'h')
```

Figure 4.38: Date and time reference

The code presented in Figure 4.39 created a data frame for an assigned incident with a priority that was critical.

```
df_assigned = df_open[(df_open['INC_STATUS'] == 'Assigned') & (df_open['INC_PRIORITY'] == 'Critical')]
```

Figure 4.39: Assigned critical incidents

The code presented in Figure 4.40 created a data frame for mean time to repair the key performance indicator, which is only applicable to closed incidents.

```
df_closed = df_all[df_all['INC_STATUS'] == 'Closed']
df_closed['MTTR'] = df_closed['INC_EVENT_FINISH'] - df_closed['INC_EVENT_START']
df_closed['MTTR']=df_closed['MTTR']/np.timedelta64(1,'h')
df_closed['date_time'] = pd.to_datetime('now')
```

Figure 4.40: Mean time to repair (MTTR) calculation

The function below (Figure 4.41) was developed for creating plots to enable reuse of the function, as it would be needed multiple times. The researcher avoided a cause of writing code for each plot to be created as that could have been inefficient.

```
def get_plot(df,title):
|
    temp={"Columns":df.index,
          "values": df.values}
    src=ColumnDataSource(temp)

    plot = figure(x_range=df.index.values, title=title,plot_height=800,plot_width=700,background_fill_color='white', ba
    plot.xaxis.major_label_orientation = "vertical"
    plot.vbar(x="Columns", top = "values", width=0.8, source=src, fill_color='red', line_color='black')


    #Anotations
    labels = LabelSet(x="Columns", y="values", text="values",
                            x_offset=-8, y_offset=0, source=src,text_font_size="9pt", render_mode='canvas')
    plot.add_layout(labels)

    plot.xgrid.grid_line_color = None
    plot.y_range.start = 0
    return plot
```

Figure 4.41: Plotting function

The preparation of the tables for in-progress and assigned information is as in the figure below; the selection of the relevant columns and the sizing of columns and naming of column headings is done using the code (Figure 4.42).

```
data = {"INCIDENT_ID" :df_inprogress.INCIDENT_ID,
    "INC_CI_SITE_NAME" : df_inprogress.INC_CI_SITE_NAME,
#     "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
    "INC_OWNER_NAME" : df_inprogress.INC_OWNER_NAME,
    "INC_DESCRIPTION" : df_inprogress.INC_DESCRIPTION
    }
src = ColumnDataSource(data)
columns = [
    TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
    TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#     TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
    TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
    TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table = DataTable(source=src, columns=columns, width=800, height=600, editable=True)
```

```
# show (data_table)
```

```
data1 = {"INCIDENT_ID" :df_assigned.INCIDENT_ID,
    "INC_CI_SITE_NAME" : df_assigned.INC_CI_SITE_NAME,
#     "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
    "INC_OWNER_NAME" : df_assigned.INC_OWNER_NAME,
    "INC_DESCRIPTION" : df_assigned.INC_DESCRIPTION
    }
src = ColumnDataSource(data1)
columns = [
    TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
    TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#     TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
    TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
    TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table1 = DataTable(source=src, columns=columns, width=800, height=600, editable=True)
```

```
# show (data_table1)
```

Figure 4.42: Tabular visualisation preparation

The base structure of the user interface was created with the following code: one function for creating a tab for plots (Figure 4.43) and the other function for creating a tab for tables.

```
def make_tab(fig1,fig2,fig3,fig4,title):
    grid = gridplot([[fig1,fig2],[fig3,fig4]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab_table(fig1,title):
    grid = gridplot([[fig1,None],[None,None]])
    tab = Panel(child=grid,title=title)
    return tab
```

Figure 4.43: Prototype 2 tabs

The code in Figure 4.44 was created to insert plots into the tabs. Px are figures (plots) and the **get_plot** function is reused eight times, reducing unnecessary repetition.

```
p3 = get_plot(df_open["INC_ASSIGNED_GROUP"].value_counts().head(20),"Open Incidents Per Team")
p4 = get_plot(df_open["INC_OWNER_NAME"].value_counts().head(20),"Open Incidents Per Person")
p1 = get_plot(df_open["INC_STATUS"].value_counts().head(20),"Open Incidents Count - Status")
p2 = get_plot(df_open["INC_PRIORITY"].value_counts().head(20),"Open Incidents Count - Priority")
tab1 = make_tab(p3,p4,p1,p2,"Active Incidents")

p3 = get_plot(df_closed["INC_ASSIGNED_GROUP"].value_counts().head(20),"Closed Incidents Per Team")
p4 = get_plot(df_closed["INC_OWNER_NAME"].value_counts().head(20),"Closed Incidents Per Person")
p1 = get_plot(df_closed["INC_DESCRIPTION"].value_counts().head(20),"Closed Incidents Count - Total")
p2 = get_plot(df_closed["INC_PRIORITY"].value_counts().head(20),"Closed Incidents Count - Priority")
tab2 = make_tab(p3,p4,p1,p2,"Closed Incidents")
```

Figure 4.44: Figures to tabs

Tabs 3 and 4 of the application presented tabular information and the code below was developed to visualise the two tables, accordingly.

```
data = {"INCIDENT_ID" :df_inprogress.INCIDENT_ID,
    "INC_CI_SITE_NAME" : df_inprogress.INC_CI_SITE_NAME,
#     "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
    "INC_OWNER_NAME" : df_inprogress.INC_OWNER_NAME,
    "INC_DESCRIPTION" : df_inprogress.INC_DESCRIPTION
    }
src = ColumnDataSource(data)
columns = [
    TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
    TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#     TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
    TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
    TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table = DataTable(source=src, columns=columns, width=900, height=600, editable=True)

tab3 = make_tab_table(data_table,"Critical In Progress Incidents")

data1 = {"INCIDENT_ID" :df_assigned.INCIDENT_ID,
    "INC_CI_SITE_NAME" : df_assigned.INC_CI_SITE_NAME,
#     "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
    "INC_OWNER_NAME" : df_assigned.INC_OWNER_NAME,
    "INC_DESCRIPTION" : df_assigned.INC_DESCRIPTION
    }
src = ColumnDataSource(data1)
columns = [
    TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
    TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#     TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
    TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
    TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table1 = DataTable(source=src, columns=columns, width=800, height=600, editable=True)

tab4 = make_tab_table(data_table1,"Critical Assigned Incidents")
```

Figure 4.45: Tabular data visualisation

All the figures were mapped to their respective tabs in the codes (Figure 4.45), and the final step was to populate the data in all four tabs, which was done as shown in Figure 4.46)

```
dash = Tabs(tabs=([tab1,tab2,tab3,tab4]))

show(dash)
```

Figure 4.46: Visualise information

4.4.6 Cycle 2 Prototype

The application with the four tabs is the artefact that was demonstrated to users. The first two tabs displayed four figures (plots) and the last two tabs presented the tabular information.

The **first tab** presented information on the incidents not in a closed state. These incidents still needed action from the users.

- Figure 4.47Figure 4.47 presents the number of incidents managed by team (there are 4 teams in the department)
- Figure 4.48 presents the number of incidents by status and priority

- Figure 4.49 presents the number of incidents by status; these statuses are all the possible statuses except the closed status since this tab excludes closed incidents
- Figure 4.50 displays information by severity of incident

The **second tab** displayed closed incidents.

- Figure 4.49 and Figure 4.50 quantify the number of faults managed by description as well as by priority (Figure 4.50)
- In Figure 4.49, the graph on the left focused on the team, while the graph on the right focused on individual

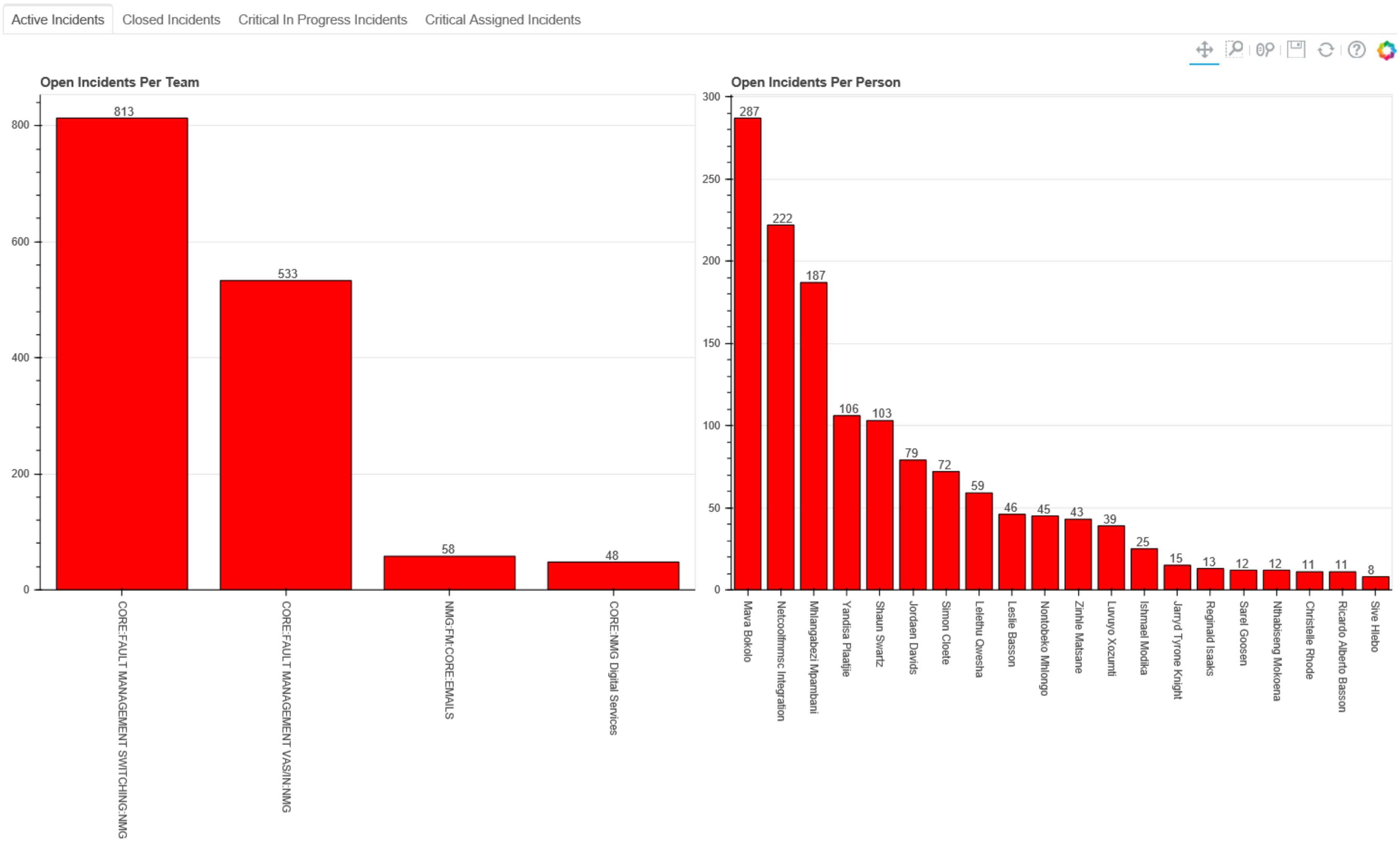


Figure 4.47: Cycle 2 – Prototype A

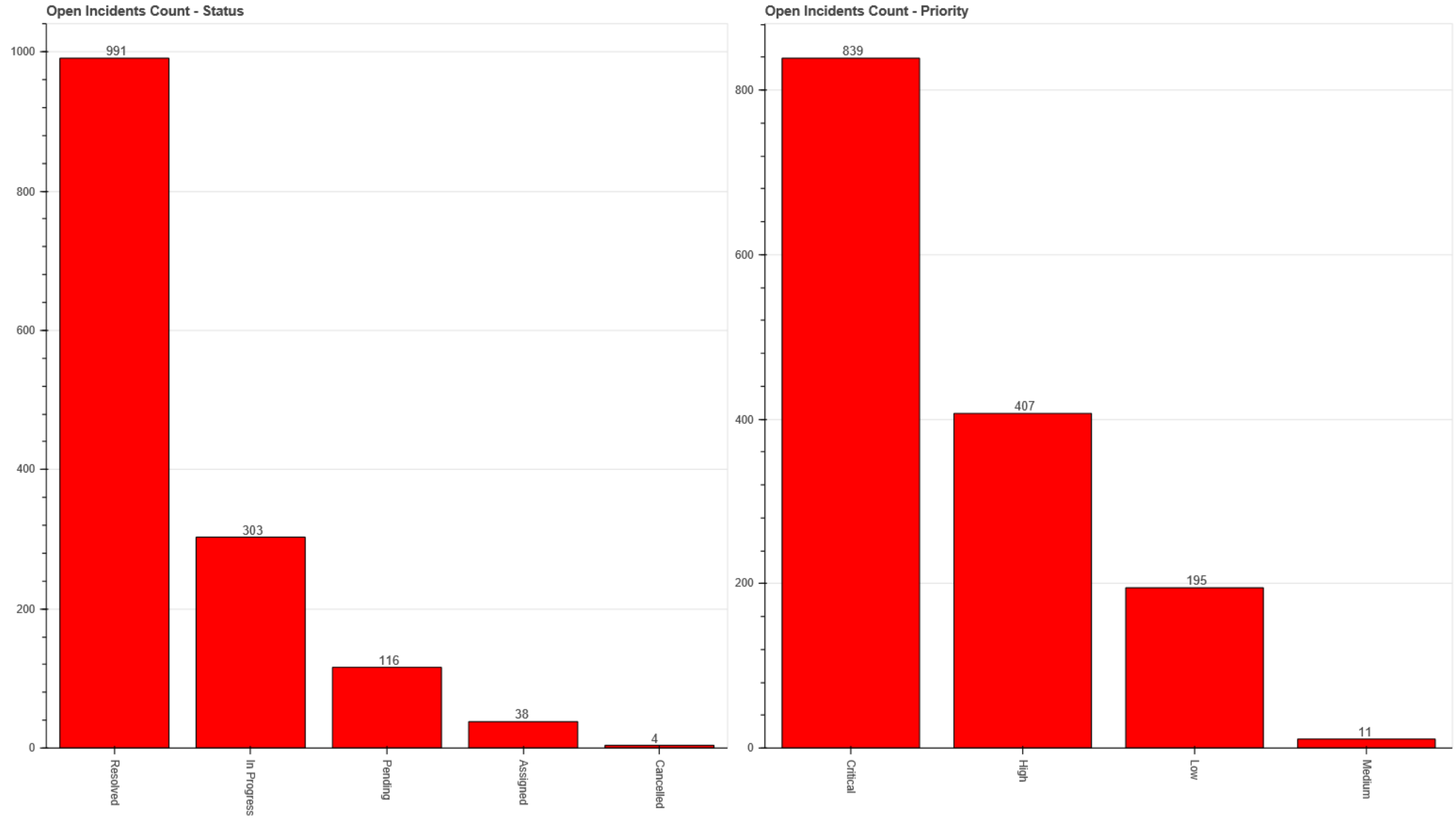


Figure 4.48: Cycle 2 – Prototype B

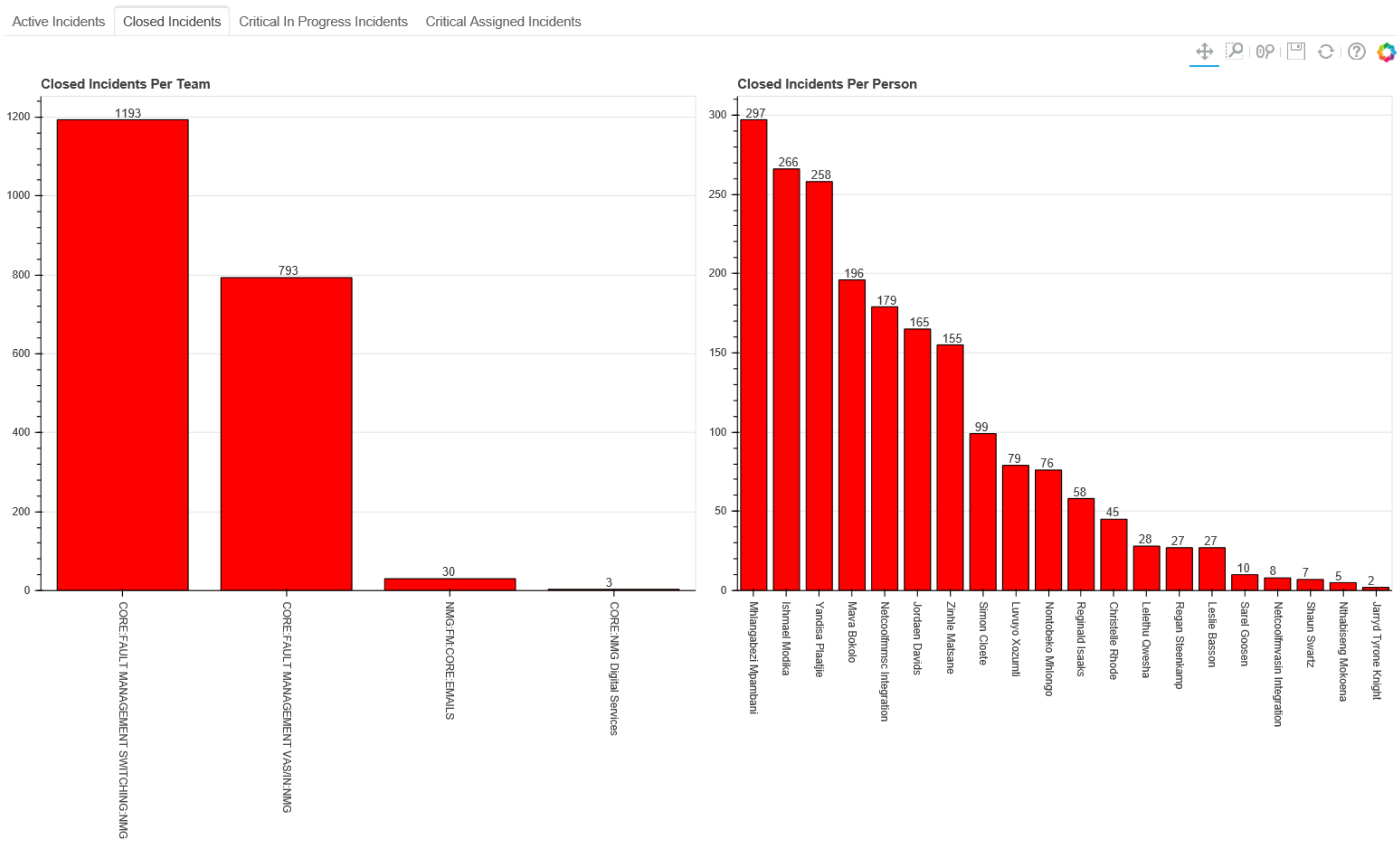


Figure 4.49: Cycle 2 – Prototype C

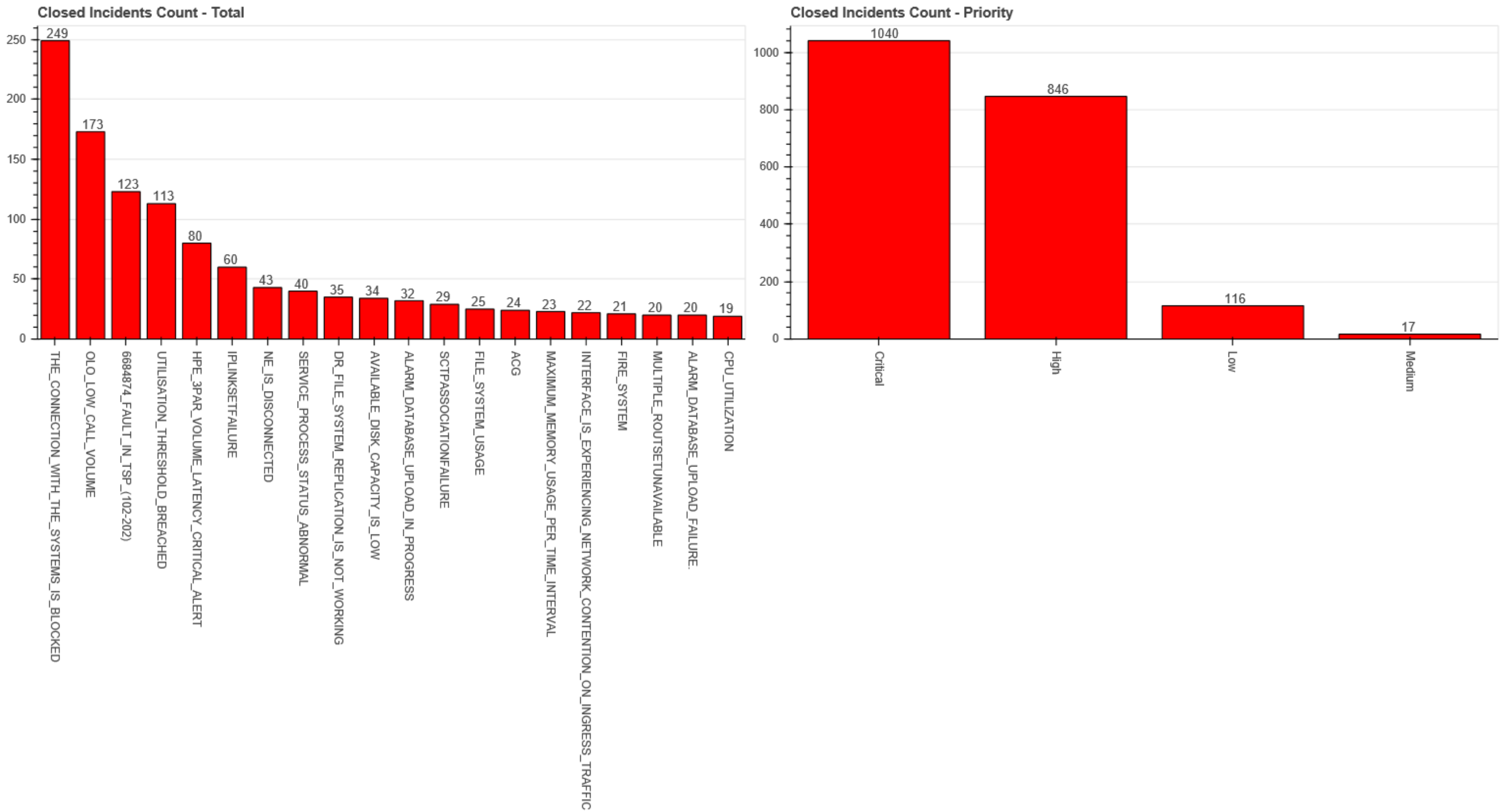


Figure 4.50: Cycle 2 – Prototype D

4.4.7 Collaborative application evaluation

The researcher experienced challenges with access to data and implemented an innovative way to reach the desired outcome, despite the challenge. The challenge is one of common scenarios that a developer can experience in any project and the researcher gained valuable experience for the future, as a result. The level of complexity of the iteration was medium in terms of the development work required. Learning how to manage incident data was useful because this process is common in the IT and engineering domain, specifically in respect to maintenance. Since this study took place in an information systems environment, the researcher experienced a practical case where the organisation had data that was not fully explored in a manner that would benefit business, and the artefact created in this iteration was well received by the users.

The users' feedback (Appendix H) on the iteration indicated that the communicated requirements were addressed and that there was lack of new capacity to learn the data and indicate the relation of the network faults. The background to this requirement was that, in multiple cases, faults that run simultaneously and the relation of these faults is understood by second level engineers (L2 engineers). The grouping of the faults was based on the knowledge of the L2 engineers, and it proved to be challenging to source this knowledge after office hours since the L2 engineers were not in the office at the time. Considering the value added by Iteration 0 and Iteration 1, the researcher agreed to accept this requirement as the last iteration of the study.

4.5 Cycle 3 – Iteration 2

Figure 4.10e indicates the current stage which Cycle 3.

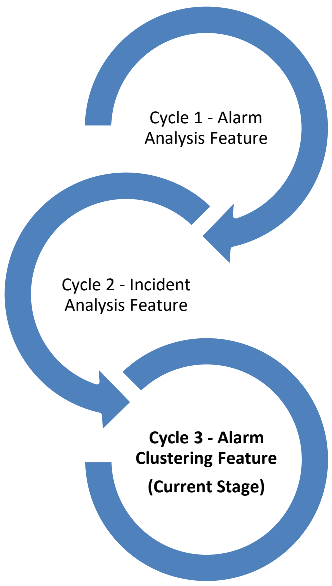


Figure 4.10e: Development Cycle 3

4.5.1 Iteration objective

The users identified the lack of an alarm grouping mechanism without human intervention, based on learning the data, as an operational pain point. The problem originated from the fact there are two levels of engineers, first level and second level, with a different extent of understanding. The second level engineers used their knowledge to do alarm grouping after events had taken place, and manually implemented the rule set based on the groupings that they would operate on. The main challenge that the Level 1 engineers faced was the fact that new occurrences of scenarios not catered for were frequent and, at the time they occurred, they (the Level 1 team) were unable to understand how to do the groupings. The only way was to contact the Level 2 team to assist with applying their knowledge by providing guidance.

With technology evolving rapidly:

i) Network element upgrades that result in alarms, change:
   - New alarms are introduced
   - Some old alarms are discontinued
   - Some alarms are modified

ii) Implementation of new technologies:
   - In this use case the main example was the migration to cloud environment in progress

Different alarm structures are introduced to the environment as more virtual network functions (VNF) are being deployed.

Level 2 engineers do not always have answers readily at hand for the Level 1 team, as they are required to learn the data and refer to manuals. Cases where answers are not readily available take longer for the team to work through. In the time that the analysis is in progress, the Level 1 team does not have any way of attending to these types of faults.

4.5.2 Data exploration

The data consumed in this exercise has already been explored in section 4.3.2 and efforts were made to clean the dataset. An effort was made to quantify data (Figure 4.51) since this data were needed to identify the best solution.

