A REVIEW OF THE SIYAKHULA LIVING LAB'S NETWORK SOLUTION FOR INTERNET IN MARGINALIZED COMMUNITIES

Submitted in fulfilment of the requirement for the degree of

MASTER OF SCIENCE

at Rhodes University

Hilbert M. Muchatibaya

28 July 2021

Declaration of Authorship

I, Hilbert Munashe MUCHATIBAYA, declare that this thesis titled, “A review of the Siyakhula Living Lab’s network solution for internet in marginalized communities”

- This work was done wholly or mainly while in candidature for a research degree at Rhodes University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at Rhodes University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: H Muchatibaya

Date: 28/07/2021

List of Figures

List of Abbreviations

ACL **A**ccess **C**ontrol **L**ist

SLL **S**iyakhula **L**iving **L**abs

BI **B**roadband **I**sland

WAN **W**ide **A**rea **N**etwork

LWAN **L**ocal **W**ireless **A**rea **N**etwork

ICT4D **I**nformation **C**ommunication **4**(for) **D**evelopment

ICT **I**nformation **C**ommunication **T**echnology

SSL **S**ecure **S**ocket **L**ayer

TLS **T**ransport **L**ayer **S**ecurity

NAT **N**etwork **A**ddress **T**ranslation

WIMAX **W**ireless **I**nteroperability for **M**icrowave **A**ccess

DNS **D**omain **N**ame **S**erver

HTTP **H**yper **T**ext **T**ransfer **P**rotocol

HTTPS **H**ypertext **T**ransfer **P**rotocol **S**ecure

PEM **P**rivacy-**E**nhancement **M**ail

DAN **D**igital **A**ccess **N**ode

Abstract

Changes within Information and Communication Technology (ICT) over the past decade required a review of the network layer component deployed in the Siyakhula Living Lab (SLL), a long-term joint venture between the Telkom Centres of Excellence hosted at University of Fort Hare and Rhodes University in South Africa. The SLL overall solution for the sustainable internet in poor communities consists of three main components – the computing infrastructure layer, the network layer, and the e-services layer. At the core of the network layer is the concept of BI, a high-speed local area network realized through easy-to-deploy wireless technologies that establish point-to-multipoint connections among schools within a limited geographical area. Schools within the broadband island become then Digital Access Nodes (DANs), with computing infrastructure that provides access to the network.

The review, reported in this thesis, aimed at determining whether the model for the network layer was still able to meet the needs of marginalized communities in South Africa, given the recent changes in ICT.

The research work used the living lab methodology – a grassroots, user-driven approach that emphasizes co-creation between the beneficiaries and external entities (researchers, industry partners and the government) - to do viability tests on the solution for the network component. The viability tests included lab and field experiments, to produce the qualitative and quantitative data needed to propose an updated blueprint.

The results of the review found that the network topology used in the SLL's network, the BI, is still viable, while WiMAX is now outdated. Also, the in-network web cache, Squid, is no longer effective, given the switch to HTTPS and the pervasive presence of advertising. The solution to the first issue is outdoor Wi-Fi, a proven solution easily deployable in grass-roots fashion. The second issue can be mitigated by leveraging Squid's 'bumping' and splicing features; deploying a browser extension to make picture download optional; and using Pi-hole, a DNS sinkhole.

Hopefully, the revised solution could become a component of South African Government's broadband plan, "SA Connect".

Acknowledgements

Firstly, I would like to thank my supervisors Prof. Alfredo Terzoli and Prof. Nomusa Dlodlo for their guidance and support. Your insightful feedback helped me sharpen my critical thinking skills and pay attention to detail

I would like to acknowledge my colleagues from Rhodes University for the encouraging words and the support throughout the years.

I would like to thank my parents, particularly my father, Dr. Gift Muchatibaya. You provided me with the right tools I needed to choose the right direction for my thesis. Thank you for the constant support and checking up on me constantly

I would like to thank my friend Mwazvita, for the encouraging words and the support from the time I started writing my thesis.

Finally, I would like to thank my girlfriend Tino, for providing stimulating discussions as well as happy distractions to rest my mind outside my research.

Contents

1 Introduction

In developing areas of the world, citizens often lack physical access to the infrastructure necessary to connect to the internet and for example, communicate with the government [1]–[3]. South Africa has attempted to connect multiple state entities such as clinics, schools and police stations at multimegabit speeds through a plan codenamed 'SA Connect' [1]. Initially, the government hinted at a centralized point of control, with Telkom being the lead implementer[1], [3]. However, through years of experimentation, the model did not work in townships and rural areas, where it was expected to bring the most significant changes [1], [3], [4]. A different model is possible, in the form of a more flexible grassroot model. This model is the result of long-term engagement with marginalized communities of South Africa by two universities, Rhodes and Fort-Hare [1], [3], [4]. This research will focus on the grassroot model, its structure and its implementation in the Siyakhula Living Labs (SLL) and how it can be improved. This chapter is structured as follows. Section 2 gives some background on community networks and the Siyakhula Living Lab (SLL). Section 3 describes the structure and topology of the Broadband Island (BI) and its implementation. In addition, section 3 will give a breakdown of the technologies used in the model and the reason why they are used. Section 4 will outline the research problem and Section 5 will give an overview of the methodology.

1.1 Background and Context

1.1.1 Community Networks

A community network is a computer-based system built with the goal to advance social goals, improve citizen engagement with the government and developing economic opportunities in the disadvantaged communities [5]. Since Community Networks are community-based, it is very likely that community members share common goals, issues and needs [5]. Therefore, they create a participatory medium because everyone in the community has a stake. Community networks provide local information to community members through forums, which are based on contributions from community members. These contributions become part of the forum itself and can be later be used as a point of reference when acting on the needs or the issues of the community. The difference between these networks and

online communities is that they are in a specific physical space, therefore, they are context sensitive [3], [5]. In addition, the people are already part of each other's social networks, hence, all the system does is amplify the pre-existing social network [3], [5]. Furthermore, community networks aim to, make communities more cohesive by encouraging participation from the community members, reduce information asymmetry by ensuring that citizens receive high-quality, timely and reliable information and provide access to citizens with education and the appropriate training so that they can leverage the benefits of technology more effectively [5].

1.1.2 The Living Lab Methodology

Living labs are an environment and a methodology that supports user-driven open innovation within communities. Unlike the traditional top-down approach, in this methodology, the user forms an integral part of the innovative and creative process [3], [6]. The Living Lab methodology is an adaption to the high failures rates of the top-down or the bottom-up approach. This methodology is a hybrid of the bottom and top up approach to provide a more balanced approach to ICT4D projects. The underlying principle in this approach is co-creation, and this means that the experts and the local community members are both involved in the design and experimentation process. This is crucial because while the knowledge of the experts is crucial to addressing the more technical problems, the community members have a better understanding of the context i.e., the limitations of the environment and the needs of the community. Living Labs provide an environment where the government, industry experts, community members and academia can come together to provide solutions for the communities long-standing issues [1].

1.1.3 Siyakhula Living Lab

The Siyakhula Living Lab (SLL) is a long-term experiment in connecting rural communities and conducting research into providing sustainable, off the shelf, ICT infrastructure for rural communities in South Africa. The SLL consists of a partnership among the following entities; academia, industry, government and the community. These entities make up the ecosystem described below. SLL was initiated in 2005, through the support of Telkom Centres of Excellence (CoEs) to conduct postgraduate research at Rhodes University and the University of Fort Hare. The SLL living labs has the following objectives:

- Develop local economies by empowering SMEs (Small, Medium and Micro-Enterprise)
- Reduce poverty in marginalized communities
- Encouraging of people through empowering them to invest in their communities and themselves
- Integrating indigenous knowledge systems into the communities existing learning infrastructures and knowledge base.

The SLL believes it can achieve all of this by introducing ICTs in low income and marginalized areas. To ensure that the ICT solutions are properly grounded in the context, the SLL uses the Living Lab Methodology explained earlier.

Over the years, the SLL has developed a blueprint for a network solution in Dwesa. This solution consists of 3 components; the network component, the ICT-infrastructure component and the application/e-services component. In this work, we will focus more on the network component

The figure below was taken from [4], and it represents entities within the SLL ecosystem, how they interact and their dates of commencement.

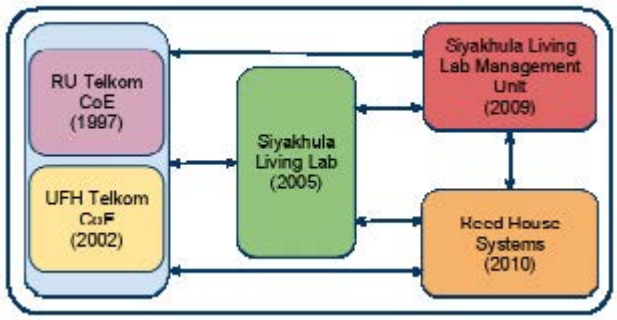

Fig 1: SLL Ecosystem

1.1.4 The Siyakhula Living Lab's original BI implementation

The network component of the Siyakhula Living Lab is based on a high-speed wireless local area network, called BI (BI), connecting a cluster of nearby schools and linked in redundant

fashion, but at much lower speed, to the Internet. The advantages of this approach to network design are as follows: it provides high speed communication between entities and members of the community within the span of the BI, facilitating local collaboration; and it allows the community to share access to the Internet in a cost-effective manner [1], [3]. The high-speed local area network allows also the sharing of common services. The most relevant service to the network component is a central cache, so that remote content is stored locally as much as possible to reduce bandwidth and data usage when reaching outside the BI.

Within this model, local schools house digital access nodes, which are computing infrastructure points of presence. In practice, the network component comprises routers, switches, wireless access points, antennas, WiMAX CPEs and in some cases micro base stations[1], [3].

The actual deployment has been using WiMAX technology from Alvarion to share Internet access. The decision to use WiMAX was based on the promise, in 2006, of a new technology, described in both academic and consumer's literature as "Wi-Fi on steroids". WiMAX offered solutions to the problems of outdoor use of Wi-Fi at that time. It could be used natively over much longer distances, without the need of a clear line of sight, supported higher data throughput and was less susceptible to interference because of running in licensed bands [1], [3].

The original deployment was divided into cells, each of them hosting a WiMAX micro base station (μbs). The WiMAX μbs are wireless access point and were located at two schools. The μbs's worked together with the BreezeMAX CPEs housed in the other connected schools. The two WiMAX base stations were mounted, together with 13 dBi omni-directional antennas, on 12m masts in the schools' property. These μbs's were then connected to a core router responsible for routing traffic from the schools where the μbs's are hosted to the Internet [1], [3].

The links to the Internet were realized through VSATs (Very Small Aperture Terminal, a satellite-based technology).

The web cache service was realized using Squid, a well-known proxy server at the time of the initial deployment. Its presence improved drastically network performance, helped by the homogeneity of the community served by the BI. Also, Squid allowed good control on

websites access, preventing for example access to inappropriate content, either because of its nature or because its impact on the links to the Internet.

1.2 Problem Statement

The current structure of the BI depends on WiMAX technology, which is no longer supported by the market. As a result, the BI will not be sustainable in the long run. Therefore, it is important that replacements are investigated, and alternatives are proposed. The investigation should consider the overall topology as well as the broadband telecommunication environment. Also, the web has changed substantially since its inception, with the transition to HTTPS as the transport mechanism and the dramatic increase in page weight. This needs adaptation of the cache services in the BI model.

1.2.1 Research Question

Given the current status of telecommunications in South Africa, Is the BI still a good solution to the internet needs of marginalized communities?

1.2.2 Research Objective

The research reported in this thesis has two main objectives, the first one is subdivided into 4 sub-objectives below. The second objective is interrelated to the first one and it includes proposing an updated blueprint based on the results from the first objective.

1. Reviewing the blueprint of the network component of the Siyakhula Living Labs project
    i. Reviewing the network topology of the SLL network solution
    ii. Reviewing the technology used to implement the SLL network solution
    iii. Reviewing the methodology used to design and implement the SLL network blueprint
2. Proposing an updated blueprint based on the review done in the first objective

1.3 Project Scope

As we stated earlier, the SLL solution is made up of three main components. ICT infrastructure, Network and application/e-services. The scope of this project is limited to the network component. However, where it is necessary, some discussions involving the other components will be included. As a result, significant contributions will only be towards components that constitute the network solution of the SLL. Some of the work done in this thesis, however, is not limited, in its applicability, to the SLL network solution, as will be seen in Chapter 5.

1.4 Methodology

1.4.1 Experimental work

The researcher engaged with communities and directly experiment with different wireless technologies. As proposed by existing literature, he was present at the field sites regularly, this was complemented by traditional lab work [3], [4].

1.4.2 Reflection

The experimental work, be it directly in the community or in a traditional lab, needs to be accompanied constantly by reflection, both individual as with collective. Continuous Reflection was present throughout the work reported here and informed the style of this report.

1.5 Document Structure

A diagrammatic representation of the structure and layout of this thesis can be seen in Fig 2 and is detailed later in these sections

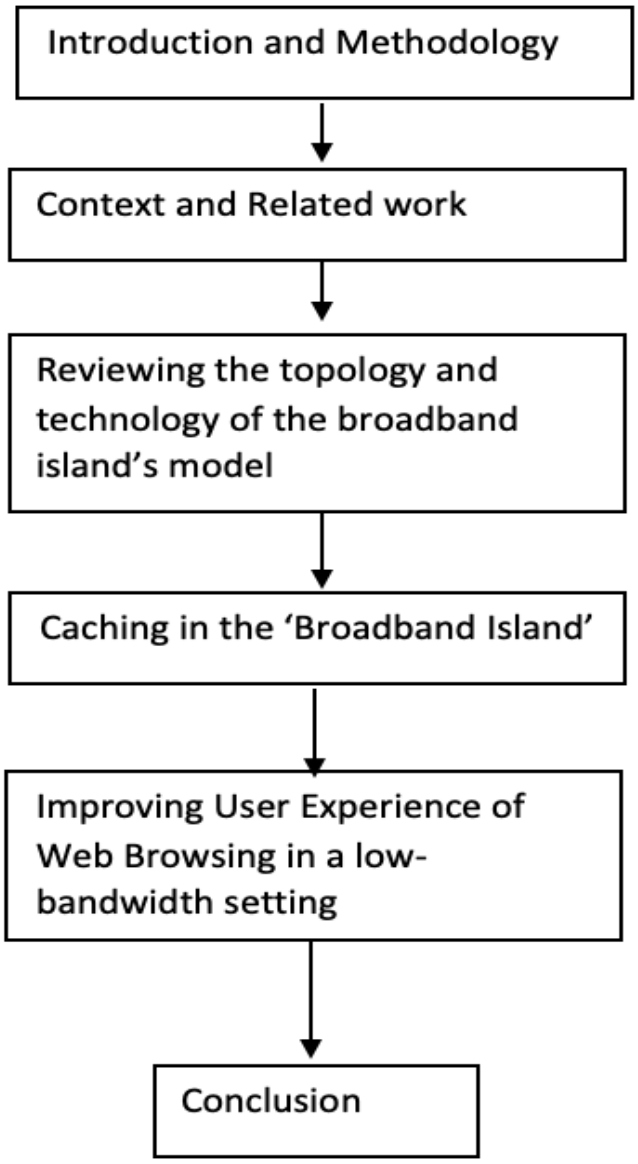

Fig 2: Document Structure

Chapter 2 - Context and related work:

This chapter is a discussion on the background and the context of the problem. The chapter explores existing literature and related work.