```
In [61]: import pandas as pd
         tmpq=pd.read_sql(mssrawq,con=db)
         desc = tmpq["TEXT"].describe()
         desc

Out[61]: count                                              2351688
         unique                                                 129
         top       NOT ENOUGH DIGITS IN CALLED NUMBER FOR ATTRIBU...
         freq                                                542163
         Name: TEXT, dtype: object
```

Figure 4.51: Quantify data

Over two million alarm entries over two 14-day periods were gathered for just one device type, a Mobile Switching Server, being studied.

4.5.3 Collaborative concept design

Data were explored and proven to be more sufficient for implementing a machine learning solution. Well known machine learning techniques where theoretically assessed against the problem being solved. The range of data assessed were in the unsupervised, supervised and reinforcement categories. The systematic design illustrated Figure 4.52 represents how the artefact was developed.

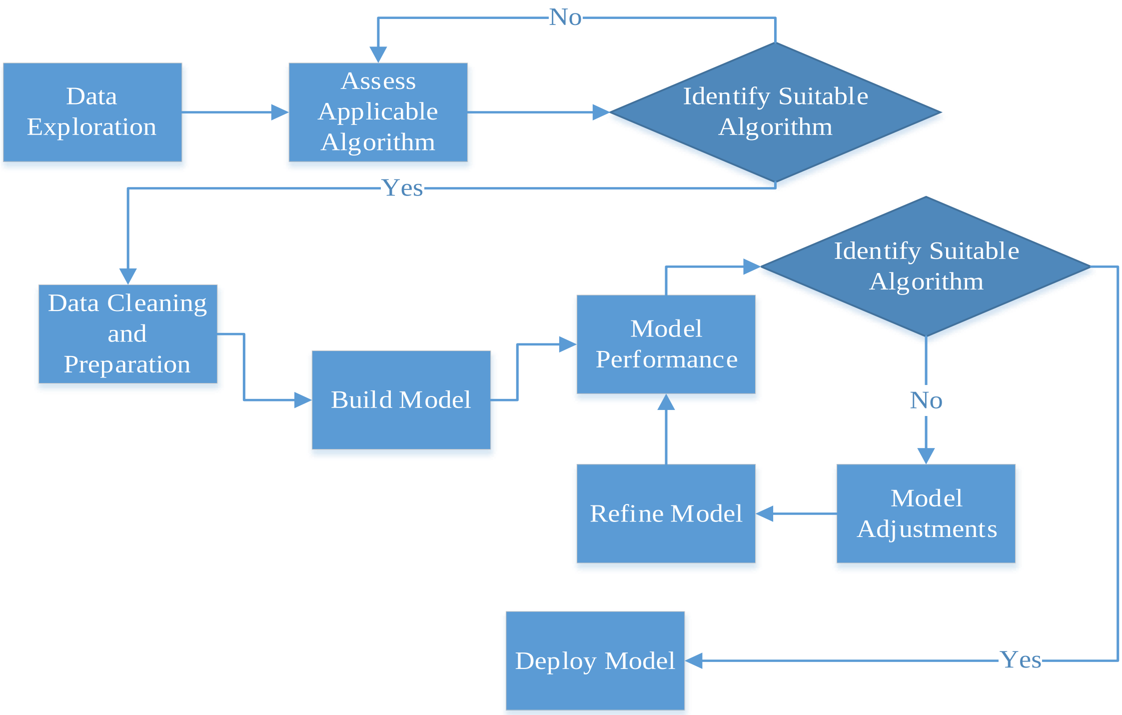


Figure 4.52: Cycle 3 – Design

Supervised models (both classification and regression) require data that are labelled, and the data dealt with in this study were not labelled. The reinforcement model proved to work well with numerical data, while this particular study's data are text data.

Unsupervised models were deemed to be the most appropriate for dealing with the problem that this study was solving since they focus on identifying patterns. Considering the large volumes of data dealt with (2.3 million data records over 14 days) meant that the KMeans clustering algorithm was the best fit for solving the problem.

Implementing the correct ML algorithm for the problem is essential, as this has a direct impact on the success or failure of the project. Implementing a model that is not appropriate for the data will lead to a problem not being solved, which is why the researcher took a systematic approach to the selection of the ML algorithm selection. KMeans is well known and has been applied in studies for years, giving good academic grounds for selecting the algorithm. The model was built, refined until it reached a satisfactory level and then deployed.

The researcher learnt that the implementation of KMeans requires a number of clusters to be defined. Defining the number of clusters can be done using trial and error, which is little more than guesswork, which was not a good approach since there was no scientific reasoning behind it. The other option was to implement a scientific mechanism to assist in selecting the number of clusters for the data being studied. There researcher applied a scientific method, which resulted in additional programming methods being explored.

4.5.4 Application development

4.5.4.1 KMeans clustering implementation

The sourcing of the data reused the code depicted in section 4.3.5 since this was a continuation exercise, taken to a more advanced level. Libraries were imported (Figure 4.53).

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler
import cx_Oracle
```

Figure 4.53: Imported libraries

Figure 4.54 illustrates the architecture of KMeans. The architecture focuses on learning patterns that involve four main fields, namely node, alarm time, text and alarm type, since their importance was clearly highlighted by the users. At any point in the future, should studies or requirements need to focus on different fields, the developer can adjust the selection of fields and the desired results will be achieved with minimal effort. Considering that this study was done to meet the required academic standards, there is a possibility that other academics may need to reproduce this implementation, either 'as is' or differently. This approach to development allows for that.

```
def cluster(num_clusters, tmp):
    X = tmp[tmp.columns.difference(['NODE','ALARM_TIME','TEXT','ALARM_TYPE'])]
    kmeans = KMeans(n_clusters=num_clusters, random_state=0).fit(X)
    tmp['CLUSTER'] = kmeans.labels_
    return tmp
```

Figure 4.54: KMeans architecture

The visualisation of the cluster was implemented with the objective of presenting patterns of each of the main data fields. This function (Figure 4.55) will be called at the model deployment stage.

The need to further clean the data for ease of consumption was needed and the date field was decomposed. This code (Figure 4.56) allows the application to analyse data on the basis of an hourly, daily, weekly, monthly, quarterly and yearly aggregation. In case where more data are analysed, for example, by loading years of data, the solution will continue to function without any adjustments required – that is, future proofing the implementation.

```
def plot_all(df,title):

    cols = ["ALARM_TIME_hours",
                                "SEVERITY",
                                "ALARM_TYPE","TEXT","CLUSTER"]

    grid = sns.pairplot(df[cols],hue='CLUSTER')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='SEVERITY')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='ALARM_TYPE')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='TEXT')
    grid.fig.suptitle(title)
```

Figure 4.55: Cluster visualisation

```
def decompose_date_time_features(df, index):
    """
    Uses a column index for a date/time attributes in a data frame and decomposes this field into its
    various atomic parts.  Each part is then added to the data frame as a new
    column/attribute.
    :param df: The data frame in which date/time attributes are to be found.
    :param index: The index of the column containing the date/time attribute.
    :return: The dataframe appended with columns representing the decomposed date/time information
    """
    if(index < 0 or index >= len(df.columns)):
        print("Column index invalid")
        return

    colString = df.columns[index] + "_"
    units = ["year", "month", "day_of_month", "day_of_week_number", "day_of_year_number", "week_of_year",
             "quarter_of_year", "hours", "minutes", "seconds"]
    units = [colString + v for v in units]
    df[units[0]] = pd.DatetimeIndex(df[df.columns[index]]).year
    df[units[1]] = pd.DatetimeIndex(df[df.columns[index]]).month
    df[units[2]] = pd.DatetimeIndex(df[df.columns[index]]).day
    df[units[3]] = pd.DatetimeIndex(df[df.columns[index]]).dayofweek
    df[units[4]] = pd.DatetimeIndex(df[df.columns[index]]).dayofyear
    df[units[5]] = pd.DatetimeIndex(df[df.columns[index]]).weekofyear
    df[units[6]] = pd.DatetimeIndex(df[df.columns[index]]).quarter
    df[units[7]] = pd.DatetimeIndex(df[df.columns[index]]).hour
    return df
```

Figure 4.56: Date and time decomposition

The code below (Figure 4.57) will provide multiple solutions, including solutions to the following:

- Define the number of clusters for our Kmeans – this is the only user-controlled input in this unsupervised ML technique. Four clusters were implemented (Figure 4.57)
- Remove fields that are not needed for clustering – a large dataset was analysed. The researcher avoided including all fields, which would have led to the model unnecessarily performing poorly
- The data frame was enriched with the decomposed date field
- Clusters were visualised

```
numClusters =4
pd.set_option('display.max_columns', 300)

tmp=pd.read_sql(mssrawq,con=db)

cols_to_keep = ['NODE','SEVERITY','ALARM_TIME','TEXT','ALARM_TYPE']

tmp = tmp[cols_to_keep]
tmp.dropna(inplace=True)
tmp = pd.concat([tmp,pd.get_dummies(tmp['ALARM_TYPE'], prefix='ALARM_TYPE'),
                 pd.get_dummies(tmp['TEXT'], prefix='TEXT')],axis=1)
tmp = decompose_date_time_features(tmp,2)

clusters = cluster(numClusters,tmp)
plot_all(clusters,'Clusters = {}'.format(numClusters))
```

Figure 4.57: Clusters visualisation

4.5.4.2 Scientific selection of number of clusters

In this study, a heuristic method called the Elbow method (Figure 4.58) was used to work out the optimal number of clusters to be defined. This method looked at variance percentage from the dataset. The range of 1 to 10 was configured and the ideal number of clusters (numClusters) was shown to be four.

```
distortions = []
K = range(1,10)
for k in K:
    X = tmp[tmp.columns.difference(['NODE',
                                    'ALARM_TIME',
                                    'TEXT',
                                    'ALARM_TYPE'])]
    kmeansModel = KMeans(n_clusters=k)
    kmeansModel.fit(X)
    distortions.append(kmeanModel.inertia_)
```

Figure 4.58: Elbow method architecture

The code shown in Figure 4.59 was developed to visualise the output of the heuristic method.

```
plt.figure(figsize=(16,8))
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```

Figure 4.59: Elbow method visualisation

The computed average of the squared distance from cluster centres and the sum of squared distances of data points closest to their cluster centres were used. Clusters at the highest and lowest distortion are not ideal for optimal clustering. The Elbow method recommends that number of clusters (*k*) has to be a value where *k* has a low sum of squared errors (SSE). Implementation of this method to the studied data confirmed the value of *k* as 4 (red line in Figure 4.60) – the start of diminishing returns with *k* increments.

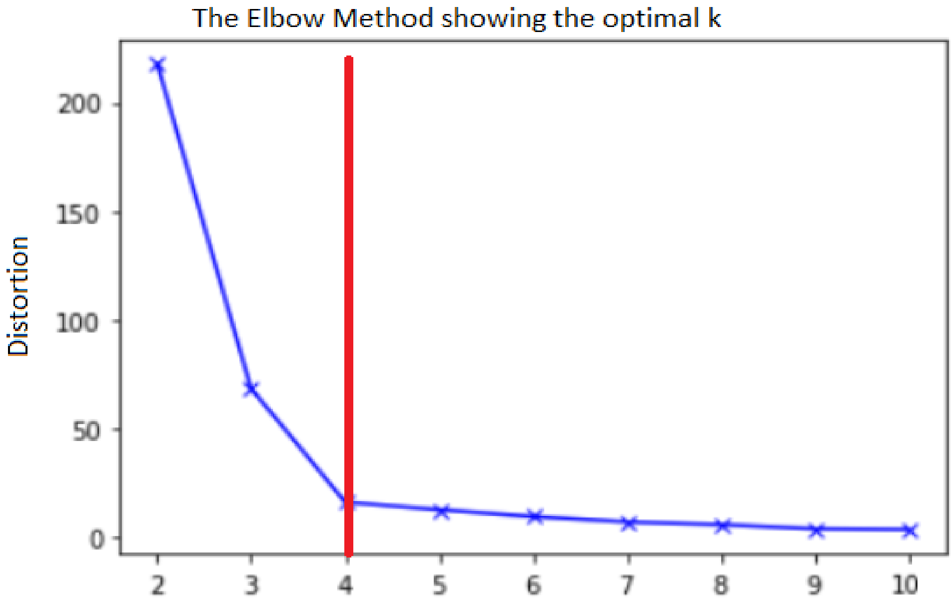


Figure 4.60: Elbow method results

4.5.4.3 How KMeans clustering works

KMeans clustering is a technique that is implemented in scenarios where the target is not known, which was the case in this study. The algorithm assesses the similarity measure, and then computes membership functions. This technique is centroid-based. Data points are plotted and grouped to the closest centroids. Each data point can only be assigned to one cluster; for this reason the number of clusters (*k* as per the code) is defined in advance. Initial centroids are set to be random in this case. The quality of the clusters is worked out by calculating if all data points are related to the best closest centroid, then the centroids will move into another different position and the grouping will be recalculated. The processes will repeated a number of times, until a point is reached where the algorithm does not need adjustment, as illustrated in Figure 4.61. The centroids do not move any longer at the point where the algorithm is satisfied. Each cycle of this process is known as an iteration. The model is intensive on computing resources; in this study, dealing with 2.3 million data points and running multiple iterations was clearly a challenging process.

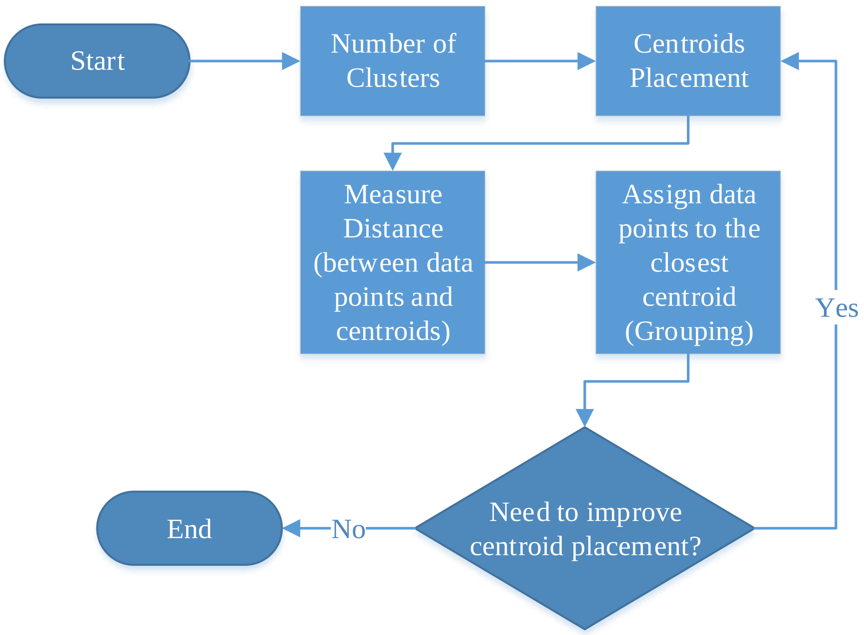


Figure 4.61: KMeans clustering algorithm

The clustering principle is illustrated in the four steps below, on a two-dimensional plane: In the initial stage (Figure 4.62), centroids, labelled C1 to C4, are randomly placed against plotted data points (data points are represented with x). During this stage, the algorithm does not compute where centroids need to be placed.

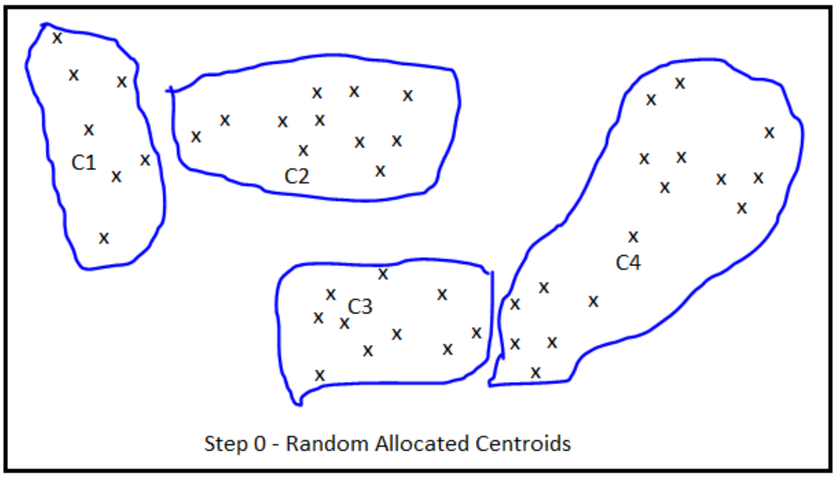


Figure 4.62: Clustering initial step

In Step Zero (after centroid placement), the computation begins and the centroids measure the distance to surrounding (closest) data points, where after they move to the ideal place. It is important to note that multiple centroids are doing the same changing-the-grouping dynamic. Take note of the size of the fourth cluster with C4 – it has a bigger area.

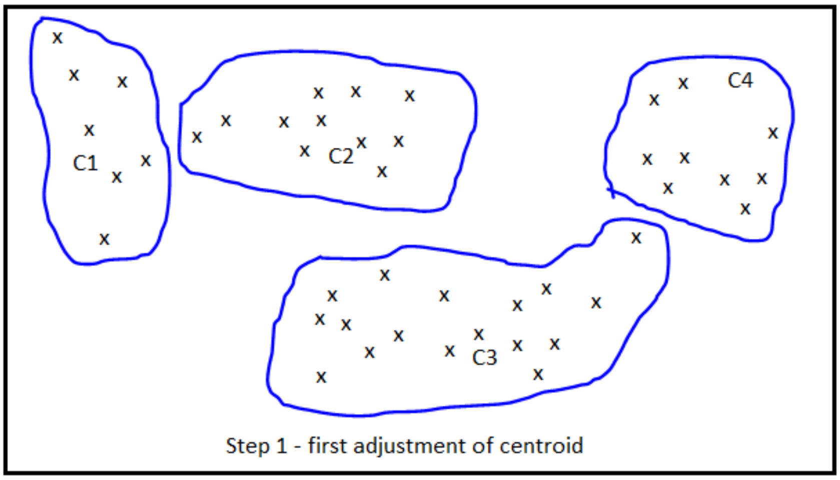


Figure 4.63: Clustering Step 1

In Step 1 (Figure 4.63), the centroids are placed as per computed results from Step Zero. The algorithm further assesses the groupings and discovers that there is opportunity for improvement, at which point re-calculation takes place.

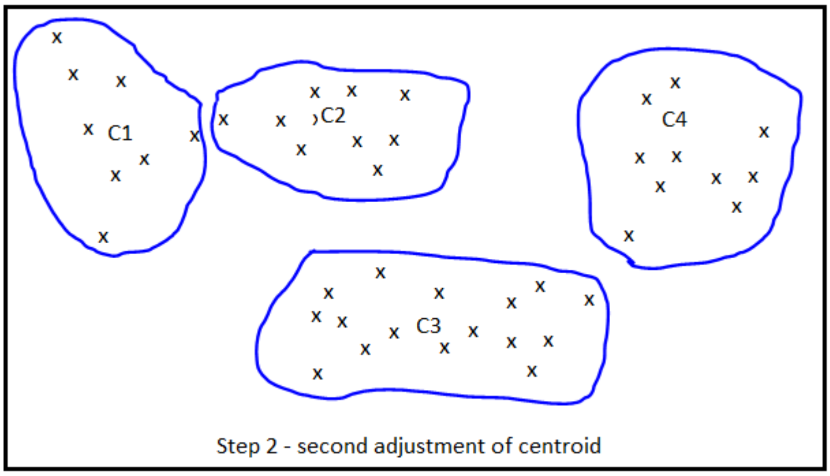


Figure 4.64: Clustering Step 2

In Step 2 (Figure 4.64), the same process that occurred in Step 1, occurs. This continues a number of times until the centroids are located at optimal positions, meaning the algorithm does not find any improvement opportunity and the result is as per step n (Figure 4.65).

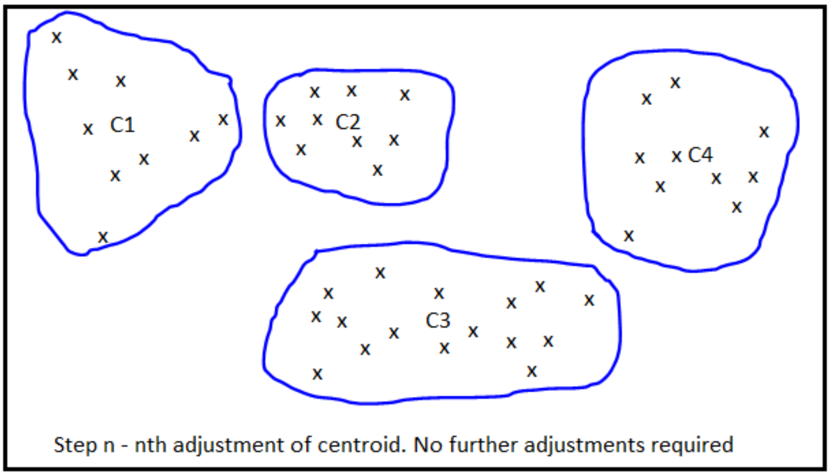


Figure 4.65: Clustering Step n

4.5.5 Demonstration

4.5.5.1 Alarm type insights learnt from the KMeans algorithm

For this study, there were two dominant clusters (Cluster 3 and Cluster 5). These two clusters displayed a different trend throughout the day. Cluster 5 was dominant from 00h00 to 08h00, reaching its peak at 04h00. Cluster 3 started to increase its dominance from 06h00 and exceeded Cluster 5 just after 08h00. Cluster 3 reached its peak at 12h00 and remained the highest until 22h00.

Figure 4.66 represents the complete results for alarm type clustering.

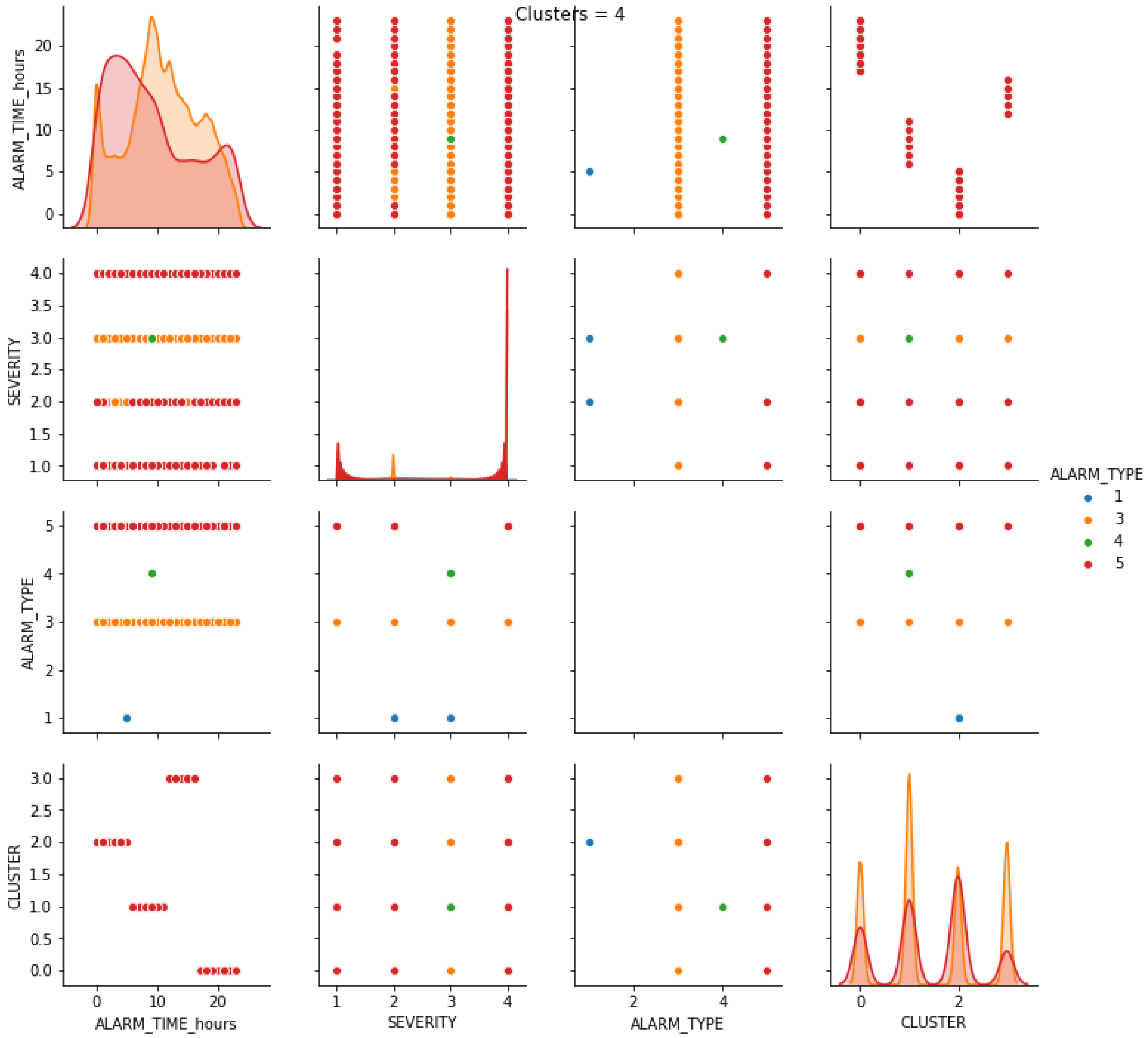


Figure 4.66: Cycle 3 – Prototype A

4.5.5.2 Severity insights learnt from the KMeans algorithm

Cluster 1 presented a unique pattern – it started building up from 04h00, continued to increase until it reached its peak at 09h00, and then gradually decreased until it reached zero at 23h00. Cluster 3 displayed changes throughout the day, reaching its peak three times, suggesting a spread throughout the day. Cluster 2 started the day very high, dominating all clusters for a short time, and then decreased as the day progressed. Cluster 4 was the only cluster to remain consistently higher throughout the day.

Figure 4.67 presents detailed clustering results for severity.

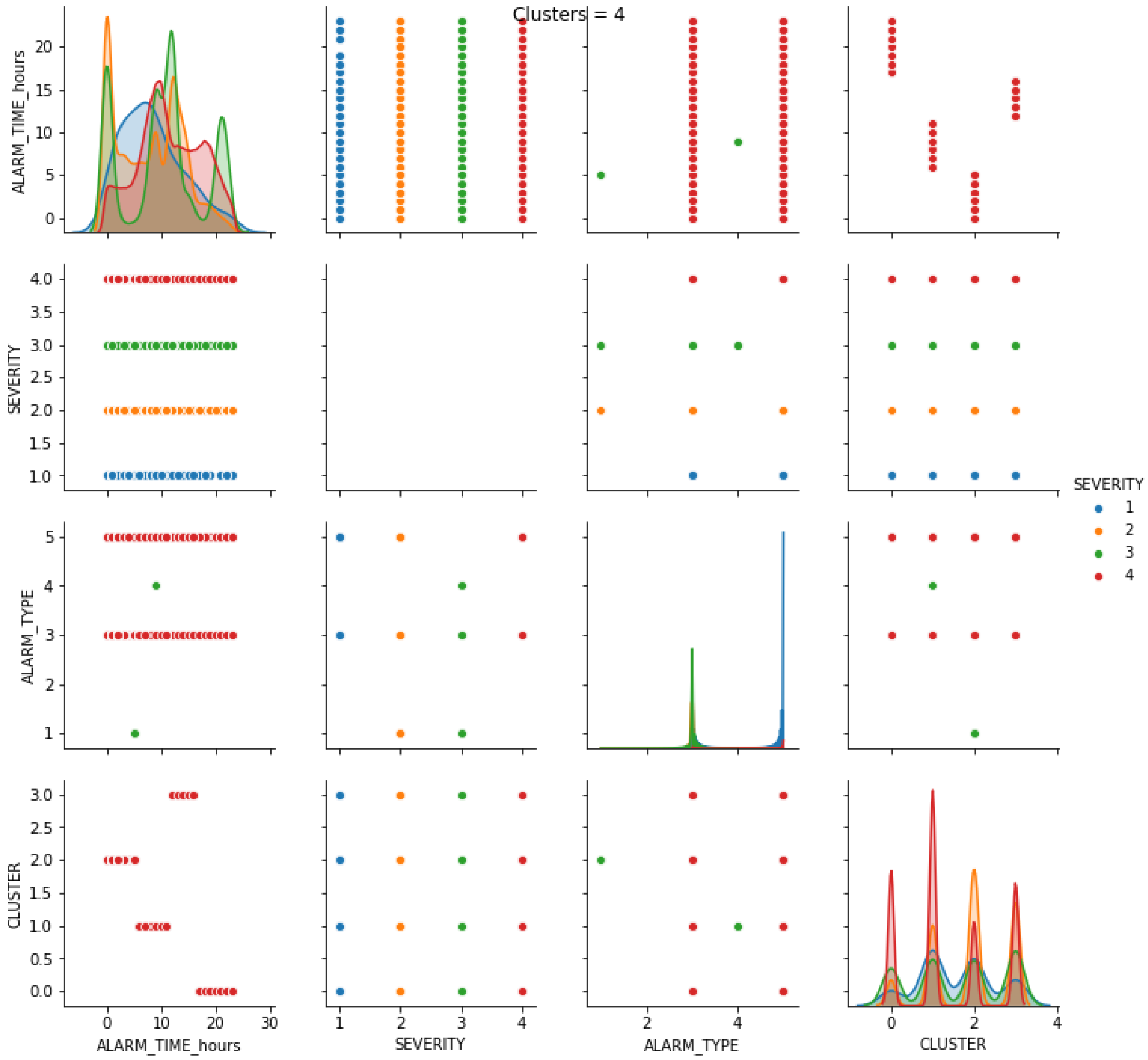


Figure 4.67: Cycle 3 – Prototype B

4.5.5.3 Text (alarm name) insights learnt from the KMeans algorithm

This component dealt with a high number of unique cases (88), which were assigned to the respective clusters. Clusters 2 and 3 were the most dominating of all the clusters. The algorithm indicated that the largest number of unique alarms are simultaneously active occurred in the evenings, between 21h00 and 23h00. The cluster represented in pink contains the group of alarms that were active throughout all hours of the day. This grouping of alarms represents different severities. The blue cluster was only dominant after 20h00. The blue cluster dominated all clusters, to a large extent, between 19h00 and 23h00. The blue cluster predominantly contained severity 4 alarms, while the orange cluster mostly contained severity 2 and 3 alarms. The green cluster was least dominant, which suggests that it covered a unique set of events that did not trigger in high volumes.

Using this algorithm enabled the researcher to outline patterns in alarms with reference to time of day, severity and alarm type. Each of the alarms was assigned to a cluster in order to define the handling of future cases. The most important aspect of the results was that all the groupings were done without human intervention, and a relationship between *alarm name*, *alarm severity* and *alarm types* was established. Each alarm was visualised with association to the cluster, giving users a simplified way to link the relationships between alarm types. Figure 4.68 represents the detailed clustering results of alarm name (text).

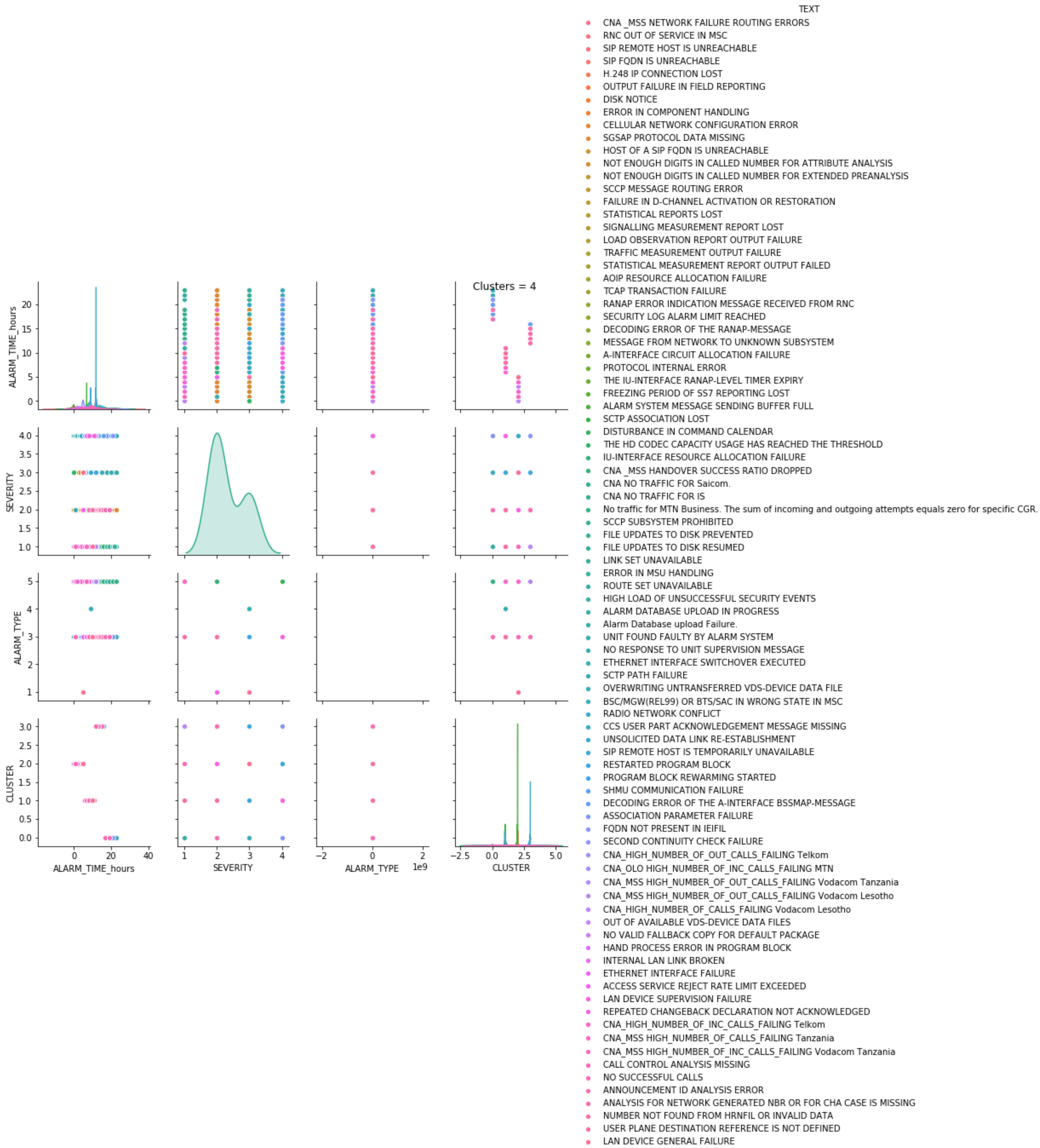


Figure 4.68: Cycle 3 – Prototype C

4.5.6 User experience

4.5.6.1 Alarm type insights

Two clusters (Cluster 3 and Cluster 5) represented transmission- and subscriber-activity related faults. Transmission network faults resulted in link failures on the core network and these faults occurred throughout the day. Subscribers were very active during the day (from 06h00 to 22h00); they were mobile between 07h00 and 18h00, causing special types of alarms to be generated as results of paging and handover activities. Alarms in these two clusters did not require immediate and urgent action from the NOC. Clusters 1 and 4 were not triggered in high volumes and, from experience, required a higher level of urgency.

4.5.6.2 Severity insights

Major network maintenance (planned network changes) occurred between 22h00 and 05h00. During that time, it was common to receive a high number of service impacting alarms that would generally not be triggered during the day, unless there was a major fault. The blue cluster indicated planned network maintenance period alarms. The red cluster represented common alarms, active throughout the day. The red cluster events were generally those that kept the person on shift occupied, thereby exposing the organisation to risk of service in cases where impacting alarms were missed.

4.5.6.3 Text (alarm name) insights

In Figure 4.69, the grouping with the alarm names represents the behaviour as per shift experience.

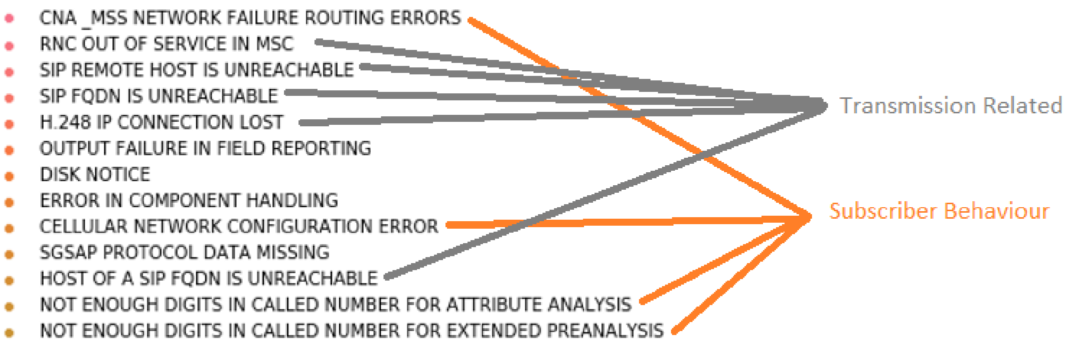


Figure 4.69: User experience feedback A

Commonly active throughout the day, multiple events were generated in the nodes (mobile switching servers) for different reasons. In the case of parameter misalignment, different network transactions received the same response since they were all trying to use the parameter, resulting in multiple valid alarms pointing to a common cause. Misaligned and/or incorrect parameters led to service impact (voice calls in this case), depending on their criticality. Figure 4.70 below is a representation of how the user-related artefact outputs to domain knowledge.

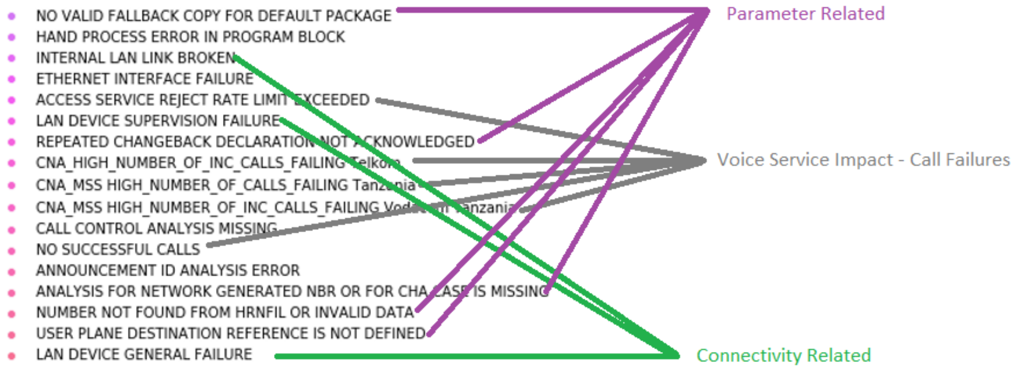


Figure 4.70: User experience feedback B

The events below displayed a high impact and therefore need urgent attention. These events did not generally trigger during the day; if they did, there was a serious fault that needed to be attended to. During the scheduled maintenance period (22h00 to 05h00), these alarms were common. The relationship of domain knowledge to cluster results for the blue cluster is illustrated in Figure 4.71.

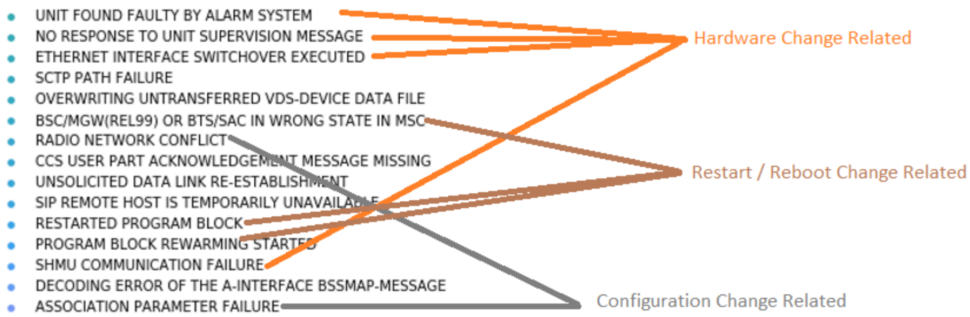


Figure 4.71: User experience feedback C

4.5.7 Reflection and review

This cycle was the most complex from a design, planning and development point of view when compared to the first two cycles. Complexity was introduced by the unsupervised ML component, that is, the implementation of KMeans clustering. The researcher experienced a number of challenges:

a) How to choose the number of clusters

The first step of KMeans was to choose the number of clusters leading to the groups. This was a challenge, as the researcher could randomly select a number; however, there would not have been any scientific basis to do this. This aspect risked diluting the outcome of the study. The researcher had to spend time learning the best way to perform this task, which led to the discovery of the Elbow method. This method was clearly understood, theoretically. The practical aspect of writing code for the dataset being studied was another challenge, which necessitated further research and reviewing multiple implementation codes (that were on numerical datasets, different from data on this study) to reach the point of a working implementation.

b) Finding meaningful fields to learn

The alarm data had over 20 fields, and learning patterns from fields not having a high impact could lead to meaningless results. For the researcher, not being an NOC subject matter expert, this task was not easy. Engagement with the users led to optimal field selection. The power of the collaborative method proved to be very successful.

c) The development environment

Clustering, or more specifically KMeans, is a very resource-intensive algorithm. The dataset was extremely large and required a high specification device for computation (Figure 4.72). The algorithm ran on the researcher's computer; however, it took over 6 hours to provide results. The solution was migrated to a high specification device, a Virtual Machine running on Windows 2016, and computation time improved tremendously.

Figure 4.72: Virtual Machine (VM) used for development

Other than the abovementioned challenges, the cycle went well and provided results that were confirmed to be meaningful to the users (see Appendix I). The immediate value was proven when users gave feedback that related to the results (detailed in section 4.5.6), and its relevance to the problem initially outlined, was clear.

4.6 Summary

In this chapter, the researcher dealt with the identification of the problem, requirements planning, and a phased approach to the development of a solution to the identified problem. Emphasis was placed on working with the users throughout all the stages, most importantly the problem identification, exploratory data analysis and demonstrations stages. The close working relationship between the researcher and the users helped to align the problem to the artefact development. The researcher described the three work streams applied during the design and deployment of the artefact, referred to as the cycles.

The first two cycles involved data analytics without the ML component. Cycles 1 and 2 unlocked the simplified and convenient data analytics capacity for incident and alarm data. Cycle 3 focused on the machine learning component and unsupervised learning with the implementation of KMeans clustering. During this process, the researcher performed the roles of: i) Business Analyst to ensure that a valid problem was identified; ii) an architectural function in order to design and plan the prototype; and iii) a developer to write the program.

Performing multiple roles presented challenges at a higher level; however, it brought value in the form of an end-to-end understanding of the study. The researcher was satisfied that the objectives of the study were met. The programs for the three developments cycles can be found in Appendix K and Appendix L and a description of the Elbow method deployment can be found in section 4.5.4.2.

CHAPTER FIVE ARTEFACT EVALUATION

The chapter flow:

- Introduction
- Evaluation model
- Evaluation approach
- The evaluation
- Summary

5.1 Introduction

Artefact evaluation is a critical stage, as it assesses the artefact developed in relation to the problem that was meant to be addressed. The artefact's main purpose was to address the problem (Baskerville, et al., 2015). The DSR approach in IS studies makes it clear that the research requires critical assessment in order to validate the outcome of the study. In this chapter, the researcher deals with the evaluation model, evaluation approach and evaluation details, as illustrated in Figure 5.1.

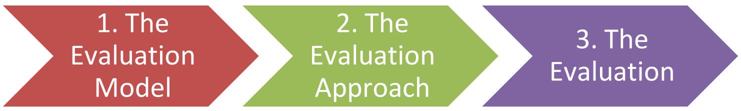


Figure 5.1: The evaluation flow

It is important that a holistic evaluation is performed to present an unbiased view of different components of the artefact. This perspective provides future researchers with informed details in order to expand on the knowledge created by this study, or for the purposes of replicating this study at any time in any part of the world.

The evaluation was done by focusing on the following dimensions:

i) **Usability:** Focusing on how the prototype works for the problem
ii) **Consumed data:** Assessment of data used
iii) **Artefact technical management:** The quality of the build (section 5.3.3.1)
iv) **Artefact foundation systems:** Underlying infrastructure used for the development

5.2 Evaluation model

Following a clear framework was important to ensure transparency in how the evaluation was performed – the model below (Table 5.1) was used for the study. This model adopts bottom-up logic and functions on the premise that the base of evaluation is the foundation plan, which needs to be clear before any evaluation takes place.

The user component is critical, as the use of the artefact focuses on solving a problem experienced by the user. Data consumed in artefact development affect the success or failure of the project and quality cannot be compromised as was done in this study. Technical management and foundation systems relate to methods and choices made in the development phase. Optimal selection of hardware specifications, such as the central progressing unit (CPU) and memory to match the programming requirement, was considered. Development efficiency is directly influenced by the IDE used by the developer.

Table 5.1: Evaluation model

Category	Evaluation Level	Evaluation	Description
Evaluation	Foundation Systems	Software	Do the hardware and software used meet development requirements?
		Hardware	
	Technical Management	Maintenance	Was there a clear structure in the development process?
		The build	
	Data	Raw data	Was the data quality reliable and applicable to the problem?
		Metadata	
	The User	Reliability	Were the user requirements met and did the solution perform in an acceptable manner?
		Operability	
		Accessibility	
		Relevance to problem	
Base	Model	Model	Was there a clear plan in how the evaluation would be performed?
	Approach	Approach	

5.3 Evaluation approach

The experimental approach was adopted for the usability dimension, and pre-defined test cases were constructed and shared with the users for execution. A control exercise was done alongside the experiment. The artefact developed (section 4.3) served the purpose of resolving the problem identified in section 4.2.1 and was used as basis of the control and experiment. A group (Team A: Control) of users within the same team was provided with the same test cases as another group (Team B: Experiment) for assessment (see APPENDIX F: EXPERIMENT). Team A carried out tests in an environment without the developed prototype and Team B worked with the prototype. This meant that Team A carried out activities in the manner in which things were done before the study (APPENDIX E: CONTROL).

Consumed data, artefact management, and foundation systems were assessed based on experience gained during the study. Since the study was undertaken in a practical environment, the researcher had little flexibility in choice of what could be used because of time and budget limitations. Choices made under these conditions were carefully assessed to confirm whether there was any valuable knowledge to be shared.

The evaluation

The evaluation was done based on four pillars, namely: i) usability; ii) consumed data; iii) artefact technical management; and iv) artefact foundation systems.

5.3.1 Usability

Usability is based on: i) the problem-to-solution alignment; ii) artefact efficiency; iii) accessibility; iv) operability; and v) reliability.

5.3.1.1 Problem to solution alignment

The problem was identified and deconstructed into three parts. Parts of the prototype were built for each of the problems. Table 5.2Table 5.2 represents the mapping of problem to solution, with user remarks.

Table 5.2: Problem-to-solution alignment

Item	Problem	Solution	User Feedback
1	Too much effort by users in manual network analysis	Prototype 1 (4.3)	Expectation met
2	Too much effort by users in performing near real-time analysis of incident management data	Prototype 2 (4.4)	Expectation met
3	Lack of automated functionality for establishing alarm patterns	Prototype 3 (4.5)	Expectation exceeded

The users were given the opportunity to assess the artefact openly with relevance to the problem, and the results showed that the requirements were met on all three components, with an indication that the third component exceeded the expectations. The solution was primarily meant for the users, and the feedback given (APPENDIX G: PROTOTYPE 1 USER FEEDBACK; APPENDIX H: PROTOTYPE 2 USER FEEDBACK; APPENDIX I: PROTOTYPE 3 USER FEEDBACK) proved beyond reasonable doubt that the project was a success.

5.3.1.2 Artefact efficiency

Test cases (Table 5.3) relative to problems identified by the users were compiled and users were requested to execute the tasks without the use of the artefact. The objective was to quantify the effort required of users when operating in a manual fashion. Table 5.3 presents the results per scenario. The granularity at scenario level gave context at the lowest level possible. The three scenarios could not be assessed in the *before* and *after* time duration state, as the user was unable to perform clustering manually prior to the artefact. It was possible to use a time duration measure for the twenty-four scenarios and to execute each scenario ranging from four to fifty-six minutes.

Table 5.3: Manual execution of tasks by user (Control)

Category	Analysis Type	Test Case	Possible	Time Taken
Alarm	Currently active	Identify top 20 active number of alarms per node	Yes	22 minutes
		Identify top 20 active number of alarms per alarm name	Yes	17 minutes
		Identify top 20 active number of alarms per supplementary information	Yes	9 minutes
		Identify the active alarms by priority	Yes	10 minutes
	Cleared	Identify top 20 cleared number of alarms per node	Yes	12 minutes
		Identify top 20 cleared number of alarms per alarm name	Yes	18 minutes
		Identify top 20 cleared number of alarms per supplementary information	Yes	11 minutes
		Identify the cleared alarms by priority	Yes	13 minutes
	Signalling route sets	Identify currently active route set	Yes	5 minutes
		Translate signalling point code to name	Yes	15 minutes
		Identify currently cleared route set	Yes	4 minutes
		Translate signalling point code to name	Yes	20 minutes
	Cellular network configuration error	Quantify active cases by reason	Yes	55 minutes
		Quantify active cases by transaction type	Yes	42 minutes
	Cellular establishment failure in user plane	Quantify active cases by reason	Yes	56 minutes
		Quantify active cases by transaction type	Yes	61 minutes
Incidents	Open Incidents	Quantify open incidents per team	Yes	9 minutes
		Quantify open incidents per person	Yes	10 minutes
		Quantify open incidents per status	Yes	10 minutes
		Quantify open incidents per priority	Yes	8 minutes
	Closed incidents	Quantify closed incidents per team	Yes	5 minutes
		Quantify closed incidents per person	Yes	4 minutes
		Quantify closed incidents per fault condition	Yes	6 minutes
		Quantify closed incidents per priority	Yes	5 minutes

Category	Analysis Type	Test Case	Possible	Time Taken
Clustering	Establish alarms by groups (patterns) based on alarm type in relation to time and severity		No	N/A
	Establish alarms by groups (patterns) based on severity in relation to time and alarm type		No	N/A
	Establish alarms by groups (patterns) based on alarm name in relation to time, severity and alarm type		No	N/A

The artefact was introduced into the operation, and tests were executed. Each of the test cases took just under 30 seconds to complete. A user only had to click the relevant tab and results were presented instantaneously.

Table 5.4: Automated execution of tasks by user (Artefact experiment)

Category	Analysis Type	Test Case	Possible	Time Taken
Alarm	Currently Active	Identify top 20 active number of alarms per node	Yes	< 30 seconds
		Identify top 20 active number of alarms per alarm name	Yes	< 30 seconds
		Identify top 20 active number of alarms per supplementary info	Yes	< 30 seconds
		Identify the active alarms by priority	Yes	< 30 seconds
	Cleared	Identify top 20 cleared number of alarms per node	Yes	< 30 seconds
		Identify top 20 cleared number of alarms per alarm name	Yes	< 30 seconds
		Identify top 20 cleared number of alarms per supplementary info	Yes	< 30 seconds
		Identify the cleared alarms by priority	Yes	< 30 seconds
	Signalling route sets	Identify currently active route set	Yes	< 30 seconds