Chapter 3 - Reviewing the topology and technology of the BI's model:

This chapter is a review of the network component of the SLL solution. The chapter reviews the network topology and the technology used to implement it, proposing changes where necessary.

Chapter 4 - Caching in the BI

This chapter reports on the work conducted to help cope with HTTPS caching in the BI.

Chapter 5 - Improving User Experience of Web Browsing in a low-bandwidth setting

This chapter provides a deeper analysis to the problems discussed briefly in the previous chapter – the increase in page weight over the last decade. And proposes a browser extension as a solution to the problem.

Chapter 6 - Conclusion

This chapter concludes the thesis by reflecting on the lessons learnt during the project, proposing future work, and recommending that the blueprint provided by this work be adopted by the government on a national scale

2 Related Work

2.1 Introduction

Since the early 2000s, there has been a considerable increase in digital economies worldwide [7]. It would seem that internet adoption by either individuals or communities might foster economic growth. However, this is contingent on the availability of affordable and the accessible telecommunications infrastructure [2], [7]. The issue of the digital divide has received considerable attention from national and international policy communities, however, relatively little research has offered explicit perspectives on the rural digital challenges [7]. In 2015, the South African government embarked on an attempt to bridge this gap with a plan codenamed 'SA Connect'. The aim of the plan was to connect multiple state entities at multimegabit speeds with the goal of accelerating the countries arrival to a digital government. This plan was not a success, particularly in areas where it was expected to make the biggest changes – the rural areas. More flexible, grassroot models were proposed, however. One of the models was constructed through the work in the Siyakhula Living Lab. This model had as a centrepiece the concept of "Broadband Island". This model was transformed into a blueprint and made public through this plan, the BI blueprint was developed [1], [3].

Just as with any blueprint, the SLL network solution blueprint has to be revised to ensure that it is still viable in current times. In fact, at the moment, WiMAX - a core technology that was used to implement the blueprint is no longer supported by the market [3], [8]. Therefore, given the current status of telecommunications in South Africa, a reflection on the BI model with respect to the recent technological trends is important. In this chapter, we will review existing literature, related work and existing community networks, in order to collect knowledge relevant to our purpose.

This chapter is structured as follows, in the first section will give more background to the reader using the lens of eGovernment and what developing countries need to do to prepare for its inevitable arrival. The second section reviews the literature on the different components of the network solution in the Siyakhula Living Lab – the methodology or approach, the network topology and the network technology. Each of the respective components will have a section dedicated to them. Finally, we will conclude with a section

that summarizes our findings and how we will address the existing issues with our current network solution in SLL.

2.2 ICT4D and E-Government

E-government is can be described as the concept of using ICT to create more efficient and convenient interactions between the government and its citizens [9]. The success of the adoption of ICT in the private sector called for the need for the public sector to reinvent itself by transforming how things are done externally and internally [9]. The main issues and components that characterize an eGovernment frame work are:

1. Transformation areas
2. Citizens, businesses, government organizations and employees
3. eGovernment application domains (e-services)

It is now generally accepted that ICT offers increased opportunities for economic development and plays a role in rapid economic change. Studies supporting this have shown a significant relationship between the Gross National Income (GNI) per capita and the adoption of ICT [10], even going as far as suggesting that an adoption of ICT by one percent will bring about an increase per person in low-income households by approximately 2.8% [10]. In the African context, a study examined the causality between ICT and public sector management between 1995 and 2015 using GMM model and Toda-Yamamoto causality tests. The evidence seems to show that ICT has a positive and statistically significant effect [11].

Nonetheless, e-government introduces the risk to deepen the digital divide and to further marginalize the underdeveloped parts of a country as seen with the 'SA Connect' plan [3]. As a result, digital government has evolved more towards contextualization as we have seen that ICT solutions are more effective when there is a good understanding of the context - there is no one size fits all solution [12], [13]. To this end, successfully implementing a digital government means much more than introducing ICT to the government. It also includes, as stated earlier, transformation of government processes both internally and externally, citizen engagement and lastly policy driven electronic governance [9], [12]. These different components ensure that government initiatives are not generic solutions but align with the

needs of their environment and are capable of adjusting to the ever-changing landscape of digital governance[12].

2.2.1 Digital Divide

The digital divide is an issue that can start to be addressed by building ICT infrastructure in marginalized communities [13]. However, if the introduction of ICT in any environment is not coupled with an understanding of the context, it might exacerbate the problem. First, we will discuss three main issues that must be considered in order to bridge the digital divide: accessibility, availability and affordability, all within the specific of the context.

2.2.2 ICT for Development and Sustainability

Sustainability should play a vital role in ICT4D in projects. Project sustainability can be defined as its resilience or ability to absorb disturbances [14]. By engaging more closely with the beneficiaries of a project and encouraging more and better communication with the public ICT4D projects can continue to receive funding and realize their influence on the international community [15]. In order to achieve this, it is important that researchers adopt implementation methodologies that are user driven and encourage participation from the beneficiaries and the public [15].

In addition, funders increasingly require to evidence of the socioeconomic impacts when they invest in ICT4D projects. Studies have shown that the majority of researchers in this domain are not interested in communicating their findings to a wider audience. In addition, their research and development approach does not involve adequate user engagement. Academic research needs to conform more to the requirements of achieving economic and societal impact if it is to attract more funders, therefore improving the sustainability of ICT4D projects.

In the following section, we focus more on the methodology and how it can impact the success of failure of an ICT4D project

2.3 Network Design Approach in the ICT4D Domain

The high failure rates in the ICT4D domain has fuelled a sense of scepticism towards the topic and has given rise for the need to revise the developmental approach when handling projects within this domain [16], [17]. The traditional approach i.e., the top-down approach has failed repeatedly and one of the common criticisms has pointed more to the human side of ICT interventions. For example, there seems to be a lack of understanding of the social context by the experts and participation from the community members is minimal to a fault [16], [17]. This section explores the different types of developmental approaches, which include the traditional top down approach, the bottom up approach and the process approach [16].

2.3.1 Top-Down Approach

The top-down approach goes from general to specific and because it has a managerial perspective, it gives the stakeholders an overview of the project. Consequently, the manager has more clarity and control over the scope [16]. However, this comes at a cost, it introduces bottlenecks because the decision making process is delegated to a select few and it hinders collaborative efforts because team members have to adhere to imposed processes, therefore, there is general little individual motivation because they feel like their ideas do not matter [16], [17]. One can draw a parallel between this and the approach used in peebles valley project discussed later in this chapter. The developmental approach in this project was mainly top down and for that reason the project experienced many setbacks, due to the lack of collaboration between the experts and the community members [16]. Because the community members were not involved in the decision-making processes, they were not inclined to understand how to operate the network to make it sustainable in the long run. In addition, because the experts did not fully immerse their work within the social context, they failed to create incentives for any socio-economic growth [16], [18].