		Translate signalling point code to name	Yes	< 30 seconds
		Identify currently cleared route set	Yes	< 30 seconds
		Translate signalling point code to name	Yes	< 30 seconds
	Cellular network configuration error	Quantify active cases by reason	Yes	< 30 seconds
		Quantify active cases by transaction type	Yes	< 30 seconds
	Cellular establishment failure in user plane	Quantify active cases by reason	Yes	< 30 seconds
		Quantify active cases by transaction type	Yes	< 30 seconds
Incidents	Open Incidents	Quantify open incidents per team	Yes	< 30 seconds
		Quantify open incidents per person	Yes	< 30 seconds
		Quantify open incidents per status	Yes	< 30 seconds
		Quantify open incidents per priority	Yes	< 30 seconds
	Closed incidents	Quantify closed incidents per team	Yes	< 30 seconds

		Quantify closed incidents per person	Yes	< 30 seconds
		Quantify closed incidents per fault condition	Yes	< 30 seconds
		Quantify closed incidents per priority	Yes	< 30 seconds
Clustering	Establish alarms by groups (patterns) based on alarm type in relation to time and severity		Yes	< 30 seconds
	Establish alarms by groups (patterns) based on severity in relation to time and alarm type		Yes	< 30 seconds
	Establish alarms by groups (patterns) based on alarm name in relation to time, severity and alarm type		Yes	< 30 seconds

After the two tests (Table 5.3 and Table 5.4), the data were comparable, which confirmed that the manual execution took 97% longer (Table 5.5) than the automated process, introducing the artefact. The user is required to use just one tool (the artefact) and obtain the results without further processing of data in a different system.

Table 5.5: Efficiency gain

#	Scenario	Before (minutes) Sba	After (minutes) Saa	Time saved by Artefact (Sd)
1	Identify top 20 active number of alarms per node	22	0.5	21.5
2	Identify top 20 active number of alarms per alarm name	17	0.5	16.5
3	Identify top 20 active number of alarms per supplementary info	9	0.5	8.5
4	Identify the active alarms by priority	10	0.5	9.5
5	Identify top 20 cleared number of alarms per node	12	0.5	11.5
6	Identify top 20 cleared number of alarms per alarm name	18	0.5	17.5
7	Identify top 20 cleared number of alarms per supplementary info	11	0.5	10.5
8	Identify the cleared alarms by priority	13	0.5	12.5
9	Identify currently active route set	5	0.5	4.5
10	Translate signalling point code to name	15	0.5	14.5
11	Identify currently cleared route set	4	0.5	3.5
12	Translate signalling point code to name	20	0.5	19.5
13	Quantify active cases by reason	55	0.5	54.5
14	Quantify active cases by transaction type	42	0.5	41.5
15	Quantify active cases by reason	56	0.5	55.5
16	Quantify active cases by transaction type	61	0.5	60.5
17	Quantify open incidents per team	9	0.5	8.5
18	Quantify open incidents per person	10	0.5	9.5

#	Scenario	Before (minutes) Sba	After (minutes) Saa	Time saved by Artefact (Sd)
19	Quantify open incidents per status	10	0.5	9.5
20	Quantify open incidents per priority	8	0.5	7.5
21	Quantify closed incidents per team	5	0.5	4.5
22	Quantify closed incidents per person	4	0.5	3.5
23	Quantify closed incidents per fault condition	6	0.5	5.5
24	Quantify closed incidents per priority	5	0.5	4.5
	Totals for the before and after the artefact	**427**	**12**	**415**
25	Establish alarms by groups (patterns) based on alarm type in relation to time and severity	Not possible	0.5	Unlocked new capability that did not exist before
26	Establish alarms by groups (patterns) based on severity in relation to time and alarm type	Not possible	0.5	
27	Establish alarms by groups (patterns) based on alarm name in relation to time, severity and alarm type	Not possible	0.5	

The results indicated that users required 427 minutes (7 hours, 11 minutes) (Table 5.5) worth of effort to perform 24 tasks. This is especially useful during shift handover. Shift handovers are processes where engineers change shifts, one incoming and the other one outgoing. A successful handover requires basic tasks to be executed, and in many cases these tasks need to be executed multiple times in a shift. A shift is 8 hours long and if the NOC engineer puts effort into performing these tasks manually, it is not possible to do much other shift work as he/she only has 49 minutes available to do the tasks.

In the formula below, Sba is *scenario before artefact* (Control), Saa (Experiment) is *scenario after artefact*, and Sd is the *difference between Sba and Saa* (Sd = Sba – Saa). The number of scenarios is represented by n; n = 24 since there are 24 scenarios. The formula to calculate the percentage efficiency gained is:

Efficiency gained (%) = $\frac{\sum_{i=1}^{n} Sd}{\sum_{i=1}^{n} Saa} * 100$ = (415 / 427) * 100 = **97%**

An efficiency of 97% was introduced by the artefact, based on one task execution run, which is the minimum level. The new capability (artefact) was introduced; however, benefit realisation could not be presented in time since there is no benchmark duration that could be used. The new capability took the NOC to a higher level with respect to handling large volumes of data in a way that directly affects positively on the day-to-day functions of the centre.

5.3.1.3 Accessibility

Users access the application through a web interface using any browser of their choice (Google Chrome, Internet Explorer, Firefox, etc.). No special installation is required, making it convenient for users. The application launches quickly. The downside of the artefact is that any user logged onto the domain of the institution can access the application, which is not ideal as the application is meant for the network operations environment only. The authentication mechanism can be improved on.

5.3.1.4 Operability

The application is not locked to a specific version of browser and does not require any additional plugin. This eliminates potential user experience problems. All contents of the application, the data extraction, cleaning, logic and algorithm are maintained from a central point, which allows the developer easy debugging of the application in cases where troubleshooting is necessary, as well as in scenarios that require managing future enhancements. The disadvantage of the development platform being centralised is that, should the server hosting the code be unavailable, then the application will not work. Considering that the study focused on a prototype, aspects of redundancy and resilience were not taken into consideration as this would form part of a deployment plan.

5.3.1.5 Reliability

The prototype was availed to users for a period of three months and was continuously available for use, with the exception of periods when planned activities took place, namely:

i) Planned installation of operating system (Windows Server 2016) security patches
ii) Planned prototype enhancements

Visualised data on the application was consistent, as all figures were presented in respective tabs at all times, as tested by the users. There were no reports of artefact failures, leading the researcher to the conclusion that the software was failure-free.

The contents of APPENDIX N: RELIABILITY confirm that there was never a case of any of the following:

i) Users being unable to access the application
ii) Application launching but not populating data
iii) Application taking long to respond
iv) Data visualised in wrong tabs
v) Tab(s) missing

5.3.2 Consumed data

Three types of data were consumed, namely: i) alarm data; ii) incident data; and iii) subject matter expert data. The users who provided input to the study had a combined work experience of 134 years, with the average work experience per employee being 6 years and 7 months (APPENDIX J: TEAM PROFILE). The combined contribution of skills and knowledge assured superior quality inputs.

5.3.2.1 Raw input data

The alarm data were well-structured. Direct access to the source data was given to the developer, which helped with flexibility during the exploratory data analysis, with the benefit of faster learning of data. Data extraction was fast and reliable.

Incident data were well-structured; however, the developer accessed the data indirectly, through a secondary system. The process of exploring data was difficult since it involved frequent approval requests, and the frequency of large-scale data extraction was limited because of the potential impact this could have on the source data system. Lack of direct access to source data was a challenge; however, the quality of the data processed was never compromised at any point.

5.3.2.2 Metadata

Subject matter experts (SMEs) were used because of their domain knowledge with the objective of helping the researcher understand the descriptions of raw data and metadata. Shared rich data made it possible to create meaningful dictionaries, which were key to saving time spent on manual activities. The data provided by SMEs were of high quality and proved to be key to the value added to the visualised data. The challenge with the metadata was that it was not stored in a structured way; it was pulled from specialised documents hosted on the local machines of Level 2 engineers. A knowledge management database would have been ideal; however, that being said, the support of domain knowledge would still be required.

5.3.3 Artefact technical management

5.3.3.1 The build

The solution was planned and designed using detailed low level and high level designs (sections 4.3.4 and 4.4.4). A structured planning approach was followed. User requirements were divided into three categories and the development was planned and implemented in three cycles. The process enabled a focused approach with aligned collaboration with the users. The delivery in phases (section 4.2.3) allowed the developer to incorporate user feedback into the iterations. Each iteration had a clear objective and was concluded with a working prototype for the users.

Python was used as the programming language for the project and proved to be efficient in the build. The main advantages of using Python were:

i) **Simplicity:** Easy to write and read programs. This level of simplicity was helpful in avoiding syntax-related mistakes in the code writing and reviewing stages.

ii) **Functionalities:** Python has a wide range of libraries that were helpful in implementing features fast and reliably. Pandas, Seaborn and NumPy were key libraries used for the analysis of data as well as visualisation of the data during unit testing. Bokeh was used for visualisation of data in the prototype. Scikit-learn was used for the ML component.

5.3.3.2 Maintenance

The development was done in the Anaconda environment, and during the process of the project, multiple Anaconda, Jupyter Notebook and Spyder updates were installed. The developer planned the time to install updates carefully and performed testing of the application after installation. Post-project maintenance is not taken into consideration, as this forms part of operations.

5.3.4 Artefact foundation systems

5.3.4.1 Hardware resources

Windows Server was used for the project and the CPU and memory capacity was scaled to meet the project requirements. The scaling was based on initial data quantification, and worked well. The application is still working as desired even though the developer leant that there were other activities performed by other developers on the same server, resulting in reducing the capacity. With data volumes continuously increasing, it is not ideal to have a shared environment since this will affect the performance of the applications going forward. The most resource-intensive part of the artefact is the ML part (i.e. running the KMeans algorithm). The most appropriate way to demonstrate hardware capacity was for the resource-intensive part, as it gave the worst-case scenario. The maximum CPU utilisation (Figure 5.2) reached was 80% and for a brief period, the usage averaged at 50%, allowing room for increased usage should the need arise.

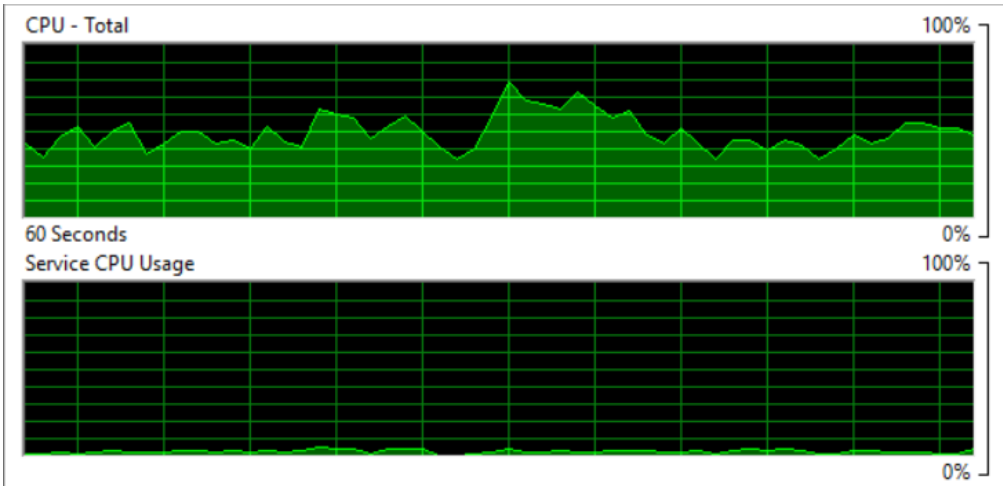


Figure 5.2: CPU usage during KMeans algorithm

Memory usage averaged at 40% (Figure 5.3) for the duration of the cycles and, with 32GB capacity, memory performance was as expected.

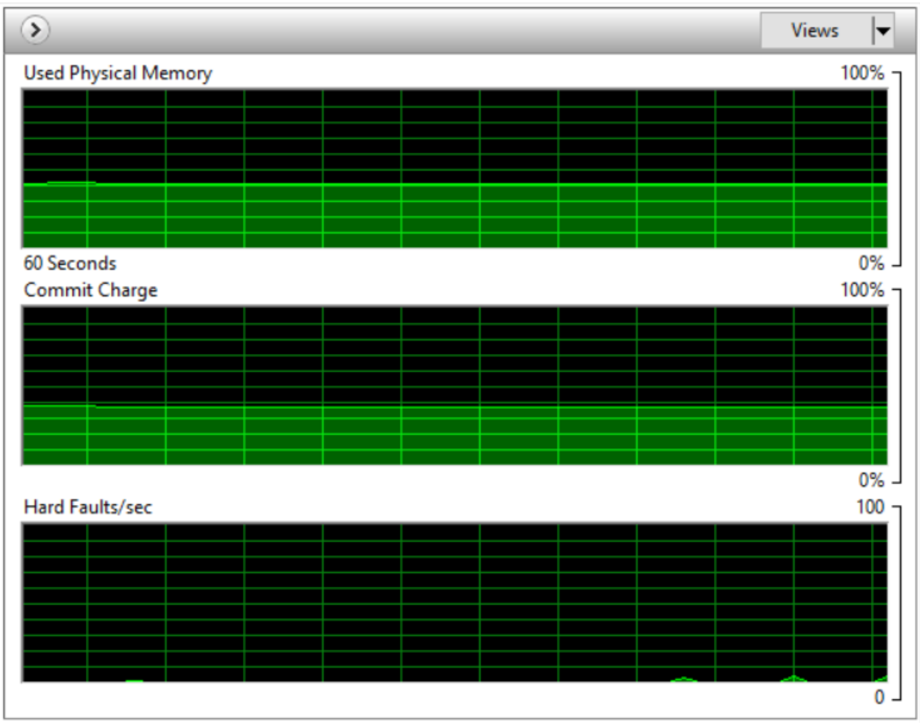


Figure 5.3: Memory usage during KMeans algorithm

The hardware resources were allocated to meet the development requirement, as there was no case of these resources running at their maximum levels. The impact of under-specification of the resources would have resulted in poor performance of the application layer.

5.3.4.2 Software resources

The open source applications Jupyter Notebook and Spyder were used from the Anaconda Navigator. The applications were free and a big advantage. Abundant support from internet forums was available. The use of proprietary software in the form of the Microsoft Windows Operating System (OS) involved initial and running costs, such as costs for licenses and upgrades. OS is commonly understood by many technical and non-technical users around the world, making this study easily replicable by a large number of people at different knowledge levels. The use of Jupyter Notebook enabled fast code deployment and testing on the same platform, saving development time, running Python 3 (the latest version). The overall landscape is illustrated in Figure 5.4.

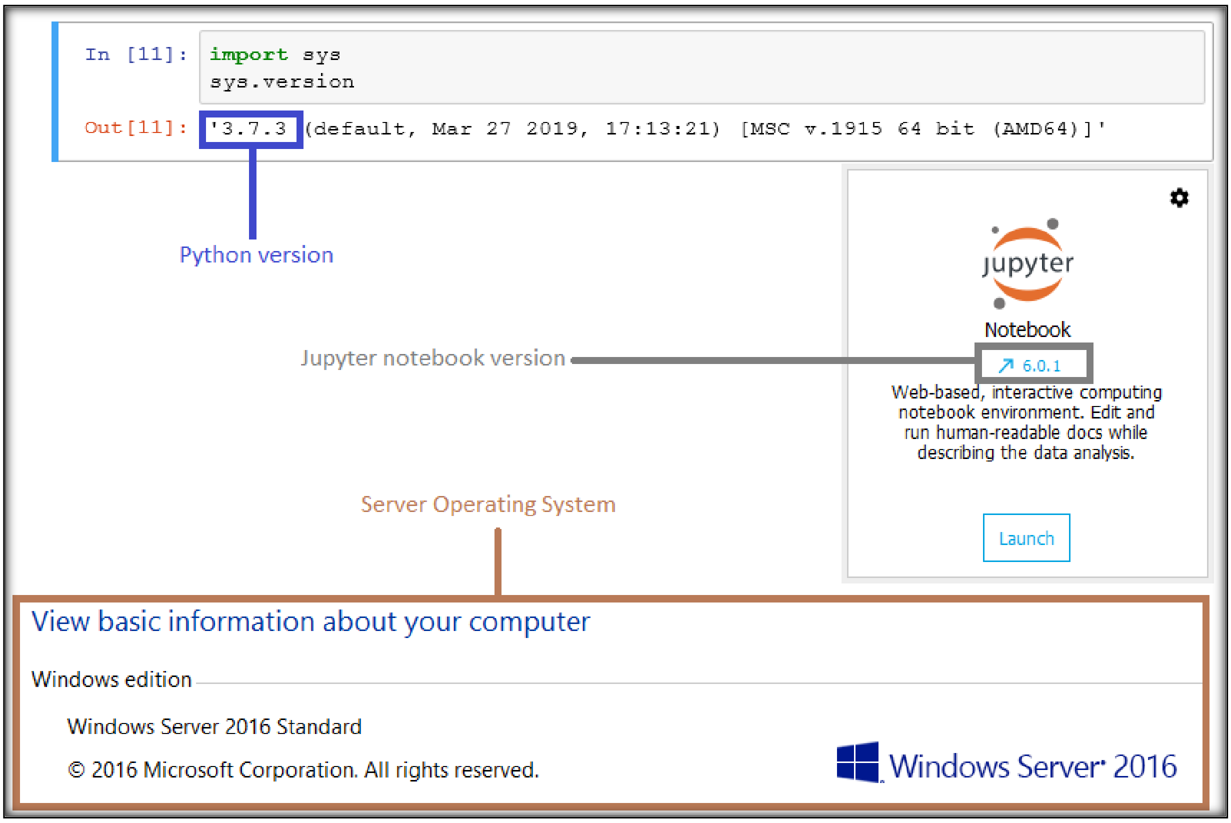


Figure 5.4: Software landscape used in artefact development

The use of the latest tools available for the development of the artefact was important to avoid complications of tools reaching end-of-life, affecting the progress made.

5.4 Summary

A multi-dimensional evaluation was put in place to assess the user-facing and back-end components. Critical evaluation was performed, indicating aspects that went well and those that did not go well. The overall results of the artefact were that the resources used, the method followed, and the satisfaction of the users all confirmed success. The artefact did solve the problem identified by the users. Table 5.6 summarises what was covered during the evaluation of the artefact.

Table 5.6: Evaluation Summary

Category	Evaluation Level	Evaluation	Description
Evaluation	Foundation Systems	Software	Hardware and software did align to the development requirement.
		Hardware	
	Technical Management	Maintenance	A clear building process that meets academic and industry standards was used
		The build	
	Data	Raw data	Data quality and consistency was very high.
		Metadata	
	The User	Reliability	The artefact did meet the user requirement (the problem); it was fault free and providing clear efficiency was gained.
		Operability	
		Accessibility	
		Relevance to problem	
Base	Model	Model	The evaluation plan was clear and detailed.
	Approach	Approach	

The researcher's interpretation of the evaluation results continues in Chapter Six.

CHAPTER SIX: DISCUSSION

The chapter flow:

- Introduction
- Interpretation of evaluation results
- Comparison with prior work
- Key learnings
- Limitations
- Future work
- Summary

6.1 Introduction

The study objectives were stated in Chapter One, while the development and evaluation of the artefact was described in Chapters Four and Five, respectively. In Chapter Six, the researcher covers the meaning of the results of the artefact in relation to the objectives of the study. Details on key learning, comparison of the study to prior work, limitations, relevance of this study to the existing body of knowledge, and recommendations for future work are covered by the researcher in this chapter. The study demonstrated knowledge gained and applied. The use case is included to provide the context in which to discuss the work covered. The DSR structure of the thesis was applied as recommended by Gregor and Hevner (2013).

6.2 Interpretation of evaluation results

The artefact addressed the problems identified in the study. The results from the evaluation chapter, together with the reviewed literature, enabled the researcher to derive meaning from the results.

The artefact resulted in a real business benefit; a benefit far greater than the NOC engineers' productivity. A 97% reduction in the time taken to execute tasks meant that NOC work output increased. The introduction of the artefact meant that it became easier for users to use the developed information system artefact (sections 4.3 and 4.4), which improved consistency in the management of faults. With manual and mundane activities removed, the users were best positioned to focus on innovative tasks. High volumes of mundane activities caused underutilisation of the talents of the NOC employees. The use of the artefact allowed users to obtain the information they required much faster, thereby enabling them to move onto addressing other issues. The perspective from a human resources view is that limited time spent by employees on manual activities allows them to do more work, thus increasing their work output. The customer was, therefore, served better with the use of the artefact in the NOC.

6.2.1 Contextualising results

A route set unavailable alarm is received when all the control plane links to a mobile switching server (MSS) failed. If the links between the MSS and a radio network controller (RNC) are down, it means the 3G voice service is down for customers. In the past, it would take the NOC five minutes to identify active route set conditions and an additional fifteen minutes to decode RNC alarm parameters to human readable names. The NOC took 20 minutes to identify the element impacted service, which was too long. Only after 20 minutes was the relevant person, such as an RNC engineer, made aware to resolve the fault. The inability of subscribers to make calls in the 20 minutes means that the customer experience gets degraded and the mobile operator is unable to make revenue because of the fault.

The artefact made it possible for the NOC user to simply click on the prototype artefact and receive information identifying an active route set unavailable alarm with the name of the RNC presented in a human readable format, all within one minute. The user did not need to know which man-machine-language command to execute from the MSS. It would take the user 15 minutes manually to transform alarm data, in numerical format, to simplified information readable by humans, which reduced the user's efficiency. The developed information system (Figure 6.1) consumed data to present information to the user in order to assist in solving a business problem.

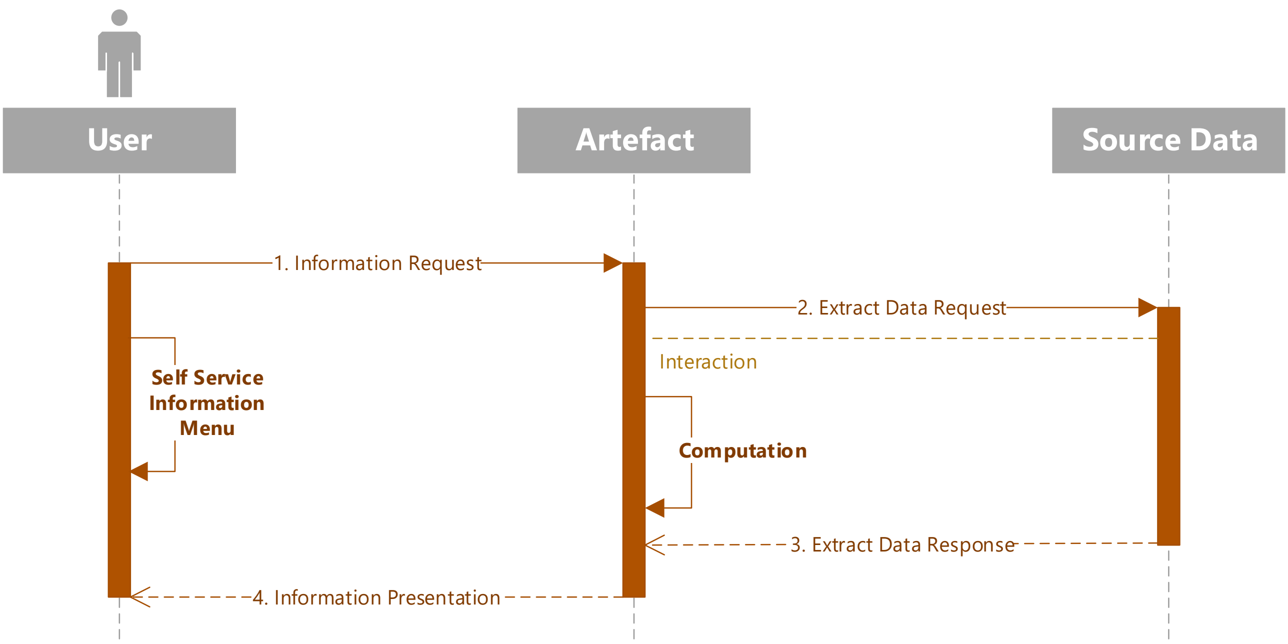


Figure 6.1: Information system artefact serving NOC requirement

6.2.2 Conclusion

The technology industry, and more specifically the telecommunications industry, will continue to change to meet global communications needs. Technology changes are not dependent on the MNOs. Technological changes have a direct impact on NOCs and other technical teams within the MNOs. The illustration below (Figure 6.2Figure 6.2) was designed from lessons learnt from this study. Business (in this case, the MNO) will build a strategy to respond to the market (including customer and industry changes) to remain relevant. Network engineering teams will plan and build the network, and the new components introduced will generate new sets of alarms (adding to what already exists), which means the methods used to manage the network will become more complicated.

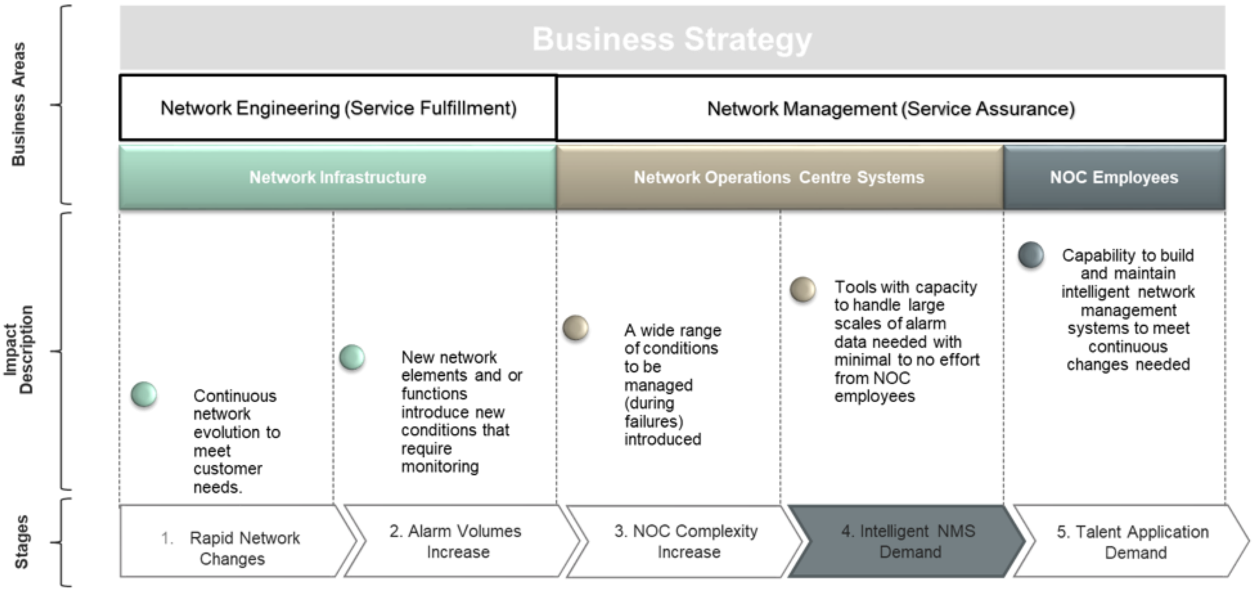


Figure 6.2: Continuous operational impact on NOCs

Large alarm and incident data are observed in the NOC and intelligent applications such as the prototype presented by this study will be needed. The option of recruiting more people to the NOC so as to manage the load is not a suitable solution because of the cost implications. Intelligent NMSs, of which this particular prototype is an example, are the way to absorb the increased workload. It takes skilled NOC people to design, build and maintain these intelligent NMSs. Efficiency brought about by advanced IS applications makes it possible to run an NOC on a day-to-day basis with fewer people, while reskilling others to design, build and maintain information systems.

6.3 Comparison to prior work

6.3.1 Rule set automation

The trend of increasing network alarms was established and classified as a risk over a decade ago (Bandini et al., 2006). Rule set based algorithms were proposed as the suitable solution at the time. Tabia and Leray's (2011) rule set based algorithms focused on alarm correlation and re-emphasised that the solution is valid, both then and in the future. Thirteen years after the study of Bandini et al. (2006), alerts automation solutions, using delay and correlation (Salah et al., 2019), were developed in order to enable NOCs to manage high volumes of data. The views of Salah et al. (2019) were limited to unique events observed in isolation of a complete picture of what is done in an NOC. The rule set based algorithms were an interim solution that helped to remediate the situation. Knowledge gained in this study enabled the researcher to apply critical thinking to mainstream rule set based algorithms and the impact they have on the NOC and business.

Kibria et al. (2018) argue that challenges experienced by NOCs as a result of large alarm volumes are addressed in next generation network (NGN) AI, and it is irrelevant that MNOs, like the one studied, still have a large investment in legacy technologies such as GSM and UMTS. These legacy technologies and the NGN infrastructure need to be managed simultaneously and, should the approach of Kibria et al. (2018) be adopted, the service assurance processes for legacy technologies will collapse, which is not an option.

6.3.1.1 Correlation

Alarm correlation is the concept of reducing the number of alarms displayed on the monitoring view by applying the knowledge of SMEs in order to establish patterns of alarms caused by the same cause. This principle presents one alarm, which is understood to be the main alarm, and hides other alarms related to it from the view. The value of the algorithm was that fewer alarms were displayed on the screen; however, the shortcomings of this algorithm lie in the fact that the insights that could have been gained from the hidden alarms and used by business are lost.

6.3.1.2 Delay

Alarm delay is the concept of configuring a time parameter that will hold selected alarms for a period of time before presenting to the monitoring view. The benefit, which was both short-term and risky, was a reduction in the number of alarms displayed on the monitoring view, while the main disadvantage was that the NOC is never aware of the key conditions that are frequently active for short durations. This method is dangerous and it encourages the NOC to look away from what has happened and what is possibly still happening.

6.3.1.3 Suppression

Network vendors in the process of building the equipment to be used in the network should ensure that the capability to indicate to MNOs is enabled in order to help detect any anomalies in the network should the equipment malfunction. The concept of suppression of alarms means permanently removing selected alarms from the NOC's visibility. The concept here is clear – the team that understands the equipment best, created a condition to be monitored which the MNO decides to ignore. Applying suppression of alarms helps to reduce the number of alarms handled.

6.3.2 Prior work – the gap

The above three rule set based algorithms (sections 6.3.1.1, 6.3.1.2 and 6.3.1.3) bring short term benefits to the NOC user, at the cost of losing important information that is required to understand the state of the network. Valuable data is one of the main factors that give businesses a competitive advantage. Rule set based algorithms hide data and are used in an attempt to make current technology, which uses outdated processes and tools, manageable. With an increase in data in NOCs, the users viewing a screen which displays alarms presented line-by-line is not practical, as confirmed in the artefact evaluation phase. An increase in alarm volumes will continue to happen and the NOC cannot influence this otherwise.

6.3.3 How does the prototype compare to prior work?

The prototype is consuming all available NOC data (no discarding or hiding) to generate new information that is adding value to the operation. Fault analysis time is reduced and very little effort is expected from the users when compared to the original way of work. The value-add brought about by the artefact enabled the MNO to receive the actual state of the network information (both active and cleared conditions), thereby allowing for higher quality decisions to be made. The simplicity of use of the artefact from the user's point of view allowed complex tasks to be executed by junior personnel. Overall, the limited and dangerous benefit brought by rule set based algorithms is not anywhere close to the data analytics and ML powered applications. Rule set based algorithms do not automatically compute the patterns of alarms by grouping (clustering), and this was introduced by the artefact. The new introduced capability enables the MNO to understand the behaviour of alarms to allow informed decisions to be taken. The prototype is in alignment with modern technologies associated with data science and ML.

6.4 Key learnings

6.4.1 The use case

The way data are consumed in network operations is very important and, if done efficiently, the benefits are realised from the NOC level to the mobile subscriber level. A review of the way in which NMSs are deployed can be improved by introducing a new layer that focuses on simplifying the consumption of data. The researcher experienced a challenge by taking long to understand the context of the NOC and the problem was that the stakeholders (Level 1 engineers) did not have all the information to help. The challenge was addressed when the researcher became transparent to the stakeholders about the gaps (i.e. missing data) in the information provided and so the subject matter experts were recommended to the researcher by Level 1 engineers.

6.4.2 Research process

DSR research for a software artefact challenged the researcher's capability on four levels:

i) **Interpersonal level:** The researcher had to interact with the people (users and system owners for data sources). It was not easy to obtain optimal participation from the people at the initial stages of the project, as the researcher was seen to be an outsider. The situation improved as relationships matured.

ii) **Cognitive level:** Once the problem was identified, it took a great deal of knowledge gathering and knowledge application to work out the solution. Research from past years helped the researcher to gain in-depth knowledge of prior work done and an understanding of existing technologies applicable to the study. Essential to a research-worthy problem is that it must be a problem that was never solved before, or it must deal with an improvement on past studies, meaning that the researcher will deal with unknowns. Cognitive levels were built by studying relevant peer-reviewed material, leading to a quality solution. The processes of knowledge gathering were done at a slow pace in the initial stages, which impacted on the time taken to develop a concept for the proposed solution. Daily one-hour reading sessions was scheduled and honoured, which reduced the challenge.

iii) **Artefact development:** Research-worthy problem identification is the most important part of DSR and everything else builds on that step. The problem needed to be understood clearly by the researcher in order to get to a solution that matched the problem.

iv) **Version control** in the programming stage is very important as the researcher happened to keep multiple versions of the program saved. The researcher found it very difficult at the initial stages to know which version was the most recent program being worked on.

v) **Thesis writing:** Writing was deferred during problem identification and literature review phases of the research, as the researcher felt at the time that is was too early to begin writing. That led to a backlog in writing that had to be done and the writing process proved to be difficult as studied material had to be re-read to refresh the researcher's memory. A decision was made resulting from the painful experience that writing should be done parallel with other activities and that, once an activity is completed, the written material should be reviewed. Deferring the writing was a big mistake.

6.5 Relevance to the body of knowledge

The study highlighted gaps in the current NOC operating model, that is, the model was not taking full advantage of all the available data. The current operating model is weakened by what was previously understood and implemented as workload management remedial action. Looking at the bigger picture, rule set based automation had major limitations and did not consider an end-to-end view of service assurance.

The new model proposed (Figure 6.3), as guided by the developed artefact, provided sufficient reasons to phase-out rule set based automation engines and to deploy the NOC advanced Information System (NOCaIS), driven from the NOC-user experience point of view. As in the prototype, an open source toolset is capable of making this transition possible. IEEE has done good work in setting standards in ICT and continues to do so; taking on this new model to formalise in the industry will benefit MNOs globally.

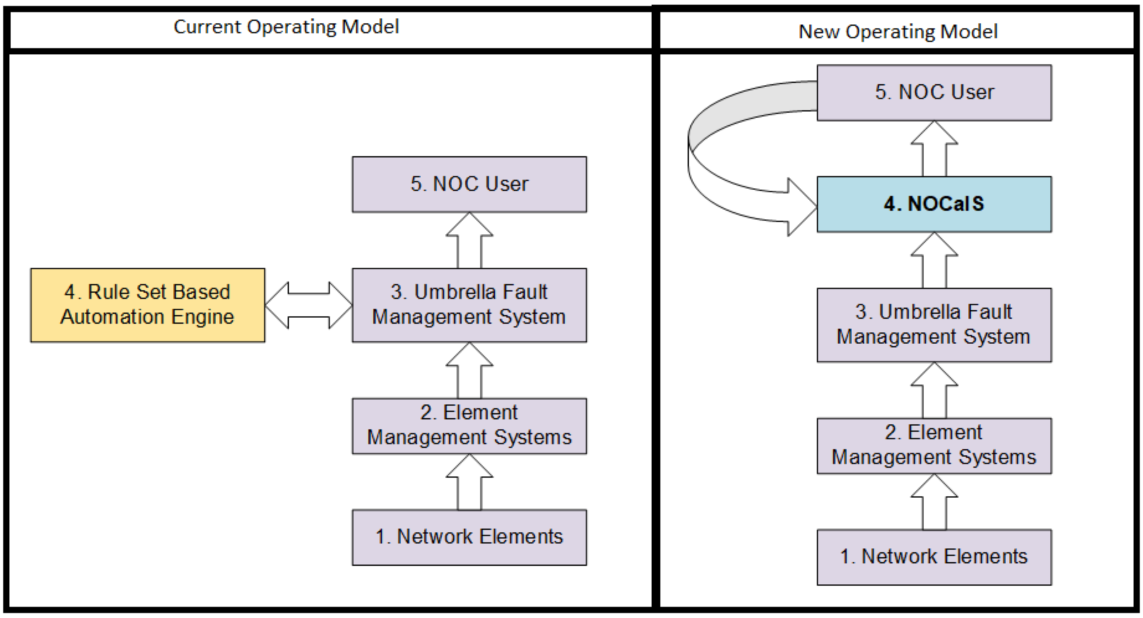


Figure 6.3: Current and new NOC operating models

The adoption of the new operating model is cost efficient (use of open source) and hardware used for the rule set based automation engine can be repurposed for NOCaIS.

6.6 Limitations

6.6.1 Scope and time

The research focused on the Core Network of a MNO in South Africa. Radio and transport networks were excluded from the research. The objective of the research was to present a working prototype. Technology differences are wide between the Core, Radio Access and Transport technologies; therefore, in-depth knowledge would not have been achieved if all areas of the network were scoped and the possibility of meeting the time allocated for a Master's degree would be unlikely.

6.6.2 Research process

Only literature in the English language was reviewed, since this is the language understood by the researcher. The environment studied operates all hours of the day, meaning that the employees work in shift cycles. This made it impossible for all team members to be present at any one time during contact sessions.

6.7 Future work

A working prototype artefact was produced in this study and, considering the limitations stated in section 6.6, there is need to continue the study in the Radio Access and Transport domains in order to provide a complete picture across all technologies used. User interface design (UX) is a field in its own right, which involves highly specialised developers. The development in this study did not focus on building a superior user interface. The application-user experience was directly affected by the user interface (UI). Network monitoring requires users to interact constantly with the application and a well-researched and built UI would have far reaching outcomes.

Only one aspect of unsupervised ML (clustering) was explored in this study, leaving a wide range of opportunities in other unsupervised leaning, reinforcement and supervised learning areas. For doctoral studies, whenever possible, the study should be undertaken to investigate a standard NOC framework for ML.

6.8 Summary

This chapter assisted the researcher with describing the problem and solution to the use case, and this was taken to the lowest level possible for the main processes, namely the incident and event manage processes. The artefact evaluation results were interpreted by the researcher to establish meaning, and the scenario was unpacked to highlight the clear differences in NOC operations before and after implementation of artefact. Prior work studied presented value that was adopted; it also raised arguments which the researcher described. The value of rule set based automation, determined from prior work, was confirmed as depriving a business of competitive advantage, which led to the development of the prototype in this study. The advantages of the prototype surpassed rule-based automation in cost and human resources efficiency. The researcher experienced challenges in undertaking the process of the study and managed these challenges as described in the key learnings. The study has made a contribution to the body of knowledge – a new NOC operating model called the NOC advanced information system (NOCaIS). The study was conducted for academic purposes and its contribution to the body of knowledge is important. Limitations of the study and proposed future research were also covered in this chapter.

CHAPTER SEVEN: CONCLUSION

The chapter flow:

- Introduction
- Research objectives
- Research questions
- Hypothesis
- Reflections
- Conclusion
- Summary of the study
- Summary

7.1 Introduction

In this chapter, the researcher draws conclusions with reference to what was described as the focus of the study and the actual work covered during the study. The content makes specific reference to the research objectives, questions, aim and hypothesis of the study. Key findings from the study are restated. Reflections on the researcher's experience are detailed. This is the final chapter of the study, which makes it critical to prove that the goal of the study set in Chapter One has indeed been achieved.

7.2 Research objectives

7.2.1 Objective 1

The first objective of this research was to propose a data analytics solution designed to reduce manual network analysis. The development of the solution, described in section 4.3.5, and its function, demonstrated in section 4.3.6, addressed Objective 1, while the implementation results, presented in section 5.3.1.2, confirmed that this objective has been met. Users were spending long periods of time undertaking manual network analysis, specifically for this objective, on alarm category determination, as per Table 5.3. Sixteen scenarios that took a user 375 minutes to accomplish, were achieved in eight (8) minutes with the use of the artefact.

7.2.2 Objective 2

The second objective of this study was to propose a near real-time analysis solution for incident management. The development of this solution, described in section 4.4.5, and the demonstration of the solution, described in section 4.4.6, addressed Objective 2 and results presented in Table 5.4 confirmed that the objective has been met. Users were spending long periods of time on manual analysis of incidents, specifically for this objective, on incident category determination as per Table 5.3.

Eight (8) scenarios that took users 57 minutes to accomplish were achieved in four (4) minutes with the use of the artefact.

7.2.3 Objective 3

The capability to identify groupings for alarms did not exist prior to the study. This final objective of the study was to propose an automated smart grouping artefact for managing alarm patterns. The development, described in section 4.5.4 and demonstrated in section 4.5.5, addressed Objective 3, and results presented in section 4.5.6.3 confirmed that the objective has been met. The capability to identify alarm groupings was developed through the artefact and made available to users.

7.3 Research questions answered

7.3.1 Research Question1

RQ1: What is the relationship of alarm volumes to network monitoring functions with reference to time taken to action?

The number of alarms flowing into the NOC requires the NOC user to analyse these alarms, as it is this analysis that enables an understanding of the root cause of alarms. The user handles each alarm individually, as the cause for each alarm is defined by unique parameters. In this case the parameter is the summary field as per Figure 4.3, which is not in a human-readable format. High alarm volumes flowing into the NOC means users require more time to identify root causes through an analysis of the data. Resolution actions cannot start if the cause of fault is not understood. The manner in which fault analysis was performed meant there was a delay between the time the alarms reached the NOC and the time when resolution actions were taken. The success of meeting Objective 1 of the study helped to close this gap in time.

7.3.2 Research Question 2

RQ2: Does the implementation of a data analytics application bring insights to NOC engineers with minimal effort required of them?

The question was answered by the work undertaken, which is covered in section 5.3.1.2, as the prototype confirmed 97% efficiency (time saving). By using the prototype, users did not need to switch between applications in order to extract and translate information. Prior to the artefact implementation, incoming alarms were manually selected from a list, which did not present a clear picture of all events on the network. The artefact extracts, translates and visualises information across the network, and users gain insights into the network situation by simply observing information on a web-based application, i.e. the artefact.

7.4 Hypothesis

The data analytics solution increased productivity in the NOC by automating the alarms analysis process.

This confirmed the hypothesis to be true, with a clear efficiency gain introduced by the analytics solution. The solution enabled users to do more work in less time.

7.5 Reflections

Interacting with users in the organisation who were working in the area of study was a privilege that came with challenges, as helping with the study was not their only focus. Helping to identify a real problem that these users experienced on a daily basis enriched the relevance of the study. The inability of the participants to all be available at the same time for contact sessions introduced complexity into the study. This could not be changed since they worked a rotating shift cycle. The willingness of the participants to assist was good motivation for the researcher. Data security is taken very seriously in the organisation where the study was performed, which led to long delays in granting permission to the researcher to access and consume data; however, the reputation of the academic institution (Cape Peninsula University of Technology) where the researcher was from helped to ease the process.

The research supervisor (Dr AC de la Harpe) had a clear understanding of the academic technicalities and demonstrated great leadership. The supervision process involved criticism that was meant to provide sound direction to the researcher. The programming part of the study proved to be complicated at the initial stage and the combination of reading books and support from online forums helped.

7.6 Conclusion

Network Operations Centres handle large volumes of data, specifically alarms and incidents data. Legacy ways of work involving manual human-driven fault analysis, do not meet current requirements to manage faults effectively. Successful implementation of data analytics and machine learning is a sustainable way for mobile network operators to meet service assurance targets. The use of open source platforms makes it feasible to deploy information systems, such as the one demonstrated in this study. Business does not operate on an unlimited budget and the rate of data growth has been rapid; this will increase further going forward. Implementation of technology to improve customer service and workplace conditions is the best way for a business to remain competitive in the market. The telecommunications sector is essential to the economy of the country as it enables the establishment of other industries and improves the lives of citizens. This study linked the academic and telecommunications fields through information systems by delivering a working prototype ready to be adopted and that can be developed further to customer specifications. The literature reviewed provided in-depth knowledge of how NOCs operate, as well as of the technology used in this space.

7.7 Summary of the study

DSR was applied in this study; problem identification formed part of the study and involved users. The identified problem led to the formulation of research objectives and the setting of research questions. Literature was reviewed, as discussed in Chapter Two, which gave the researcher a good idea of what research had been conducted in the past to address the problem. In order to fulfil academic requirements, research methodology is one of the key components that a researcher needs to understand well. This requirement led to an in-depth study of DSR, as described in detail in Chapter Three. The application of this methodology formed the foundation for how the research was planned, performed and evaluated. The development and implementation of an artefact was the output of the DSR process and the researcher provided a concise description of the artefact in Chapter Four. The artefact could only be confirmed successful if it solved the problem identified, which increased the seriousness and value of the evaluation of the artefact. This was addressed by the researcher in Chapter Five. Understanding the context of the results from the evaluation helped to relate the prototype to prior work done by the NOC, and the significance of the solution was discussed in Chapter Five. The study started with a problem being identified, after which the objectives and questions were set. In the final chapter, the researcher established whether the objectives of the study were met and the research questions answered, and summarised the researcher’s experience.

7.8 Conclusion

This chapter linked the aim, objectives and questions of the findings together. The objective to propose a data analytics solution for reducing manual network analysis was met; the objective to propose a near real-time analysis solution for incident management was met; as was the final objective of proposing an automated smart grouping artefact for alarm patterns. The research questions related to the established objectives were answered in sections 7.3.1 and 7.3.2. The answers to the research questions indicated that the higher the alarm volumes, the more effort is required from NOC users.

The prototype brought value to the users who did not have the ability to view the network status in human readable format at a click on a web-based application prior to the introduction of the prototype. The complexity of having to interrogate a network element manually was completely eliminated, thereby enabling low skilled employees to access the same level of information as skilled employees can access. The hypothesis was accepted since the evidence confirmed that employees were now able to execute more tasks in a fraction of the time by using the prototype.

The main benefit of this prototype is the improvement of customer experience by simplifying complicated telecommunications assurance processes. The cost effectiveness of deploying this solution is an aspect that stands out and that will make sense to leadership teams in the engineering, human resources and customer facing environments. The recommendation is that this prototype be adopted by the Core Network Operations Centre.

REFERENCES

3GPP. (2013). *3GPP TR 32.859 V12.1.0 (2013-12): 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; Study on Alarm Management (Release 12).* Valbonne, France: 3GPP.

Al-Naymat, G., Al-Kasassbeh, M. & Al-Hawari, E. (2018). Using machine learning methods for detecting network anomalies within SNMP-MIB dataset. *International Journal of Wireless and Mobile Computing*, 15(1):67–75. doi:10.1504/IJWMC.2018.10015860.

Ali, A.M.A. & Alshibani, A.M. (2018) Analysis of telecommunication companies' data mining techniques in Philippines: A case study of globe telecom. *International Journal of Current Research*, 10(10):74117–74123. doi:https://doi.org/10.24941/ijcr.32520.10.2018.

Alusha, D., Analyst, S. & Saadi, M. (2018). *Artificial Intelligence and Machine Learning Use Cases in Telecoms*. Allied Business Intelligence. Available at: www.abiresearch.com. [Accessed: 19 November 2019].

Aye, T.T., Kee, G., Lee, K., Su, Y., Zhang, T., Lee, C., Kasim, H., Hoe, I., Lee, F., Gih, T. & Hung, G. (2017). Layman analytics system: A cloud-enabled system for data analytics workflow recommendation. *IEEE Transactions on Automation Science and Engineering*, 14(1):160–170. doi:10.1109/TASE.2016.2610521.

Ayoubi, S., Limam, N., Salahuddin, M.A., Shahriar, N., Boutaba, R., Estrada-Solano, F. & Caicedo, O.M. (2018). Machine Learning for Cognitive Network Management', *IEEE Communications Magazine,* 55(1):158–165. doi:10.1109/MCOM.2018.1700560.

AXELOS.com. (2019). *ITIL Foundation Glossary ITIL V4.* Available at: https://purplegriffon.com/downloads/resources/itil4-foundation-glossary-january-2019.pdf. [Accessed: 15 June 2020].

Bandini, S., Mosca, A. & Palmonari, M. (2006). Intelligent alarm correlation and abductive reasoning. *Logic Journal of the IGPL*, 14(2):347–362. doi:10.1093/jigpal/jzk023.

Barron, J., Crotty, M., Elahi, E., Riggio, R., Lopez, D.R. & De Leon, M.P. (2016). Towards self-adaptive network management for a recursive network architecture. *Proceedings.* 2016 IEEE/IFIP Network Operations and Management Symposium (NOMS 2016), Spain, 23 August, 1–6. doi:10.1109/NOMS.2016.7502977.