Another example that illustrates the problem with a top down approach point is the ‘SA Connect’ plan that was initiated by the South African Government, in an attempt to electronically connect state entities, the majority being schools [19]. The decision to make Telkom, the former telecommunications incumbent in South Africa, the lead implementer introduced a centralized point of control. Moreover, this suggested that the project would use the traditional top down approach [19]. Two years after the project had been announced it

was clear that there was very little progress made in the rural communities where it was expected to produce the biggest changes. Not only does the idea of the 'lead implementer' introduce a bottleneck, it also creates a single point of failure within the project [16], [19]. One should remember that communication infrastructure does not only differ from country to country but also within different parts of the same country. The infrastructural needs vary within each tier of the economy; therefore, a uniform infrastructure is not a good proposition because each tier has different needs and requirements [2], [16], [19].

2.3.2 Bottom-up approach

A bottom-up approach promotes distributed and diverse architectures. The advantages of this approach include the removal of a central point of failure and bottlenecks [16], [19]. A bottom-up approach is detail oriented and that makes it a favourable approach when dealing with problems that are context sensitive [2], [7], [16]. However, it has its own disadvantages: the project could lack cohesion because of decision-making made at difference points within the project. In addition, the decision-making process is made more complicated because it is difficult to reach a consensus when dealing with diverse groups of people [16].

Guifi.net, described later in this chapter, is an example of a project that used the bottom-up approach. The guifi.net is a citizenship-driven project with the aim of creating free, open and neutral telecommunications network [20], [21]. The project is a result of collaboration between four groups of participants: i) volunteers in the community ii) experts in telecommunications, demand and service supply iii) customers in need of affordable means to connect to the internet iv) public administrators [20]. These participants are the stakeholders of the project, therefore, their participation in the decision-making is crucial, especially for the sustainability of the project. In addition, network participants have the right to use their preferred means of connectivity to the network, which means each participant has the freedom to purchase equipment they know they can afford [20].

While this approach is ideal in the case of Guifi.net, the same does not hold true in certain communities, e.g., the Dwesa Community in South Africa the original site of the Siyakhula Living Lab. In Dwesa, the majority simply cannot afford to purchase the equipment. In addition, the Dwesa community does not have pre-existing infrastructure they can leverage when building their own community network. This means that there is a need for the government to intervene at some level and contribute to the initial investment [17]. Based on

this observation, one can infer that places like Dwesa will benefit from leveraging both the top-down and bottom-up approach [16]. It is worth noting that it would be very misguided to suggest that a better approach is the only feature required to ensure success of a development project. Many aspects contribute to the overall effectiveness of development, e.g. the laws and national policies [17].

2.3.3 The Process Approach and The Siyakhula Living Labs Approach

The process approach emerged from the shortcomings of the traditional top-down approach and its inability to adapt to the different complexities of the varying projects [16]. Fundamentally, this approach is flexible and adaptive and focuses on learning through experimentation. The process approach has the following components: i) beneficiary participation ii) flexible phased implementation iii) learning from experience iv) institutional support v) programme management [16]. It is clear that the BI model borrows a lot from this approach because it shares similar components.

The BI model was developed and tested in the Siyakhula Living Lab, which is a long-term experiment in connecting the unconnected. The living lab methodology is an approach that deals with user driven innovation i.e. beneficiary participation [19]. These beneficiaries include academia, industry, government and community members [3], [19]. Based on this, it is clear that the process approach and the SLL approach are very similar in general. Of course, they differ when we consider specifics such as the fact that the SLL approach includes a business model on the application layer through providing e-services. However, the key takeaway is the fact that they both focus on collective learning based on the outcome of failures, which is at the foundation of this research.

2.4 Network Topology

This section will discuss the various network topologies highlighted in the current literature. The network topology is an arrangement of nodes and links in a communication network. There are currently two approaches to the construction of the community networks. The first approach requires thorough network planning with carefully chosen antenna configurations and IP addressing structures to engineer high-quality links with good throughput [18]. The second approach makes use of a ad hoc type of network in which community members join

the network using easy-to-use, scalable hardware communicating using an agreed mesh routing protocol [22]. The literature on these two approaches will be presented in this section

2.4.1 Community Wireless Mesh Networks

The aim for community wireless mesh networks is to provide sustainable bottom-up implementation of wireless networks in remote communities. A mesh network is a network in which the nodes within the network are connected directly to each other dynamically and in a non-hierarchical manner.

2.4.1.1 Zenzeleni Network – Mankosi Village - South Africa

In South Africa, a partnership between the Mankosi community and the University of Western Cape was started to provide cheaper voice services. The network topology consists of 12 points of presence, which use a network device called a Mesh Potato . A Mesh potato is an easy-to-use, scalable and standards-based Wi-Fi device that connects to other Mesh potato devices to form a network [13], [23], [24]. Each of the points have analogue phones installed for those without mobiles phones. The network is operated and maintained locally and the University of Western Cape provided the initial funding which was raised to acquire the hardware and training for its installation[25]. Revenue is provided by charging community members for charging the battery of their devices using the excess electricity generated by solar panels installed to power the network infrastructure the revenue covers the maintenance costs [25.]. In addition, Mesh Potato devices are relatively easy to operate for low skill technicians; as a result, the operational costs are reduced significantly. Thus, the source of income is enough to sustain the business model.

2.4.1.2 TakNet – North-West Thailand

In Thailand, a cost-sharing model named TakNet allows remote communities to share bandwidth as a way to provide affordable access to the commercial internet and communicate with other community members [26], [27]. TakNet and Village Telco have a few commonalities, which are 1) the technology is simple enough to get local low skill technicians to maintain and 2) the monthly cost is affordable for low-income users. The topology consists of mobile routers running optimized link state routing protocol they sharing

the internet access with one of two ADSL gateways. One of the limitations TakNet has is that it introduces the hidden terminal problem. The hidden terminal problem occurs when a node can communicate directly with the access point but cannot communicate with other nodes within the same network. Despite this limitation, the TakNet model offers possible solutions to long standing problems in other community networks. For example, The TakNet model implements caches for storing video content locally using either high-capacity dongles attached to mobile routers/access points or raspberry Pi's where the users access the contents via web servers running either raspberry Pi or web server-enabled access points [27]. The idea to store remote content locally is a common one in community networks as it reduces the rate of data consumption within the network. Typically, a community network would use an in-network cache to perform that function, but in this case, TakNet uses high-capacity dongles.

2.4.1.3 Peebles Valley Mesh – South Africa

The Peebles Valley network is located in a poor remote community named Masoyi Tribal Land. The network consists of nine nodes deployed over an area of about 15 kilometres and a VSAT setup at an AIDS Care Training and Support clinic located in the area [22]. The VSAT link provides 2GB per month with a downlink and uplink of 256kb and 64kb respectively. According to Johnson and Roux, approximately 60% of the available bandwidth is used per month; therefore, the bandwidth is underutilized [22]. The network was set up to allow VoIP calls between the hospice and the clinic which resulted in the community saving roughly $400 per month. The network used Linux based systems to minimize the risk of virus attacks as well as reducing the cost of PCs. In addition, the internet connectivity was used to look for jobs and grow community members' IT skills [22]. As a result, the idea of connecting the school to the mesh network became a very logical step. However, departmental issues within the school resulted in heavy pushback [22].