Baskerville, R.L., Kaul, M. & Storey, V.C. (2015). Genres of inquiry in design-science research: Justification and evaluation of knowledge production. *MIS Quarterly: Management Information Systems*, 39(3):541–564. doi:10.25300/MISQ/2015/39.3.02.

Baskerville, R.L., Baiyere, A., Gregor, S. & Hevner, A. (2018). Design Science Research contributions: Finding a balance between artefact and theory. *Journal of the Association for Information Systems,* 19(5):358-376. doi:10.17705/1jais.00495.

Baştanlar, Y. & Özuysal, M. (2014). Introduction to machine learning. *In* Yousef, M. & Allmer, J. (eds.), *miRNomics: MicroRNA Biology and Computational Analysis. Methods in Molecular Biology (Methods and Protocols),* 1107. Totowa, NJ: Humana Press, doi:10.1007/978-1-62703-748-8_7.

Boutaba, R., Salahuddin, M.A., Limam, N., Ayoubi, S., Shahriar, N., Estrada-Solano, F. & Caicedo, O.M. (2018). A comprehensive survey on machine learning for networking: Evolution, applications and research opportunities. *Journal of Internet Services and Applications,* 9(16):1–99. doi:10.1186/s13174-018-0087-2.

Brady, D. (2000). Why Service Stinks. *BusinessWeek.* October 23, 2000, 6:00 AM GMT+2. https://www.bloomberg.com/news/articles/2000-10-22/why-service-stinks. [Accessed: 15 June 2019].

Broussard, M. (2016). Big data in practice. *Digital Journalism*, 4(2):266–279. doi:10.1080/21670811.2015.1074863.

Bruno, T. & Abrahão, J. (2012). False alarms and incorrect rejections in an information security centre: Correlation with the frequency of incidents. *Work*, 41(Suppl 1):2902–2907. doi:10.3233/WOR-2012-0542-2902.

Brynjolfsson, E. & Mitchell, T. (2017) What can machine learning do? Workforce implications. *Science*, 358(6370):1530-1534. doi:10.1126/science.aap8062.

Burgess, A. (2018). *The executive guide to Artificial Intelligence*. 1st ed. Palgrave Macmillan. doi:10.1007/978-3-319-63820-1.

Caldas, M.P. (2009). *Research design: Qualitative, quantitative, and mixed methods approaches*. 5th ed. Sage. doi:10.1590/s1415-65552003000100015.

Casey, M. (2018). *How to train your network: The role of Artificial Intelligence in network operations*. Available at: https://www.itproportal.com/features/how-to-train-your-network-the-role-of-artificial-intelligence-in-network-operations/. [Accessed: 15 May 2019].

Cater-Steel, A., Toleman, M. & Rajaeian, M.M. (2019). Design science research in doctoral projects: An analysis of Australian theses. *Journal of the Association for Information Systems*, 20(12):1844–1869. doi:10.17705/1jais.00587.

Cherrared, S., Imadali, S., Fabre, E., Gössler, G. & Yahia, I.G.B. (2019). A Survey of fault management in network virtualisation environments: Challenges and solutions. *IEEE Transactions on Network and Service Management*, 16(4):1537–1551, December. doi:10.1109/TNSM.2019.2948420.

Choi, T.M., Wallace, S.W. & Wang, Y. (2018). Big data analytics in operations management. *Production and Operations Management*, 27(10):1868–1883. doi:10.1111/poms.12838.

Da Silva, H.C.C., De Oliveira Siqueira, A., Araújo M.A.V. & Dornelas, J.S. (2018). Let's be pragmatic: Research in information systems with relevance and rigor. *International Journal of Business Management and Economic Research,* 9(4):1314–1321.

Das, S.B., Mishra, S.K. & Sahu, A.K. (2018). Alarm colouring and grouping algorithm for root cause analysis. *Journal of King Saud University – Computer and Information Sciences:* 1–7. doi: 10.1016/j.jksuci.2018.04.005.

Davenport, T.H. (2018). *The AI advantage: How to put the Artificial Intelligence revolution to work*. MIT Press.

Department of Justice and Constitutional Development (1996). *The Constitution of the Republic of South Africa.* doi:10.1017/S0021855300011499.

Dorton, S.L. & Liu, D. (2016). Effects of baggage volume and alarm rate on airport security screening checkpoint efficiency using queuing networks and discrete event simulation. *Human Factors and Ergonomics in Manufacturing*, 26(1):95–109. doi:10.1002/hfm.20616.

Feilzer, M.Y. (2010). Doing mixed methods research pragmatically: Implications for the rediscovery of pragmatism as a research paradigm. *Journal of Mixed Methods Research,* 4(1):6–16. doi:10.1177/1558689809349691.

Feng, M., Zheng, J., Ren, J., Hussain, ., Li, X., Xi, Y & Liu, Q. (2019). Big data analytics and mining for effective visualization and trends forecasting of crime data. *IEEE Access*, 7:106111-106123. doi:10.1109/ACCESS.2019.2930410.

Fullen, M., Schuller, P. & Niggemann, O. (2017). Defining and validating similarity measures for industrial alarm flood analysis. *Proceedings.* 2017 IEEE 15th International Conference on Industrial Informatics (INDIN 2017), Emden, 24–26 July, 781-786. doi:10.1109/INDIN.2017.8104872.

Gebre-Amlak, H. (2018). *Toward a reliable network management framework*. PhD Thesis, University of Missouri. https://hdl.handle.net/10355/67031.

Goel, P., Pistikopoulos, E.N., Mannan, M.S. & Datta, A. (2019). A data-driven alarm and event management framework. *Journal of Loss Prevention in the Process Industries*, 62: 103959. doi:10.1016/j.jlp.2019.103959.

Goldkuhl, G. (2012a). From action research to practice research. *Australasian Journal of Information Systems*, 17(2):57–78. doi:10.3127/ajis.v17i2.688.

Goldkuhl, G. (2012b). Pragmatism vs. interpretivism in qualitative information systems research. *European Journal of Information Systems,*21(1):135–146. doi:10.1057/ejis.2011.54.

Gregor, S. & Hevner, A.R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly: Management Information Systems*, 37(2):337–355. doi:10.25300/MISQ/2013/37.2.01.

Hamister, J.W., Magazine, M.J. & Polak, G.G. (2018). Integrating analytics through the big data information chain: A case from supply chain management. *Journal of Business Logistics,* 39(3): 220–230. doi:10.1111/jbl.12192.

Haryadi. S. (2018). Basic calculation of the network's availability and reliability (Chapter Four of network performance and quality of service). *INA-Rxiv Papers*, 1–13. doi:10.17605/osf.io/z5mwq.

He, Y., Yu, R., Zhao, N.A.N., Yin, H., Yao, H. & Qiu, R.C. (2016). Big data analytics in mobile cellular networks. *IEEE Access*, 4:1985–1996. doi:10.1109/ACCESS.2016.2540520.

Henry, T.F., Mest, D.P. & Safar, M.L. (2019). Pirate wireless: Revenue recognition in the telecommunications industry. *Accounting Perspectives,* 18(1):7–12. doi:10.1111/1911-3838.12167.

Hevner, A.R., March, S.T., Park, J. & Ram, S. (2004). Design science in information systems research. *MIS Quarterly: Management Information Systems,* 28(1):75–105. doi:10.2307/25148625.

Hou, S. & Zhang, X. (2008). Analysis and research for network management alarms correlation based on sequence clustering algorithm. *Proceedings.* International Conference on Intelligent Computation Technology and Automation (ICICTA 2008), Hunan, 982–986. doi:10.1109/ICICTA.2008.263.

Hubballi, N., Biswas, S. & Nandi, S. (2011). Network specific false alarm reduction in intrusion detection system. *Security and Communication Networks,* 4(11):1339–1349. doi:10.1002/sec.261.

Hubballi, N., Biswas, S. & Nandi, S. (2013). Towards reducing false alarms in network intrusion detection systems with data summarisation technique. *Security and Communication Networks,* 6(3):275–285. doi:10.1002/sec.562.

Huddleston, S.H. & Brown, G.G. (2018). Machine learning. *In Informs analytics body of knowledge*. John Wiley & Sons. doi:10.1002/9781119505914.ch7.

Hutson, M. (2019). Bringing machine learning to the masses. *Science*, 365(6452):416-417. doi:10.1126/science.365.6452.416.

ICASA (2019). *The state of the ICT sector report in South Africa.* Pretoria. Available at: https://www.icasa.org.za/uploads/files/state-of-ict-sector-report-2019.pdf. [Accessed: 9 July 2019].

International Telecommunication Union. (2014). *The use of International Mobile Telecommunications for broadband public protection and disaster relief applications.* ITU-R: Radiocommunication Sector of ITU. Geneva. http://www.itu.int/dms_pub/itu-r/opb/rep/R-REP-M.2291-2013-PDF-E.pdf. [Accessed: 17 July 2020].

ITU *see* International Telecommunication Union.

Keim, D.A., Mansmann, F., Schneidewind, J. & Ziegler, H. (2006). Challenges in visual data analysis. *Proceedings.* International Conference on Information Visualisation, London, 9–16. doi:10.1109/IV.2006.31.

Kibria, M.G., Nguyen, K., Villardi, G.P., Zhao, O., Ishizu, K. & Kojima, F. (2018). Big data analytics, machine learning, and Artificial Intelligence in next-generation wireless networks. *IEEE Access,* 6:32328-32338. doi:10.1109/ACCESS.2018.2837692.

Kim, D.S., Shinbo, H. & Yokota, H. (2011). An alarm correlation algorithm for network management based on root cause analysis. Proceedings. *13th International Conference on Advanced Communication Technology (ICACT2011),* Phoenix Park, Korea, 13–16 February.

Kipf, A., Pandey, V., Böttcher, J., Braun, L., Neumann, T. & Kemper, A. (2019). Scalable analytics on fast data. *ACM Transactions on Database Systems*, 44(1): Article No. 1. doi:10.1145/3283811.

Kogan, A., Mayhew, B.W. & Vasarhelyi, M.A. (2019). Audit data analytics research—an application of design science methodology. *Accounting Horizons*, 33(3):69–73. doi:10.2308/acch-52459.

Kumara, P.K.A.M. (2018). *Performance analysis: A case study on network management system using machine learning.* MSc Thesis, University of Tampere School.

Langdon, D.G. & Langdon, K.M. (2019). Work analytic tools: Needs assessment tools for HPT professionals. *Performance Improvement*, 58(2):6–12. doi:10.1002/pfi.21841.

Lauridsen, M., Giménez, L.C., Rodriguez, I., Sørensen, T.B. & Mogensen, P. (2017). From LTE to 5G for connected mobility. *IEEE Communications Magazine*, 55(3):156–162. doi:10.1109/MCOM.2017.1600778CM.

Le, K.H., Lyskov, S. & Gray, J.J. (2020). Multimedia Jupyter Notebook for learning structure prediction and design. *Biophysical Journal*, 118(3):309–318. doi:10.1016/j.bpj.2019.11.1787.

Lin, S., Yu, J. & Ma, J. (2018). Big data driven mobile cellular networks: Modelling, experiments, and applications. *IOP Conference Series: Materials Science and Engineering,* 466. *Proceedings.* 2018 2nd Annual International Conference on Cloud Technology and Communication Engineering, Nanjing, China, 17–19 August.

Linge, S. & Langtangen, H.P. (2020). *Programming for computations.* Python: Springer. doi:10.1007/978-3-030-16877-3.

Loshin, D. (2003). The value of business intelligence. *In Business intelligence*, 11–25. doi:10.1016/B978-155860916-7/50003-8.

Maździarz, A. (2018). Alarm correlation in mobile telecommunications networks based on k-means cluster analysis method. *Journal of telecommunications and information technology*, 1(2):95–102. doi:doi.org/10.26636/jtit.2018.124518.

Menczer, F., Fortunato, S. & Davis, C.A. (2020). Appendix A - Python tutorial. *In A first course in network science*, 221-237. Cambridge: Cambridge University Press. doi:10.1017/9781108653947.010.

Meng, W., Li, W. & Kwok, L.F. (2015). Design of intelligent KNN-based alarm filter using knowledge-based alert verification in intrusion detection. *Security and Communication Networks,* 8(18):3883–3895. doi:10.1002/sec.1307.

Mihailovic, J. (2018). The future of mobile operators - New business models. *Management: Journal of Sustainable Business and Management Solutions in Emerging Economies,* 24(2): 73–84. doi:https://doi.org/10.7595/management.fon.2018.0033.

Mikalef, P., Pappas, I.O., Krogstie, J. & Giannakos, M. (2018). Big data analytics capabilities: A systematic literature review and research agenda. *Information Systems and e-Business Management*, 16(3):547–578. doi:10.1007/s10257-017-0362-y.

MTN. (2019). *MTN Group Annual Results 2019*. https://www.mtn.com/investors/financial-reporting/annual-results/ [Accessed: 18 July 2020].

Mulvey, D., Foh, C.H., Imram, M.A. & Rahim, T. (2019). Cell fault management using machine learning techniques. *IEEE Access*, 7:124514–124539. doi:10.1109/ACCESS.2019.2938410.

Nawal, A. & Cherif, B.M. (2019). Telecommunications sector transformations, innovations and the market structure: The case of Algeria. *International Journal of Technology Management and Sustainable Development,* 18(3):279–299. doi:10.1386/tmsd_00010_1.

Niyazmand, T. & Izadi, I. (2019). Pattern mining in alarm flood sequences using a modified PrefixSpan algorithm. *ISA Transactions,* 90: 287–293. doi:10.1016/j.isatra.2018.12.050.

Ntara, C. (2015) An analysis of M-Pesa use in international transactions: An analysis of M-pesa use in international transactions. *European Journal of Business and Management*, 7(17):73–79.

Omonona, S. (2019). The relationship between political affiliation and performance of mobile telecommunication organisations in South Africa. *Journal of Management Information and Decision Sciences*, 22(3):284–295.

Onstott, E.I. & Heaton, R.C. (2012). Hydrous oxide compositions of rare earth (III) cations and mixed valence chromium (Iii), (V), (Vi). *Procedia Engineering,* 31:588–592. doi:10.1016/j.proeng.2012.01.1164.

Osahenvemwen, O.A. & Omorogiuwa, O. (2011). Analysis on GSM fault management unit. *Journal of Engineering Science and Technology Review,* 4(1):62–67. doi:10.25103/jestr.041.09.

Otani, T., Toube, H., Furutani, M. & Kimura, T. (2017). *Application of AI to mobile network operation*, *ITU Journal: ICT Discoveries,* (1): [online]. https://www.itu.int/en/journal/001/Pages/default.aspx.

Ozcan, B. (2018). Information and communications technology (ICT) and international trade: Evidence from Turkey. *Eurasian Economic Review*, 8(1):93–113. doi:10.1007/s40822-017-0077-x.

P’erez, F. & Granger, B.E. (2018). *Jupyter Notebook documentation.* https://jupyter.org/documentation [Accessed: 27 July 2020].

Pandey, K.N. (2016). Research methodology. *In Paradigms of knowledge management.* Part of the Studies in Systems, Decision and Control book series (SSDC, volume 60). New Delhi: Springer, doi: 10.1007/978-81-322-2785-4_4.

Pavard, B. & Dugdale, J. (2006). The contribution of complexity theory to the study of sociotechnical cooperative systems. *In* Minai, A. & Bar-Yam, Y. (eds.), *Unifying themes in complex systems*, 39–48. Massachusetts (USA): Springer.

Peffers, K., Tuunanen, T., Rothenberger, M.A. & Chatterjee, S. (2007). A design science research methodology for information systems research’, *Journal of Management Information Systems*, 24(3):45–77. doi:10.2753/MIS0742-1222240302.

Piirainen, K.A. & Gonzalez, R.A. (2013). Seeking constructive synergy: Design science and the constructive research approach. *In* Vom Brocke, J., Hekkala, R., Ram, S. & Rossi, M. (eds.), *Design science at the intersection of physical and virtual design*. Lecture Notes in Computer Science, 7939. Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-38827-9_5.

Priyadharshini, A. (2018). An overview of network operation and management. *International Journal for Research in Science Engineering and Technology*, 5(2):1–7.

Ran, L., He, Y. & Ludwig, P.A. (2018). An algorithm for intelligent detection of network abnormal data in dynamic data environment. *Journal of Intelligent and Fuzzy Systems*, 35(4):4361–4371. doi:10.3233/JIFS-169756.

Reali, G., Femminella, M. & Monacelli, L. (2018). Probabilistic codebook-based fault localisation in data networks. *IEEE Transactions on Network and Service Management,* 15(2):567–581. doi:10.1109/TNSM.2018.2795083.

Ruppert, T. & Abonyi, J. (2018). Software sensor for activity-time monitoring and fault detection in production lines. *Sensors,* 18(7):2346. doi:10.3390/s18072346.

Russom, P. (2011) Big Data analytics – TDWI best practices report: Introduction to Big Data analytics. *TDWI Research*, 1:3–5. Available at: https://vivomente.com/wp-content/uploads/2016/04/big-data-analytics-white-paper.pdf. [Accessed: 15 July 2019].

Salah, S., Maciá-fernández, G. & Díaz-verdejo, J.E. (2019). Fusing information from tickets and alerts to improve the incident resolution process. *Information Fusion,* 45(December):38–52. doi:10.1016/j.inffus.2018.01.011.

Sanchez-Pi, N., Leme, L.A.P. & Garcia, A.C.B. (2015). Intelligent agents for alarm management in petroleum ambient. *Journal of Intelligent and Fuzzy Systems*, 28(1):43–53. doi:10.3233/IFS-141198.

Schoenherr, T. & Speier-Pero, C. (2015). Data science, predictive analytics, and big data in supply chain management: Current state and future potential, *Journal of Business Logistics*, 36(1):120–132. doi:10.1111/jbl.12082.

Silva, B.N., Diyan, M. & Han, K. (2019). Big data analytics. *In deep learning: Convergence to big data analytics*, 13–30. Singapore: Springer.

Smith, J., Bekker, H. & Cheater, F. (2011). Theoretical versus pragmatic design in qualitative research. *Nurse Researcher*, 18(2):39–51. doi:10.7748/nr2011.01.18.2.39.c8283.

Solmaz, S.E., Gedik, B., Ferhatosmanoğlu, H., Sözüer, S., Zeydan, E. & Etemoğlu, Ç.Ö. (2017). ALACA: A platform for dynamic alarm collection and alert notification in network management systems. *International Journal of Network Management,* 27(4):1–20. doi:10.1002/nem.1980.

Styan, J.-B. (2019). Deconstructing Eskom. *Finweek*, 1:32–35. https://hdl.handle.net/10520/EJC-136b3407d4.

Su, J., Guo, C., Zang, H., Yang, F., Huang, D., Gao, X. & Hao, Y. (2018). A multi-setpoint delay-timer alarming strategy for industrial alarm monitoring. *Journal of Loss Prevention in the Process Industries*. 54:1–9. doi:10.1016/j.jlp.2018.02.004.

Swaminathan, J.M. (2018). Big data analytics for rapid, impactful, sustained, and efficient (RISE) humanitarian operations. *Production and Operations Management*, 27(9):1696–1700. doi:10.1111/poms.12840.

Tabia, K. & Leray, P. (2011). Alert correlation: Severe attack prediction and controlling false alarm rate tradeoffs. *Intelligent Data Analysis*, 15(6):955–978. doi:10.3233/IDA-2011-0504.

Tang, C., Tara, S., Chang, R.N. & Zhang, C. (2019). Black-box performance control for high-volume non-interactive systems. *Proceedings.* The 2009 USENIX Annual Technical Conference, California, 1–27.

Taori, P. & Dasararaju, H.K. (2019). Introduction to Python. *In* Pochiraju, B. & Seshadri, S. (eds.), *Essentials of business analytics.* International Series in Operations Research and Management Science, 264. doi:10.1007/978-3-319-68837-4_29.

Teagarden, M.B. (2019). *Africa rising!* Thunderbird International Business Review. Wiley Online Library. https://doi.org/10.1002/tie.22034.

Troisi, O., D'Arco, M., Loia, F. & Maione, G. (2018). Big data management: The case of Mulino Bianco's engagement platform for value co-creation. *International Journal of Engineering Business Management,* 10(1):1–8. doi:10.1177/1847979018767776.

United Nations. (2016). *The sustainable development goals report 2020*. https://unstats.un.org/sdgs/report/2016/. [Accessed: 18 July 2020],

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J. & Van der Walt, S.J. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods,* 1(1):1–22. doi:10.1038/s41592-019-0686-2.

Vodacom. (2019). *Vodacom Group Ltd Consolidated Annual Financial Statements 2018*. Available at: http://vodacom-reports.co.za/integrated-reports/ir-2019/documents/downloads/Consolidated-annual-financial-statements-2019.pdf. [Accessed: 23 January 2020].

Wallin, S. & Landén, L. (2008). Telecom alarm prioritisation using neural networks. *Proceedings.* 22nd International Conference on Advanced Information Networking and Applications – Workshops (AINA workshops 2008), Okinawa, Japan, 25–28 March, 1468-1473. IEEE. doi:10.1109/WAINA.2008.105.

Wang, J., Li, H., Huang, J. & Su, C. (2016). Association rules mining based analysis of consequential alarm sequences in chemical processes. *Journal of Loss Prevention in the Process Industries,* 41(1):178–185. doi:10.1016/j.jlp.2016.03.022.

Wang, J., He, C., Liu, Y., Tian, G., Peng, I., Xing, J., Ruan, X., Xie, H. & Wang, F.L. (2017a). Efficient alarm behavior analytics for telecom networks. *Information Sciences,* 402:1–14. doi:10.1016/j.ins.2017.03.020.

Wang, J., Yang, Z., Chen, K. & Zhou, D. (2017b). Practices of detecting and removing nuisance alarms for alarm overloading in thermal power plants. *Control Engineering Practice*, 67:21–30. doi:10.1016/j.conengprac.2017.07.003.

Wang, X., McArthur, S.D.J., Strachan, S.M., Kirkwood, J.D. & Paisley, B. (2018). A data analytic approach to automatic fault diagnosis and prognosis for distribution automation. *IEEE Transactions on Smart Grid,* 9(6):6265–6273. doi: 10.1109/TSG.2017.2707107.

Weber, S. (2010). Design science research: Paradigm or approach? *Proceedings.* Americas Conference on Information Systems (AMCI 2010), Lima, Peru, August, 214. http://aisel.aisnet.org/amcis2010/214.

Yannis, P. & Nikolaos, B. (2018). Quantitative and qualitative research in business & technology: Justifying a suitable research methodology. *Review of Integrative Business and Economics Research*, 7(1):91–105.

Yu, Y., Li, X., Leng, X. & Song, L. (2019). Fault management in software-defined networking: A survey. *IEEE Communications Surveys & Tutorials,* 21(1):349–392. Firstquarter. doi:10.1109/COMST.2018.2868922.

Zeng, Z., Tan, W. & Zhou, R. (2017). Performance assessment for generalised delay-timers in alarm configuration. *Journal of Process Control*, 57:80–101. doi:10.1016/j.jprocont.2017.06.013.

Zeydan, E., Yabas, U., Sözüer, S. & Etemoğlu, Ç.Ö. (2016). Streaming alarm data analytics for mobile service providers. *Proceedings.* 2016 IEEE/IFIP Network Operations and Management Symposium (NOMS 2016), Istanbul, 1021–1022. doi:10.1109/NOMS.2016.7502953.

Zhang, H., Wang, H., Li, J. & Gao, H. (2018). A generic data analytics system for manufacturing production. *Big Data Mining and Analytics*, 1(2):160–171. doi:10.26599/BDMA.2018.9020016.

Zupan, J. & Medhi, D. (2003). An alarm management approach in the management of multi-layered networks. *Proceedings.* The 3rd IEEE Workshop on IP Operations and Management (IPOM 2003), Kansas City, 77–84. doi:10.1109/IPOM.2003.1251227.

APPENDIX A: MoU

APPENDIX A - MoU.pdf

APPENDIX B: ETHICS FORM

APPENDIX B - Ethics
Form.pdf

APPENDIX C: PROBLEM IDENTIFICATION NOT REFINED

APPENDIX C -
Problem Identification

APPENDIX D: PROBLEM IDENTIFICATION REFINED

APPENDIX D -
Problem Identification

APPENDIX E: CONTROL

APPENDIX E -
Control.doc

APPENDIX F: EXPERIMENT

APPENDIX F -
Experiment.doc

APPENDIX G: PROTOTYPE 1 USER FEEDBACK

APPENDIX G -
Prototype 1 User Feec

APPENDIX H: PROTOTYPE 2 USER FEEDBACK

APPENDIX H -
Prototype 2 User Feec

APPENDIX I: PROTOTYPE 3 USER FEEDBACK

APPENDIX I -
Prototype 3 User Feec

APPENDIX J: TEAM PROFILE

APPENDIX J - Team
Profile.docx

APPENDIX K: ALARM AND INCIDENT ANALYTICS PROGRAM