The system faced a number of challenges, which included the lack of collaboration between the experts and the community members [22]. As a result, the hand over process became difficult as there was a lack of investment from community members and they had no idea how to maintain or operate the network. In addition, making the internet in the community affordable for the community members proved to be difficult [22].

2.4.1.4 Guifi.net – Catalonia - Spain

The guifi.net is grassroots, citizenship-driven model with the objective of creating a free and open telecommunications network. The guifi.net is the largest community network in the world with over 40 000+ nodes [28], [29]. This community network is self-managed and self-sustained. The project started in 2004, with the objective to solve the broadband internet access difficulties in rural areas [29]. Given the lack of traditional infrastructures in rural areas, radio links built with commodity Wi-Fi, individuals within the community deployed their own network to interconnect to different locations. This resulted in a network that consists of heterogeneous devices with a common objective of connecting and increasing network coverage [28], [29]. The community network is based on the following principles: sharing infrastructure, the presence of economic activities and presence of the professionals the network is open, free and neutral [28], [29]. Sharing infrastructure increases the efficiency of the network and significantly reduces costs, which is essential in low-income communities [19], [28], [29]. In addition, the presence of economic activity ensures that the network is sustainable because it generates the revenue for maintenance [20], [28], [29].

2.4.2 Community Wireless Local Area Networks

2.4.2.1 Siyakhula Living Lab – South Africa

The Siyakhula Living Lab's network infrastructure uses WiMAX technology from Alvarion to share internet access [3]. The decision to use WiMAX was based on the promise, in 2006, of a new technology, described in both academic literature and to the consumer as "Wi-Fi on steroids". WiMAX offered solutions to the problems of outdoor use of Wi-Fi; namely, it could be used over longer distances, without clear line of sight, supported higher data throughput and was less susceptible to interference [3], [30]. In addition, WiMAX had relatively low costs associated with its deployment due to the fact that the hardware was provided by one of our industry partners, SAAB Grintek, therefore, making it an economically viable option [19].

Internet access in the BI was realized through VSAT solutions from YahClick/Vox and SAAB Grintek. VSAT and WiMAX are complementary when put together because WiMAX

will allow the sharing of internet access between schools that are geographically dispersed at high speed through point to point links while VSAT offers internet access at a relatively high through-put with no need for established infrastructure [19]. The BI was divided into three root node cells and two of the root node cells host a WiMAX μbs (micro base station). The WiMAX μbs is a link layer device, therefore, it's primarily a wireless access point to the BI. The μbs's work together with the vertically polarized BreezeMAX CPEs (BMAX-CPE-ODU-AV-3.5) which were housed in the other 17 schools [1], [3]. These CPEs provided the point-to-multi-point connection from the school housing it to the μbs's and vice versa. The μbs's in use in the SLL BI were Alvarion BreezeMAX 3500 both μbs's are fixed-wireless WiMAX units (802.16d) [1]. The two WiMAX base stations have a point to multi-point configuration with Omni-directional antennas radiating signal uniformly over a 360 degrees angle. The two WiMAX base stations used 13 dBi omni-directional antennas with the μbs's radio interface on 12m masts on schools property [19]. The μbs's are each connected to a core router which is responsible for routing traffic from the schools where the μbs is hosted to the internet [19].

The advantages to this approach to network design are as follows: it provides high communication channels between entities and members of the community; remote content is stored locally to limit the bandwidth usage when trying to reach sites outside the BI i.e. caching. It so it allows the community to share internet access therefore reducing the costs significantly and facilitating collaboration and cooperation between members [1]. Local schools within the BI houses digital access notes which were computing infrastructure points of presence [1], [3].

2.4.2.2 Venezuela

Although the 802.11 a/g/n standard is ideal in local area networks based on its design, it proves to be a cost effective alternative for wide area networks in poor communities [31] . The major limitations associated with using the 802.11 standard over long distances include the requirement of line of sight between endpoints and vulnerability to interference of the unlicensed spectrum [31], [32]. In addition, the researchers considered power budget limitations. Flickenger et al, found solutions to these limitations and a successful 279km link made in Venezuela by wireless experts.

To overcome the first limitation of using Wi-Fi to make a long distance point-to-point connection, the researchers searched for a terrain with high elevation at the ends and low ground in between [31], [33], [34]. The search resulted in the researchers settling for a location in Andes which proved to be adequate for the task and had clear line of site. With the first limitation resolved, the researchers had to account for signal attenuation between the two ends which are 279km apart. To overcome this limitation, the researchers used 30dBi antennas on both endpoints [35].

To make Wi-Fi more suitable for long distance applications, the TIER group led by Dr. Eric Brewer from Berkeley University modified the Wi-Fi MAC layer. The modification includes replacing the CSMA Media Access Control with TDMA [36]. Since TDMA does not require the reception of ACKs, it is a better alternative for long distance point-to-point links.

2.4.2.3 India

Over the years, India has achieved success in IT services and business process outsourcing. As future economies become more knowledge driven, India has aimed at making every citizen have access to the internet [2]. Gunasekaran explores how communication infrastructure differs based on context, e.g. urban areas vs rural areas [2]. However, in this section, I will focus on the strategies used in rural areas/poor communities with a dispersed population.

The social and economic value of a rural Wi-Fi network is proportionate to the number of groups in the given rural community. Many villages are remote and potential access nodes within the community will likely be several kilometres apart. Therefore, the cost of a village network is determined by the choice of backhaul between these different nodes [2]. An entity in this context can either be a school, a clinic or community access infrastructure. Typically, villagers cannot afford any kind of personal communication device or the subscription fee to access network infrastructure [2]. VSAT is ideal for providing commercially viable connectivity in these hard-to-reach regions. In addition, VSAT and Wi-Fi are complementary when combined. In areas where community members, clinics or schools are geographically dispersed, VSAT provides an ideal backhaul between the different entities in the communities [2].

Each village houses a VSAT system in a kiosk. These VSATs provide network connectivity to end-users within a 1km radius. In each location, there is Wi-Fi outdoor access point that

connects on a satellite channel of 512kbs – 2Mbps for downlink and between 516-256 kbps for uplink [2].

2.4.3 Software Defined Networks (SDN)

The software-defined network is a step away from the traditional network approach to networking described above. With the traditional approach, we have a distributed control plane consisting of switches and routers that handle all traffic based on network policies and routing protocols [37], [38]. However, with SDNs, we have a centralized control plane that sends instructions to the rest of the network. With this approach, the data plane and the control plane are decoupled. The routers and switches become packet forwarding hardware that receives routing instructions from a central control point [40].

This approach to networking was designed to improve the agility and scalability of network solutions [40]. It is common knowledge that ICT solutions are very context-sensitive, and we need to take a case-by-case approach when designing solutions for different environments. Suppose SDNs are successfully adopted in rural communities. In that case, we can hone their potential to provide one homogeneous solution that can easily be scaled across different contexts from a hardware and software perspective.

2.5 Network Infrastructure

2.5.1 Backhaul Infrastructure

The backhaul portion of the network comprises the intermediate links that join the small subnetworks at the edge of the network to the core of the network. Commonly, rural areas are in remote locations that are hundreds of kilometres away from fibre backbone and there are two main reasons for that [2]. As mentioned, the first reason is that there is no incentive for profit driven organisations because there are low prospects of generating revenue in these communities. Secondly, in the case of South Africa, installation of fibre infrastructure within certain areas would require digging through areas prohibited [2], [3], [34]. As a result, wireless alternatives like VSAT, Wi-Fi and WiMAX offer most benefits because of their low initial investment and their flexibility [35]. The following section will discuss VSAT, WiMAX and Wi-Fi in finer detail to highlight how they work in different scenarios.

2.5.1.1 Very Small Aperture Terminals (VSAT)

VSAT offers a commercially viable backhaul solution especially in geographically dispersed communities [2], [3], [39]. VSAT eliminates structural issues in data transmission because they transmit the signal via orbital satellite, as oppose to a physical medium e.g. Ethernet connection [2], [3], [39]. Furthermore, the technological advances in VSATs have simplified installation, antenna size and lowered space requirements.

Poor communities have poor mobile network coverage. Gwaka highlights in his study that the rural communities in Zimbabwe have access to 29% of the country's base stations [40]. This is the common setup in most poor communities and many researchers have favoured the use of VSAT instead. A good example would be the Dwesa community in South Africa where the researchers opted initially to use VSAT because it offers greater and more stable throughput than the mobile networks in the community [19].

The main limitation VSAT has is latency. This is because one part of the system is way up in the geosynchronous orbit above the earth. This means that applications or protocols that require synchronous communication experience lag [41]. Nonetheless, this does not outweigh the fact that once it is coupled with other technologies like WiMAX or Wi-Fi it has the potential to provide underserved areas broadband connectivity at an affordable cost. In other words, the benefits outweigh the cost; therefore, the poor latency is a fair trade off [2], [42].

The BI model used in SSL, uses the VSAT to provide internet access to the wireless community network in Dwesa. This decision was because VSAT offers greater throughput than the unreliable cellular connectivity in the area [1], [3]. However, there is some contrast about how VSAT is used in India and South Africa. According to one of the researches in India Gunarasekan, VSAT is more suitable as a backhaul between nodes within the community [2]. The contrast further supports the idea that ICT infrastructural choices vary drastically depending on the location and in most cases socio-economic factors [7].

2.5.1.2 World Interoperability for Microwave Access (WiMAX)

WiMAX is based on the 802.16 IEEE standard and orthogonal frequency division multiple access is the method of sharing communication resources with a large number of users. It is worth noting that OFDMA is a variation of Frequency division multiplexing (FDM) [30], [43]. FDM divides the available bandwidth into different non-overlapping sub channels in

order to allow multiple users to communicate over a single network. In between the non-overlapping channels, a guard band of a narrow frequency is there to ensure there is no interference between the channels [30], [43]. On the other hand, the OFDMA the sub channels are closely spaced and there are no guard bands between them. The reason why there is no interference among the overlapping signals is that there are orthogonal, which means that the signals are transmitted in such a way that the peak of each signal happens at the null of the others; therefore, there is no interference. At the end of the signal, a demultiplexer then separates them based on this orthogonal feature [30], [43]. It is clear why OFDMA would be better in community networks than FDM because it better utilizes the available bandwidth thus offering higher data transmission. However, its potential disadvantage is that is requires extra power because it is always on and ready for transmission [30], [43].

2.5.1.3 WiMAX mesh

This is a network architecture where nodes can communicate with each other via multi-hop routing and forwarding [13], [44]. The characteristics of a wireless mesh network is self-organization, self-configuration and self-correction. These characteristics enable flexible integration, quick deployment and easy maintenance. In contrast to ad hoc or client mesh, nodes in this type of network do not forward packets [13], [44].

One of the advantages of WiMAX mesh is its potential to reduce operating expenditures significantly when used for backhauling. WiMAX mesh infrastructure can aggregate all wireless backhaul links into higher capacity lines as a way to reduce backhaul cost [13], [44].

2.5.2 Long distance Wi-Fi

Wi-Fi stands for wireless fidelity and it belongs to the IEEE 802.11x family of standards and these standards are mainly for short range networks [34]. The coverage area of the normal access point is 20 meters indoors and 100 meters outdoors [45]. Wi-Fi technology requires the point-to-point connections to be within line of sight and there is a higher probability of interference as it mainly operates in the unlicensed 2.5 GHz spectrum [2], [46]. It is clear that the standard is suited for short-range, indoor networks. However, a number of studies have proposed ways to overcome these problems and provide practical uses of Wi-Fi outdoors [34]. It is worth noting that the main motivation is the cost effectiveness of Wi-Fi technology

as compared to WiMAX and wired architectures [47]. This cost effectiveness is in part due to the enormous success of Wi-Fi that has allowed for economies of scale [35]. As a result, Wi-Fi technology infrastructure can be set up for a low initial investment and it is cheaper to maintain in the end because of operational costs since it does not require high skilled technicians to maintain [24], [48]. Therefore, it is a commercially viable solution for poor communities. With careful planning and optimal use of antennas, 802.11 technology works efficiently outdoors and over long ranges. An example is in Venezuela, where a group of researchers connected two points that were 279 km apart [35], [36].

The two major limitations of using Wi-Fi over long distances are that the endpoints have to be in line of sight and the unlicensed band is vulnerable to interference. To address the first limitation, researchers suggest taking advantage of the terrain elevations to overcome obstacles. In addition, moving over to the less crowded 5 GHz spectrum can be a solution to the second limitation. However, in rural areas, the 2.5 GHz spectrum is not as overcrowded so there might be no need for the second step.

2.5.3 Antennas

The two main types of antennae mentioned in the literature are omnidirectional and directional antennas. Antennas provide a means for radiating and receiving radio waves within a network; therefore, their setup process is of paramount importance [33], [48]. Omnidirectional antenna and directional antennae behave differently and they both have different physical layer requirements. Jain et al, states that point-to-multipoint links are more suitable with omnidirectional antennas, while point-to-point links employ directional antennas [48]. Typically, in a WLAN, the central node or core node has an omnidirectional antenna as it receives signals from many directions while the nodes at the edge of the network have a directional antenna. In addition, environmental conditions also need to be considered when deciding the type of antennas to be used in a network [33]. Antenna directivity and height are the most important factors when trying to increase coverage range [33]. When setting up a WLAN network in a rural community, it is important to focus more on increasing backhaul distance more than increasing coverage area. This is intuitive given the fact that most literature highlights the fact that these communities are geographically dispersed and have a low population density.

2.5.4 Sharing communication resources in WLANs

In remote communities, one of the limiting factors is the lack of infrastructure to build the core network. In this context, there is no incentive for telecommunication companies to invest in these communities [38]. These companies cannot apply their business models in the rural communities with the prospects of high returns because a significant portion of the population consists of low-income households [38], [49]. Because of the low income, internet connectivity is an unnecessary expense because individuals do not have an excess of disposable income after spending on basic amenities. Therefore, there is no incentive in the grassroots either. Which is why sharing network infrastructure is a driving principle in building WLANs because it lowers the barriers of entry for low-income households through economies of scale [38], [49]. In addition, there is an increase in the efficiency because there are sno duplicated efforts across the network. The shared resources include the access infrastructure needed for individuals to connect to the core of the network and the shared link layer medium used to connect multiple users to the network at any given time [38], [49].

2.5.4.1 Multiplexing

The most common link layer technology used in community networks is Wireless LANs. Wireless LANs are an example of broadcast link-layer technologies. In a broadcast network, multiple nodes are communicating with another nodes within a shared medium, therefore, it is important to have a systemic manner or a protocol that handles how each of these nodes communicate in the shared medium [43], [50]. The solution for this common problem; the multiple access problem, is multiplexing. Multiplexing is the process of combining multiple signals into one signal, over a shared medium. This allows wireless operators to maximise the use of their spectrum to accommodate a large number of users over few channels. Each link layer technology has its own unique multiple access method that is based on the IEEE standards [43], [50].