```
import matplotlib
from bokeh.models import ColumnDataSource, Plot, LabelSet
from bokeh.models.widgets import Tabs, Panel, PreText, DataTable, TableColumn
from bokeh.io import curdoc,show, output_notebook, output_file
from bokeh.layouts import gridplot, layout
from bokeh.plotting import figure, show
import cx_Oracle
import pandas as pd
import seaborn as sns
import numpy as np

print("establishing connection with data source")
ip = "10.132.85.11" #xx added to mask actual IP address
port = 1521
SID = "oss"
dsn_tns = cx_Oracle.makedsn(ip, port, SID)

f= open("cred")
# print(f.readline(),f.readline())
db = cx_Oracle.connect(f.readline().strip(),f.readline().strip(), dsn_tns)
f.close()
#cursor = db.cursor()
print("established connection with data source")

print ("collect incident management data")
df_all = pd.read_excel("\\\\PCOGDS1ZATCWI\\Shared_Export\\Cognos\\file\\CoreNOC\\VSA_Core_Incidents_Daily-en-gb.xlsx")

print("mss data extraction starting")
mssrawq="""select dn, alarm_status, alarm_type, alarm_time,insert_time, cancel_time, acked_by, alarm_number, text,
user_additional_info, supplementary_info, diagnostic_info, lifted_dn, CASE
    when lifted_dn like '%MSC-397824%' then 'POOL1_MSJFR'
    when lifted_dn like '%MSC-913028%' then 'POOL1_MSJMR'
    when lifted_dn like '%MSC-397825%' then 'POOL1_MSMTH'
    when lifted_dn like '%MSC-373024%' then 'POOL2_MSPRL'
    when lifted_dn like '%MSC-373025%' then 'POOL2_MSPPE'
    when lifted_dn like '%MSC-372984%' then 'POOL2_MSPSO'
    when lifted_dn like '%MSC-913029%' then 'POOL3_MSDME'
    when lifted_dn like '%MSC-372857%' then 'POOL3_MSDNR'
    when lifted_dn like '%MSC-913031%' then 'POOL3_MSBEO'
    when lifted_dn like '%MSC-373001%' then 'POOL4_MSCTN'
    when lifted_dn like '%MSC-404656%' then 'POOL3_MSDTJ'
    when lifted_dn like '%MSC-373003%' then 'POOL4_MSCFO'
    when lifted_dn like '%MSC-372803%' then 'POOL4_MSPFS'
    when lifted_dn like '%MSC-404655%' then 'POOL2_MSPCE'
    when lifted_dn like '%MSC-395187%' then 'TAS_TSPSI'
    when lifted_dn like '%MSC-395188%' then 'TAS_TSMTA'
    when lifted_dn like '%MSC-399311%' then 'POOL5_MSIPS'
    when lifted_dn like '%MSC-399308%' then 'POOL5_MSIGM'
    when lifted_dn like '%MSC-326915%' then 'VCL_MSLKK01'
    when lifted_dn like '%MSC-396741%' then 'VCL_MSMSU01'
    when lifted_dn like '%MGW-326917%' then 'VCL_MGLKK01'
    when lifted_dn like '%MGW-396742%' then 'VCL_MGMSU01'
    ELSE 'Unknown'
    END as Node,case when Severity = 1 then 'Critical'
                when Severity = 2 then 'Major'
                when Severity = 3 then 'Minor'
                when Severity = 4 then 'Warning'
    ELSE 'Unknown'
    END as Severity
from fm.fx_alarm
where alarm_time >= trunc(sysdate)-1
and lifted_dn like '%MSC%'"""
mssraw=pd.read_sql(mssrawq,con=db)
print("mss data extraction completed")

print("omgw data extraction starting")
omgwrawq="""select dn, alarm_status, case when Severity = 1 then 'Critical'
                when Severity = 2 then 'Major'
                when Severity = 3 then 'Minor'
                when Severity = 4 then 'Warning'
    ELSE 'Unknown'
    END as Severity,alarm_type alarm_time,insert_time, cancel_time, acked_by, alarm_number, text, user_additional_info,
supplementary_info, diagnostic_info, lifted_dn, substr(lifted_dn,16,5) as node
from fm.fx_alarm
```

```
where alarm_time >= trunc(sysdate)-1
and oc_id = 'OMGW:OMGW'
"""
#cursor.execute(omgwrawq)
#one_records = cursor.fetchall()
#for i in one_records:
    #print(i)
omgwraw=pd.read_sql(omgwrawq,con=db)
#omgwraw
print("omgw data extraction completed")

print("shortning omgw supplementary info")
def shorten(x):
    """Shortens Supp_info names for plotting"""
    if x == None or x == 'None':
        return "/none"
    import re
    dontinclude =
['fsClusterId','fshaRecoveryUnitName','fsipHostName','fsFragmentId','fsFragmentId','fshaProcessInstanceName','
DSPHASProxy','DSPHASProxy','FSDSPHasProxyServer','ClusterRoot','SGWSS7LM','851*','NaN']
    l = re.split('=|,',x)
    l = [el.strip() for el in l if el not in dontinclude]
#     print(l)
    return ";".join(l)

print ("creation of event dataframes")
mssalarmsactive = mssraw[mssraw["ALARM_STATUS"] == 1]
mssalarmsclear = mssraw[mssraw["ALARM_STATUS"] == 0]
omgwalarmsactive = omgwraw[omgwraw["ALARM_STATUS"] == 1]
omgwalarmsclear = omgwraw[omgwraw["ALARM_STATUS"] == 0]
cncedf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2745]
cefiupdf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 3263]
cefdf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2668]
aiaedf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2194]
rsudf = mssalarmsclear[mssalarmsclear["ALARM_NUMBER"] == 2064]
# omgwalarmsclear.SUPPLEMENTARY_INFO = str(omgwalarmsclear.SUPPLEMENTARY_INFO)
omgwalarmsclear.SUPPLEMENTARY_INFO = omgwalarmsclear.SUPPLEMENTARY_INFO.apply(shorten)
omgwalarmsactive.SUPPLEMENTARY_INFO = omgwalarmsactive.SUPPLEMENTARY_INFO.apply(shorten)

df_open = df_all[df_all['INC_STATUS'] != 'Closed']
df_inprogress = df_open[(df_open['INC_STATUS'] == 'In Progress') & (df_open['INC_PRIORITY'] == 'Critical')]
df_inprogress['date_time'] = pd.to_datetime('now')
df_inprogress['Last_Updated'] = df_inprogress['date_time'] - df_inprogress['REG_MODIFIED']
df_inprogress['Last_Updated']=df_inprogress['Last_Updated']/np.timedelta64(1,'h')

df_assigned = df_open[(df_open['INC_STATUS'] == 'Assigned') & (df_open['INC_PRIORITY'] == 'Critical')]
df_closed = df_all[df_all['INC_STATUS'] == 'Closed']
df_closed['MTTR'] = df_closed['INC_EVENT_FINISH'] - df_closed['INC_EVENT_START']
df_closed['MTTR']=df_closed['MTTR']/np.timedelta64(1,'h')
df_closed['date_time'] = pd.to_datetime('now')
df_all['date_time'] = pd.to_datetime('now')

def get_plot(df,title):

    temp={"Columns":df.index,
         "values": df.values}
    src=ColumnDataSource(temp)

    plot = figure(x_range=df.index.values, title=title,plot_height=800,plot_width=700,background_fill_color='white',
background_fill_alpha=0.5, border_fill_color='white', border_fill_alpha=0.2)
    plot.xaxis.major_label_orientation = "vertical"
    plot.vbar(x="Columns", top = "values", width=0.8, source=src, fill_color='red', line_color='black')

    #Anotations
    labels = LabelSet(x="Columns", y="values", text="values",
                      x_offset=-8, y_offset=0, source=src,text_font_size="9pt", render_mode='canvas')
    plot.add_layout(labels)
    plot.xgrid.grid_line_color = None
    plot.y_range.start = 0
    return plot

print("dictionary for cnce")
def extract(s):
    return s.split()

def extract_code_from_list_as_string(l, index):
    if index >= len(l):
        return 'NONE'
    else:
```

```
        return l[index].strip()

def first_col_dict(x):
    dictionary = {'03' : "Cell Identity (CI)",
                  '04' : "Location Area Code (LAC)",
                  '05' : "Location Area Code and Cell Identity (LAC + CI)",
                  '06' : "Service Information Octet and Signalling Point Code (SIO+SPC)",
                  '08': 'Cell global Identity (CGI), (includes: MCC,MNC,LAC,CI)',
                  '09': 'Location Area Identity (LAI), (includes: MCC,MNC,LAC)',
                 '0A': 'Service Area Identity (SAI), (includes: MCC,MNC,LAC,SAC)',
                 '0B': 'Global RNC Identity, (includes: MCC,MNC,RNC_ID)',
                 '0C': 'RNC identity',
                 '0D': 'Service Area Code (SAC)',
                 '0E': 'Multimedia gateway (MGW(Rel99))'
    }
    return dictionary[x]

def seventh_col_dict(x):
    dictionary = {  '00' :  "unknown",
                    '01' :  "supplementary service",
                    '02' :  "mobile-originated call",
                    '03' :  "IMSI detach",
                    '04' :  "location update",
                    '05' :  "mobile-originated short message",
                    '06' :  "mobile-terminated short message",
                    '07' :  "mobile-terminated call",
                    '08' :  "emergency call",
                    '09' :  "handover",
                    '0A' :  "re-establishment of the connection",
                    '0B' :  "SCCP subsystem state change",
                    '0C' :  "SDCCH handover",
                    '0D' :  "handover in MSC-B",
                    '0E' :  "resource indication",
                    '0F' :  "handover candidate",
                    '10' :  "load announcement"

    }
    return dictionary[x]

def translate_cols(df):
    functions = {"Supplementary_0" : first_col_dict,
                 "Supplementary_7" : seventh_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = cncedf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    cncedf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))
cncedf = translate_cols(cncedf)
#cncedf

print("Dictionary for RSU")
def extract(s):
    return s.split()

def extract_code_from_list_as_string(l, index):
    if index >= len(l):
        return 'NONE'
    else:
        return l[index].strip()

def second_col_dict(x):
    dictionary = {  '00' :  "international network 0 (IN0)",
                    '04' :  "international network 1 (IN1)",
                    '08' :  "national network 0 (NA0)",
                    '0C' :  "national network 1 (NA1)"

    }
    return dictionary[x]
def third_col_dict(x):
    dictionary = {  '00' :  "ordinary signalling point",
                    '03' :  " group of signalling points",
                    '04' :  "adjacent signalling point"
```

```
    }
    return dictionary[x]

def translate_cols(df):
    functions = {"Supplementary_1" : second_col_dict,
                 "Supplementary_2" : third_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = rsudf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    rsudf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))

rsudf = translate_cols(rsudf)
#rsudf

print ("Sourcing network configuration data")
query="""select distinct dsp_sig_pnt_name, dsp_sig_pnt_code
from cmdmss.c_mss_dsp
where dsp_sig_pnt_name IS NOT NULL"""
#cursor.execute(query)
#one_records = cursor.fetchall()
#for i in one_records:
    #print(i)
RAN=pd.read_sql(query,con=db)
#RAN

pc = mssalarmsactive[mssalarmsactive['TEXT'] == 'ROUTE SET UNAVAILABLE'].sort_values('SUPPLEMENTARY_INFO')
pc["Point_code"] = pc['SUPPLEMENTARY_INFO'].apply(lambda x : x[4:8])
pc.sort_values('Point_code')
pc.drop('SUPPLEMENTARY_INFO',axis=1,inplace=True)
#alarms.set_index("TEXT",inplace = True)
pc

print ("Insert point code names to IDs")
alarm=pc.merge(RAN,left_on=["Point_code"],right_on=["DSP_SIG_PNT_CODE"]).drop(["DSP_SIG_PNT_CODE"],axis=1)
alarm

from bokeh.models.widgets import DataTable, TableColumn

def get_ROUTE_SET_UNAVAILABLE_ACTIVE(df):
    df2 = alarm.head(10)
    """Get a table with the latest Pointcode"""
    data = {"ALARM_TIME" : df2.ALARM_TIME,"TEXT" : df2.TEXT,
            "NODE" : df2.NODE,"Point_code" : df2.Point_code,
            "DSP_SIG_PNT_NAME" : df2.DSP_SIG_PNT_NAME}
    src = ColumnDataSource(data)

    columns = [
            TableColumn(field="ALARM_TIME", title="Alarm time"),
            TableColumn(field="TEXT", title="Text"),
            TableColumn(field="NODE", title="Node"),
            TableColumn(field="Point_code", title="Point code id"),
            TableColumn(field="DSP_SIG_PNT_NAME", title="Point code name")
        ]
    data_table = DataTable(source=src, columns=columns, width=300, height=500)
    panel = Panel(child=data_table,title="Unavailable Active")
    return panel