2.6 Data Management in the BI network

The use of proxies to minimize bandwidth consumption and improve latency is a common practise within community networks. In-network proxies store remote content locally so that instead of sending the request over the internet or intranet, the network cache fulfils the

request. For example, in the Siyakhula Living Lab, the BI used SQUID proxy to cache frequently visited sites and that ensures that the WAN bandwidth usage is not excessive. However, in-network caching was very useful in the Web 1.0 days, and that is mainly because most web pages were relatively static. The ever-changing trend that are affecting web proxies are dynamic content, HTTPS, increase in page weight and streaming media. In this section, I will expand more on each of these sub topics to paint a clearer picture and then discuss the alternatives to caching given the current trends in technology.

2.6.1 Web Trends affecting the use of web proxies

2.6.1.1 Dynamic Content

As websites have moved from relatively static web pages to rich media applications. This change coupled with web 2.0, which emphasizes user-generated content and interoperability of users has resulted in a significant shift in web traffic [51], [52]. Understanding this shift is not only important for overall system design but it is important for analysing the redundancy and effectiveness of caching. This understanding will shape the design of web server, proxies and browsers to improve response times and bandwidth consumption [51], [53].

The rise of AJAX and streaming content has affected a number of different traffic measures. AJAX has caused an increase in the sizes of the JavaScript and CSS files and browsers increased their simultaneous connection limit to better accommodate it [51]. Flash videos have dominated traffic, pushing the share of other video formats lower and increasing bandwidth consumption. In addition, the majority of web traffic is a result of the client-side interactions after the initial page load.

2.6.1.2 HTTPS

Due to the increase in user concern over security in the web, there has been a worldwide adoption of HTTPS; the secure version of HTTP [54]. HTTPS runs on top of the Transport Layer Security protocol (TLS) or the Secure Socket Layer protocol (SSL) [7]. These protocols give web servers the ability to provide immunity against Man in the Middle Attacks (MITM) by authenticating the communicating ends [6, 7]. In addition, HTTPS uses encryption to enforce data confidentiality if the information is intercepted in transit. Even though HTTPS was initially aimed towards services that require data authentication and or data confidentiality e.g., e-commerce websites, the increased personalization of the web

through social media has resulted in more services adopting it [7]. As a result, the majority of websites use HTTPS. Network proxies can filter through network traffic between a client and a server and they can intercept the packets. This mechanism is useful from a bandwidth management perspective because it enables the proxy to serve cached content based on the user's request. However, from a security stand point; this mechanism introduces many risks because this mechanism is analogous to the man-in-the-middle attack. Hence, the major adoption of HTTPS in recent times. Consequently, any in-network value added service that requires viewing the application layer content such as , have become ineffective.

2.6.1.3 HTTP/2 Server Push

HTTP/2 was standardised in 2015 to replace HTTP/1 [55], [56]. The key difference between the two protocols is that HTTP/2 introduced Server Push, which replaces the classic request/response model. The request/response model would have the browser requesting a base document and then parsing it while requesting all discovered objects individually [55], [56]. In contrast, the HTTP/2 can push objects without an explicit request from the browser e.g., the server pushes the CSS files and image files upon the request of the index.html file[55], [56]. Hence, it allows transferring resources before the browser finishes parsing [55], [56]. Before the push, the server announces information about the object and stream it will use and then it sends the data. The server will not push data from an origin it has no authority over and the client has an option to cancel any unannounced push. This is an important feature because of Cross Origin Resource Sharing (CORS). CORS is a sharing mechanism that allows restricted resources on a web page to be requested from another domain. This mechanism is a popular technological trend utilized by many websites in recent time. To deactivate the Server Push feature, the client can change the SETTINGS_ENABLE_PUSH header to 0 at connection start-up [55], [56]. In addition, Server push can improve page load time (PLT) by reducing the round-trip times (RTTs) [55], [56].

In an experiment done on the top-100 websites according to Alexa, <20% of the websites had pushable objects, i.e., the other objects reside on servers beyond their authority. Hence, many websites cannot push all objects. The study further revealed that pushing all objects can be harmful, as it can delay processing and use unnecessarily waste bandwidth. Lastly, certain objects; particularly images, lead to a worse Speed Index for 74% of the websites because they do not contribute to the creation of the Document Object Model or the CSS Object Model. In conclusion, due to the adaption of CORS, a large number of resources on websites

are not push able. In addition, websites can benefit from different object types being pushed and pushing images usually results in negative effects. Overall, there is no standard way of pushing objects that works for all websites and it has to be on a case-by-case basis.

2.6.1.4 Content Delivery Networks (CDN)

A CDN is a geographically distributed network of proxy servers and their data centres. These servers help delays by reducing the physical distance between the server and the user [57], [58]. However, it is worth noting that the problem caching was solving is two-part and CDNs only address the issues concerning the page load time and even that is arguable considering page load time is also affected by the user's bandwidth. CDN do not improve bandwidth consumption but they improve latency [57], [58].

2.6.1.5 Streaming Media

Video streaming is a major source of internet traffic in modern times and usage continues to grow rapidly [59]. To cope with this, ISPs used caching to reduce the traffic congestion and manage bandwidth. However, due to the potentially wide attacking surfaces in untrusted networks, encryption protocols like HTTPS have become popular. The HTTPs protocol either fall, short of fully leveraging in-network caching or require decrypting the traffic in the middle with no guarantee of end-to-end security [59].

2.6.2 Alternatives to caching

Network and Browser caches were introduced primarily to improve latency and reduce bandwidth usage [54]. The issues regarding latency and bandwidth usage remain prevalent despite the evolution of web technologies [52], [54], [59]. It is well understood that the reason for this is that the problems are typically user related. For example, the fact that the world has shifted to a digital society has made the internet a basic commodity for every individual regardless of the socio-economic background. As a result, poorer communities that initially did not have internet access are getting it and it is clear that they stand to benefit more from caching. In addition, this shift has also made the prospect of digital government in the near future plausible. Therefore, it is clear that while the recent technological trends have made proxy caching obsolete, they have also introduced better and more effective alternatives to proxy caches. However, it is important to investigate these alternatives and decide which ones are more effective and applicable in the marginalized communities [54].

The ideal alternative should be able to provide efficient delivery of the encrypted content while preserving the benefits of in-network caching, which are managing bandwidth consumption and reducing latency. In addition, they should also be applicable within the constraints of a rural community. This means that the solution has to be scalable, relatively easy to configure and the community members should not incur any extra costs.

2.6.2.1 Browser Extensions

Browser extensions are a piece of software than allow the user to modify their browsing experiences based on their preferences. An example is an Adblocker extension, this extension filters out unwanted adverts as the client is browsing the internet. The architecture of a browser extension consists of three parts, the background scripts, the content scrips and the manifest file [60] [61]. The background scripts handle events triggered by the user on the front end, these events include clicking on a link or opening new tab. Background scripts and content scripts can interact with messages [60] [61] [62]. The primary goal of content scripts is to manipulate the document object model (DOM). Lastly, the manifest file is the configuration file of the extension [60] [61]. This file contains details about the extension that are crucial for the browser to run the extension. The manifest files are the backbone of browser extensions because they contain paths to important resources the extension will use and permissions to important resources [60] [61].