# show(get_ROUTE_SET_UNAVAILABLE_ACTIVE(alarm))

PC = mssalarmsclear[mssalarmsclear['TEXT'] == 'ROUTE SET UNAVAILABLE'].sort_values('SUPPLEMENTARY_INFO')
PC["Point_code"] = PC['SUPPLEMENTARY_INFO'].apply(lambda x : x[4:8])
PC_counted = PC.groupby(["NODE","TEXT","Point_code"]).count().reset_index()
PC_counted.drop(PC_counted.columns.difference(["Point_code","NODE","TEXT","SEVERITY"]),axis=1,inplace=True)
PC_counted.rename({"SEVERITY":"COUNT"},axis=1,inplace=True)
PC.sort_values('Point_code')
PC.drop('SUPPLEMENTARY_INFO',axis=1,inplace=True)
#alarms.set_index("TEXT",inplace = True)
#PC_counted
```

```
alarm2=PC_counted.merge(RAN,left_on=["Point_code"],right_on=["DSP_SIG_PNT_CODE"]).drop(["DSP_SIG_PNT_CODE"],a
xis=1)
#alarm2

print ("RSU table")
from bokeh.models.widgets import DataTable, TableColumn

def get_ROUTE_SET_UNAVAILABLE_CLEAR(df):
    df2 = alarm2.head(10)
    """Get a table with the latest Pointcode"""
    data = {"COUNT" : df2.COUNT,"TEXT" : df2.TEXT,
            "NODE" : df2.NODE,"Point_code" : df2.Point_code,
            "DSP_SIG_PNT_NAME" : df2.DSP_SIG_PNT_NAME}
    src = ColumnDataSource(data)

    columns = [
            TableColumn(field="COUNT", title="Count"),
            TableColumn(field="TEXT", title="Text"),
            TableColumn(field="NODE", title="Node"),
            TableColumn(field="Point_code", title="Point code id"),
            TableColumn(field="DSP_SIG_PNT_NAME", title="Point code name")
        ]
    data_table = DataTable(source=src, columns=columns, width=300, height=500)
    panel = Panel(child=data_table,title="Unavailable Clear")
    return panel

# show(get_ROUTE_SET_UNAVAILABLE_CLEAR(alarm2))

print ("DIctionary for cefei")
def first_col_dict(x):
    dictionary = {  '01' :  "opening of the file failed",
                    '02' :  "initialization of file pointer failed",
                    '03' :  "creation of hand failed",
                    '04' :  "setting of TNSDL-timer failed",
                    '05' :  "wrong answer received for a service request",
                    '06' :  "use of parameter file failed",
                    '07' :  "no free handover number",
                    '08' :  "no free roaming number",
                    '09' :  "the table of IMSI numbers is full",
                    '10' :  "the table of TMSI numbers is full"

    }
    return dictionary[x]

def second_col_dict(x):
    dictionary = {  '00' :  "undefined",
                    '01' :  "all",
                    '02' :  "MS",
                    '03' :  "PBX",
                    '04' :  "PSTN",
                    '05' :  "handover",
                    '06' :  "call forwarding",
                    '07' :  "call drop-back",
                    '08' :  "call transfer",
                    '09' :  "location update"

    }
    return dictionary[x]

def third_col_dict(x):
    dictionary = {  '00' :  "undefined",
                    '01' :  "all",
                    '02' :  "MS",
                    '03' :  "PBX",
                    '04' :  "PSTN",
                    '05' :  "handover",
                    '06' :  "call forwarding",
                    '07' :  "call drop-back",
                    '08' :  "call transfer",
                    '09' :  "location update"

    }
    return dictionary[x]

def fourth_col_dict(x):
    dictionary = {  '01' :  "partly",
                    '02' :  "the whole computer unit in question",
                    '03' :  "the whole network element in question"
```

```
    }
    return dictionary[x]

def translate_cols(df):
    functions = {"Supplementary_0" : first_col_dict,
                 "Supplementary_1" : second_col_dict,
                 "Supplementary_2" : third_col_dict,
                 "Supplementary_3" : fourth_col_dict
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = cefdf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    cefdf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))

cefdf = translate_cols(cefdf)
#cefdf

print ("Dictionary for cefiup")
def extract(s):
    return s.split()

def extract_code_from_list_as_string(l, index):
    if index >= len(l):
        return 'NONE'
    else:
        return l[index].strip()

def first_col_dict(x):
    dictionary = {'03' :  "Cell Identity (CI)",
                  '00' :  "No translation info from manual",
                  '01' :  "No translation info from manual",
                  '02' :  "No translation info from manual",
                  '04' :  "Location Area Code (LAC)",
                  '05' :  "Location Area Code and Cell Identity (LAC + CI)",
                  '06' :  "Service Information Octet and Signalling Point Code (SIO+SPC)",
                  '08' :  "Cell global Identity (CGI), (includes: MCC,MNC,LAC,CI)",
                  '09' :  "Location Area Identity (LAI), (includes: MCC,MNC,LAC)",
                 '0A' :  "Service Area Identity (SAI), (includes: MCC,MNC,LAC,SAC)",
                 '0B' :  "Global RNC Identity, (includes: MCC,MNC,RNC_ID)",
                 '0C' :  "RNC identity",
                 '0D' :  "Service Area Code (SAC)",
                 '0E' :  "Multimedia gateway (MGW(Rel99))"
    }
    return dictionary[x]

def second_col_dict(x):
    dictionary = {  '00' :  "unknown",
                    '01' :  "supplementary service",
                    '02' :  "mobile-originated call",
                    '03' :  "IMSI detach",
                    '04' :  "location update",
                    '05' :  "mobile-originated short message",
                    '06' :  "mobile-terminated short message",
                    '07' :  "mobile-terminated call",
                    '08' :  "emergency call",
                    '09' :  "handover",
                    '0A' :  "re-establishment of the connection",
                    '2A' :  "congestion",
                    '0B' :  "SCCP subsystem state change",
                    '0C' :  "SDCCH handover",
                    '0D' :  "handover in MSC-B",
                    '0E' :  "resource indication",
                    '0F' :  "handover candidate",
                    '10' :  "load announcement",
                    '18' :  "ring back tone"

    }
    return dictionary[x]

def translate_cols(df):
    functions = {"Supplementary_0" : first_col_dict,
                 "Supplementary_1" : second_col_dict
```

```
    }

    for col, func in functions.items():
        df[col] = df[col].apply(lambda x : func(x))

    return df

tokens = cefiupdf.SUPPLEMENTARY_INFO.apply(extract)
tokens = pd.DataFrame(tokens)
col_name = "Supplementary_"
for i in range(0,8):
    cefiupdf[col_name+str(i)] = tokens.SUPPLEMENTARY_INFO.apply(lambda x : extract_code_from_list_as_string(x,i))

cefiupdf = translate_cols(cefiupdf)
#cefiupdf

print ("create plots")
def get_plot(df,title):
    """Generate and returns a plot"""

    temp={"Columns":df.index,
         "values": df.values}
    src=ColumnDataSource(temp)

    plot = figure(x_range=df.index.values, title=title,plot_height=800,plot_width=700,background_fill_color='white',
background_fill_alpha=0.5, border_fill_color='white', border_fill_alpha=0.2)
    plot.xaxis.major_label_orientation = "vertical"
    plot.vbar(x="Columns", top = "values", width=0.8, source=src, fill_color='red', line_color='black')


    #Anotations
    labels = LabelSet(x="Columns", y="values", text="values",
                          x_offset=-8, y_offset=0, source=src,text_font_size="10pt", render_mode='canvas')
    plot.add_layout(labels)

    plot.xgrid.grid_line_color = None
    plot.y_range.start = 0
    return plot

print ("crate tabs and figures")
def make_tab(fig1,fig2,fig3,fig4,title):
    grid = gridplot([[fig1,fig2],[fig3,fig4]])
    tab = Panel(child=grid,title=title)
    return tab

def make_to_figs(fig1,fig2,title):
    grid = gridplot([[fig1,fig2]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab2(fig1,fig2,title):
    grid = gridplot([[fig1,fig2]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab_table(fig1,title):
    grid = gridplot([[fig1,None],[None,None]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab_b(fig1,fig2,fig3,fig4,title):
    grid = gridplot([[fig1,fig2],[fig3,fig4]])
    tab = Panel(child=grid,title=title)
    return tab

def make_tab_table_b(fig1,title):
    grid = gridplot([[fig1,None],[None,None]])
    tab = Panel(child=grid,title=title)
    return tab

print ("populating data into the dashboard")
p1 = get_plot(mssalarmsactive["NODE"].value_counts().head(20),"MSS-Plot Top 20 Count Per MSS (Active)")
p2 = get_plot(mssalarmsactive["TEXT"].value_counts().head(20),"MSS-Plot Top 20 Count Per ALARM  (Active)")
p3 = get_plot(mssalarmsactive["SUPPLEMENTARY_INFO"].value_counts().head(20),"MSS-Plot Top 20 Count Per Supp Info
(Active)")
p4 = get_plot(mssalarmsactive["SEVERITY"].value_counts().head(20),"MSS-Plot Alarm Count Per SEVERITY (Active)")
tab1 = make_tab(p1,p2,p3,p4,"Active MSS")

p1 = get_plot(mssalarmsclear["NODE"].value_counts().head(20),"MSS-Plot Top 20 Count Per MSS (Clear)")
```

```
p2 = get_plot(mssalarmsclear["TEXT"].value_counts().head(20),"MSS-Plot Top 20 Count Per ALARM  (Clear)")
p3 = get_plot(mssalarmsclear["SUPPLEMENTARY_INFO"].value_counts().head(20),"MSS-Plot Top 20 Count Per Supp Info
(Clear)")
p4 = get_plot(mssalarmsclear["SEVERITY"].value_counts().head(20),"MSS-Plot Alarm Count Per SEVERITY (Clear)")
tab2 = make_tab(p1,p2,p3,p4,"Cleared MSS")

p1 = get_plot(omgwalarmsactive["NODE"].value_counts().head(5),"OMGW-Plot Top 5 Count Per OMGW (Active)")
p2 = get_plot(omgwalarmsactive["TEXT"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Alarm (Active)")
p3 = get_plot(omgwalarmsactive["SUPPLEMENTARY_INFO"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Supp Info
(Active)")
p4 = get_plot(omgwalarmsactive["SEVERITY"].value_counts().head(5),"OMGW-Plot Top Count Per Severity (Active)")
tab3 = make_tab(p1,p2,p3,p4,"Active OMGW")

from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.ensemble import RandomForestRegressor
p1 = get_plot(omgwalarmsclear["NODE"].value_counts().head(5),"OMGW-Plot Top 5 Alarm Count Per OMGW (Clear)")
p2 = get_plot(omgwalarmsclear["TEXT"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Per Alarm (Clear)")
p3 = get_plot(omgwalarmsclear["SUPPLEMENTARY_INFO"].value_counts().head(5),"OMGW-Plot Top 5 Count Per Per Supp
Info (Clear)")
p4 = get_plot(omgwalarmsclear["SEVERITY"].value_counts().head(5),"OMGW-Plot Top Count Per Severity  (Clear)")
tab4 = make_tab(p1,p2,p3,p4,"Cleared OMGW")

p5 = get_ROUTE_SET_UNAVAILABLE_CLEAR(alarm2)
p6 = get_ROUTE_SET_UNAVAILABLE_ACTIVE(alarm)
# tab5 =  make_to_figs(p5,p6,"Route set unavailable")
t5 = Tabs(tabs=[p5,p6])
tab5 = Panel(child=t5,title="ROUTE SET")

data = {"NODE" :rsudf.NODE,
   #"ALARM_TIME" : rsudf.AlARM_TIME,♥
   "Supplementary_0" : rsudf.Supplementary_0,
   "Supplementary_1" : rsudf.Supplementary_1,
   "Supplementary_2" : rsudf.Supplementary_2
   }
src = ColumnDataSource(data)
columns = [
   TableColumn(field="NODE",title="NETWORK_ELEMENT"),
   #TableColumn(field="ALARM_TIME",title="EVENT_TRIGGER"),
   TableColumn(field="Supplementary_0",title="POINT_CODE_ID"),
   TableColumn(field="Supplementary_1",title="PEER_ELEMENT_TYPE"),
   TableColumn(field="Supplementary_2",title="SIGNALLING_POINT")
]
data_table = DataTable(source=src, columns=columns, width=600, height=900, editable=True)
tab6 = make_tab_table(data_table,"RSU_TYPES")

p7 = get_plot(cncedf["Supplementary_0"].value_counts().head(20),"CELLULAR NETWORK CONFIGURATION ERROR -
Reasons")
p8 = get_plot(cncedf["Supplementary_7"].value_counts().head(20),"CELLULAR NETWORK CONFIGURATION ERROR -
Transaction Type")
#t6 = Tabs(tabs=p7,p8)
tab7 = make_tab2(p7,p8,"CNCE")

p9 = get_plot(cefiupdf["Supplementary_0"].value_counts().head(20),"CALL ESTABLISHMENT FAILURE IN USER PLANE
CONTROL - Reasons")
p10 = get_plot(cefiupdf["Supplementary_1"].value_counts().head(20),"CALL ESTABLISHMENT FAILURE IN USER PLANE
CONTROL - Transaction Type")
tab8 = make_tab2(p9,p10,"CEFIUP")

##############
pb3 = get_plot(df_open["INC_ASSIGNED_GROUP"].value_counts().head(20),"Open Incidents Per Team")
pb4 = get_plot(df_open["INC_OWNER_NAME"].value_counts().head(20),"Open Incidents Per Person")
pb1 = get_plot(df_open["INC_STATUS"].value_counts().head(20),"Open Incidents Count - Status")
pb2 = get_plot(df_open["INC_PRIORITY"].value_counts().head(20),"Open Incidents Count - Priority")
tab9 = make_tab_b(pb3,pb4,pb1,pb2,"Active Incidents")

pb3 = get_plot(df_closed["INC_ASSIGNED_GROUP"].value_counts().head(20),"Closed Incidents Per Team")
pb4 = get_plot(df_closed["INC_OWNER_NAME"].value_counts().head(20),"Closed Incidents Per Person")
pb1 = get_plot(df_closed["INC_DESCRIPTION"].value_counts().head(20),"Closed Incidents Count - Total")
pb2 = get_plot(df_closed["INC_PRIORITY"].value_counts().head(20),"Closed Incidents Count - Priority")
tab10 = make_tab_b(pb3,pb4,pb1,pb2,"Closed Incidents")

data = {"INCIDENT_ID" :df_inprogress.INCIDENT_ID,
   "INC_CI_SITE_NAME" : df_inprogress.INC_CI_SITE_NAME,
#    "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
   "INC_OWNER_NAME" : df_inprogress.INC_OWNER_NAME,
   "INC_DESCRIPTION" : df_inprogress.INC_DESCRIPTION
   }
src = ColumnDataSource(data)
```

```
columns = [
   TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
   TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#    TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
   TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
   TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table = DataTable(source=src, columns=columns, width=900, height=600, editable=True)

tab11 = make_tab_table_b(data_table,"Critical In Progress Incidents")

data1 = {"INCIDENT_ID" :df_assigned.INCIDENT_ID,
   "INC_CI_SITE_NAME" : df_assigned.INC_CI_SITE_NAME,
#    "INC_EVENT_START" : df_inprogress.INC_EVENT_START,
   "INC_OWNER_NAME" : df_assigned.INC_OWNER_NAME,
   "INC_DESCRIPTION" : df_assigned.INC_DESCRIPTION
   }
src = ColumnDataSource(data1)
columns = [
   TableColumn(field="INCIDENT_ID",title="INCIDENT_ID"),
   TableColumn(field="INC_CI_SITE_NAME",title="INC_CI_SITE_NAME"),
#    TableColumn(field="INC_EVENT_START",title="INC_EVENT_START"),
   TableColumn(field="INC_OWNER_NAME",title="INC_OWNER_NAME"),
   TableColumn(field="INC_DESCRIPTION",title="INC_DESCRIPTION")
]
data_table1 = DataTable(source=src, columns=columns, width=800, height=600, editable=True)

tab12 = make_tab_table_b(data_table1,"Critical Assigned Incidents")
#############

dash = Tabs(tabs=([tab1,tab2,tab3,tab4,tab5,tab6,tab7,tab8,tab9,tab10,tab11,tab12]))

show(dash)
```

APPENDIX L: KMEANS CLUSTERING PROGRAM

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler
import cx_Oracle

ip = "1.1.1.1"
port = 1521
SID = "oss"
dsn_tns = cx_Oracle.makedsn(ip, port, SID)
f= open("cred")
# print(f.readline(),f.readline())
db = cx_Oracle.connect(f.readline().strip(),f.readline().strip(), dsn_tns)
f.close()
#cursor = db.cursor()

def cluster(num_clusters, tmp):
    X = tmp[tmp.columns.difference(['NODE','ALARM_TIME','TEXT','ALARM_TYPE'])]
    KMeans = KMeans(n_clusters=num_clusters, random_state=0).fit(X)
    tmp['CLUSTER'] = KMeans.labels_
    return tmp

def plot_all(df,title):

    cols = ["ALARM_TIME_hours",
                        "SEVERITY",
                        "ALARM_TYPE","TEXT","CLUSTER"]

    grid = sns.pairplot(df[cols],hue='CLUSTER')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='SEVERITY')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='ALARM_TYPE')
    grid.fig.suptitle(title)

    grid = sns.pairplot(df[cols],hue='TEXT')
    grid.fig.suptitle(title)

def decompose_date_time_features(df, index):
    """
    Uses a column index for a date/time attributes in a data frame and decomposes this field into its
    various atomic parts.  Each part is then added to the data frame as a new
    column/attribute.
    :param df: The data frame in which date/time attributes are to be found.
    :param index: The index of the column containing the date/time attribute.
    :return: The dataframe appended with columns representing the decomposed date/time information
    """
    if(index < 0 or index >= len(df.columns)):
        print("Column index invalid")
        return

    colString = df.columns[index] + "_"
    units = ["year", "month", "day_of_month", "day_of_week_number", "day_of_year_number", "week_of_year",
           "quarter_of_year", "hours", "minutes", "seconds"]
    units = [colString + v for v in units]
    df[units[0]] = pd.DatetimeIndex(df[df.columns[index]]).year
    df[units[1]] = pd.DatetimeIndex(df[df.columns[index]]).month
    df[units[2]] = pd.DatetimeIndex(df[df.columns[index]]).day
    df[units[3]] = pd.DatetimeIndex(df[df.columns[index]]).dayofweek
    df[units[4]] = pd.DatetimeIndex(df[df.columns[index]]).dayofyear
    df[units[5]] = pd.DatetimeIndex(df[df.columns[index]]).weekofyear
    df[units[6]] = pd.DatetimeIndex(df[df.columns[index]]).quarter
    df[units[7]] = pd.DatetimeIndex(df[df.columns[index]]).hour
    return df

mssrawq="""select dn, alarm_status, alarm_type, alarm_time,insert_time,cancel_time, acked_by, alarm_number, text,
user_additional_info, supplementary_info, diagnostic_info, lifted_dn, CASE
    when lifted_dn like '%MSC-397824%' then 'POOL1_MSJFR'
    when lifted_dn like '%MSC-913028%' then 'POOL1_MSJMR'
    when lifted_dn like '%MSC-397825%' then 'POOL1_MSMTH'
    when lifted_dn like '%MSC-373024%' then 'POOL2_MSPRL'
```

```
    when lifted_dn like '%MSC-373025%' then 'POOL2_MSPPE'
    when lifted_dn like '%MSC-372984%' then 'POOL2_MSPSO'
    when lifted_dn like '%MSC-913029%' then 'POOL3_MSDME'
    when lifted_dn like '%MSC-372857%' then 'POOL3_MSDNR'
    when lifted_dn like '%MSC-913031%' then 'POOL3_MSBEO'
    when lifted_dn like '%MSC-373001%' then 'POOL4_MSCTN'
    when lifted_dn like '%MSC-404656%' then 'POOL3_MSDTJ'
    when lifted_dn like '%MSC-373003%' then 'POOL4_MSCFO'
    when lifted_dn like '%MSC-372803%' then 'POOL4_MSPFS'
    when lifted_dn like '%MSC-404655%' then 'POOL2_MSPCE'
    when lifted_dn like '%MSC-395187%' then 'TAS_TSPSI'
    when lifted_dn like '%MSC-395188%' then 'TAS_TSMTA'
    when lifted_dn like '%MSC-399311%' then 'POOL5_MSIPS'
    when lifted_dn like '%MSC-399308%' then 'POOL5_MSIGM'
    when lifted_dn like '%MSC-326915%' then 'VCL_MSLKK01'
    when lifted_dn like '%MSC-396741%' then 'VCL_MSMSU01'
    when lifted_dn like '%MGW-326917%' then 'VCL_MGLKK01'
    when lifted_dn like '%MGW-396742%' then 'VCL_MGMSU01'
    ELSE 'Unknown'
    END as Node,Severity
from fm.fx_alarm
where alarm_time >= trunc(sysdate)-1
and lifted_dn like '%MSC%'"""
#one_records = cursor.fetchall()
#for i in one_records:
    #print(i)

tmp=pd.read_sql(mssrawq,con=db)

numClusters =4
pd.set_option('display.max_columns', 300)

tmp=pd.read_sql(mssrawq,con=db)

cols_to_keep = ['NODE','SEVERITY','ALARM_TIME','TEXT','ALARM_TYPE']

tmp = tmp[cols_to_keep]
tmp.dropna(inplace=True)
tmp = pd.concat([tmp,pd.get_dummies(tmp['ALARM_TYPE'], prefix='ALARM_TYPE'),
                 pd.get_dummies(tmp['TEXT'], prefix='TEXT')],axis=1)
tmp = decompose_date_time_features(tmp,2)

clusters = cluster(numClusters,tmp)
plot_all(clusters,'Clusters = {}'.format(numClusters))
```

APPENDIX M: ELBOW METHOD

```
distortions = []
K = range(2,10)
for k in K:
    X = tmp[tmp.columns.difference(['NODE',
                                  'ALARM_TIME',
                                  'TEXT',
                                  'ALARM_TYPE'])]
    KMeansModel = KMeans(n_clusters=k)
    KMeansModel.fit(X)
    distortions.append(kmeanModel.inertia_)

plt.figure(figsize=(16,8))
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```

APPENDIX N: RELIABILITY

APPENDIX N -
Reliability.msg