Over the past decade, there has been an 18.1% and 47.0% increase in image sizes on desktop and mobile webpages respectively, according to http archive [63]. With broadband speeds increasing, every year developers have added richer content to their webpages and this content includes images and videos. As a result, the average page weight over the decade has increased and it is known that page weight has a direct relationship with bandwidth consumption and page load time, which all affect the user experience. Therefore, users can have access to a browser extension that can effectively unwantedly images or videos; it has a positive effect on the user experience. At the moment, there are a number of browser extensions that are attempting to address this issue. The three most popular image blocker extensions are Fast Image blocker, Image Blocker EX+ and Block image. However, these extensions have very crucial limitations that have resulted in negative user reviews. The first limitation is that they do not actually block the images from being transferred over the network. This is crucial because the primary goal of the user when they block images is to manage their bandwidth and if the extension does not meet that objective, it is ineffective.

The second limitation is that they are only limited to images and they do not also block videos and this is crucial because videos have contributed significantly to the increase of page weight over the years.

Based on the failures of the existing extensions, it is clear that there is a criterion for an effective image blocker extension in marginalized communities. The criterion is that the extension has to block or cancel all image requests before the content is transferred over the network. Secondly, the extension should not only block images but all forms of media that constitute a significant portion of the page weight e.g., videos. An extension that meets this criterion is ideal because its primary goals will align with that of a proxy cache.

2.6.2.2 Quality of Service (QoS)

QoS is a set of technologies that work on a network to guarantee its ability to effectively run high priority applications and traffic under limited network capacity [64], [65]. These technologies can be used to set in place policies that will prioritize important traffic over streaming media, large file downloads or less critical sites. Having these network shaping applications can ensure that high bandwidth sites and users have limited impact on the network [64], [65]. The key measurements for QoS are bandwidth, latency and error rate. Therefore, QoS is of more importance to traffic that is sensitive to latency or jitter (variance in latency).

To use the QoS technology effectively, traffic needs to be classified based on the priorities dictated by the policies within the network. This differentiation ensures that the QoS mechanisms such as queuing can be effectively implemented [64], [65]. Traffic can be classified by port or IP, or using a more sophisticated approach such as by application or user. Based on this classification, queuing and bandwidth management tools are assigned rules to handle the traffic flows based on the classification they received up entering the network. The primary goal for priority queuing is to ensure there is minimal latency and adequate bandwidth for the most important applications within the network. To reach its objective, it allows for packets within a traffic flow to be stored and only process after the network is ready. In addition, each traffic flow is assigned a particular bandwidth based on the priority assigned to it upon entering the network [65], [66].

The network users play a major role in deciding the bandwidth management technique. For example, in the SSL in Dwesa, the network is primarily set up at schools; therefore, traffic from educational websites should be of higher priority while streaming from social media sites should be of lower priority or even blocked. When deciding on the classification policies, these factors have to be put into account in order to effectively manage bandwidth within the network. Network proxies can be used to block unwanted websites to avoid bandwidth mis-usage and for controlling the number of people connecting to the network at a given time.

It is clear that despite HTTPS rendering traditional proxy caching ineffective, the network proxies can still be utilized in conjunction with other bandwidth management techniques to provide a hybrid solution. As mentioned previously, much of the policies that are used to filter traffic are based on the users, therefore, it is worth noting that community members within the network need to agree on policies that can easily translate to rules that can be mapped onto the network firewalls, proxies or any technology that will be used to manage bandwidth.

2.7 Conclusion

The purpose of this review was to view the extent to which recent technological trends have affected community networks. It is clear that there have been significant changes in the physical layer and the web that has caused researchers to rethink their former strategies. On the physical layer, technological innovations such as long-distance Wi-Fi have reduced costs significantly and this is compounded by the regulatory decisions to allocate the microwave spectrum for unlicensed use. This has resulted in the point-to-multi-point wireless model being widely adopted to provide service to sparsely populated areas. On the web, there has been a shift from HTTP to HTTPS and the introduction of web2.0 has seen the rise of dynamic websites. Consequently, traditional proxy caches have become less popular and web pages generally use a lot more bandwidth. These are the most significant changes stated in the literature and because

Despite the existing literature coming from different backgrounds, they all aim to solve one fundamental issue, the digital divide. As a result, the majority of the papers share common constraints, which are the accessibility, availability and affordability of network infrastructure. It is also clear that the solutions varied significantly even though the researchers were addressing a common issue. The reason for this is the fact that rural and

remote communities have heterogeneous environments. The two main contrasting ideas in the literature are the use of a top-down approach and the use of the bottom-up approach to implement community networks in remote communities. Both of these approaches yielded positive results in their respective communities but in most cases, they often caused more issues. For example, in cases where the researchers took an entire top-down approach, they often overlooked important context-based factors that contributed to the failure of the project. On the other hand, when researchers overemphasized a bottom-up approach, they downplayed the lack of cohesion.

The importance of this thesis lies in the fact that it aims to leverage advantages from both approaches with a hybrid solution the living lab methodology. This is because it is apparent that the existing literature does not put enough emphasis on it. In addition, the thesis addresses the impact the HTTPS and HTTP/2 protocol has had on proxy caches in community networks. For future research, software defined networks appear to be a promising solution in rural communities. This will require research into innovative approaches that will drive down the costs of setting up the network, therefore, lowering technical and business barriers to entry.

3 Reviewing the topology and technology of the BI model

This chapter is a review of the Siyakhula Living Labs (SLL) network solution excluding services such as the web cache. The review is divided into three components; the network architecture, the network implementation and the methodology. The purpose of this review was to determine whether the BI solution was still viable in given the changes in ICT. We will review each of these components and suggest changes where they are necessary. The review sets the scene and lays the ground work for the following chapters on the network service layer.

3.1 Introduction

In the first chapter, we have established that changes within the ICT environment have prompted us to do a review of the BI model to establish whether it was still viable. The SLL network solution was implemented using the living lab methodology, within the context of a long-term experiment in connecting the unconnected. The methodology is iterative in nature, and that means with each iteration, we are required to reflect on previous iterations and use the lessons learnt to improve or adapt the (network) solution in the next phase of implementation. The SLL complete solution is comprised of the 3 layers: the network layer, the computer infrastructure, and the application layer, as said, in this thesis, we focus on the network layer review the following aspects: the network topology, the network technology and lastly the methodology. For each aspect of the network layer, we will do a detailed technical review that is experiment driven and then propose necessary changes.

3.2 BI topology review

A network topology describes how the nodes, devices and connections within the network are physically and logically arranged. The SLL topology is based on the star network topology [67]. The diagram below is a typical star network topology, which is characterized by a central hub and nodes connected to it and not directly to each other [68] [3]. The network topology in Dwesa follows the exact blueprint but instead of having only one central hub, it

has two, making it a double star network topology [4]. Unlike urban communities, rural communities have populations that are geographically sparse, hence why we have two central nodes – each of them acts as an extender of the other to ensure that the network spans the entire community [2], [19]. Usually, in a STAR topology, the central hub is represented by a hub or a switch, in this case however, we have computing infrastructure points of presence which we refer to as digital access nodes (DANs) [19], [68] [69]. These DANs are housed in schools for the following reasons, schools typically have the necessary infrastructure required for housing computer labs, such as electricity and appropriate venues; they are by definition education centres, therefore, they are in a position to educate both learners and community members [3]. Lastly, they are generally community focal points in rural areas and they enable us to fully leverage the social capital within communities which will result in greater buy-in for the network solution.

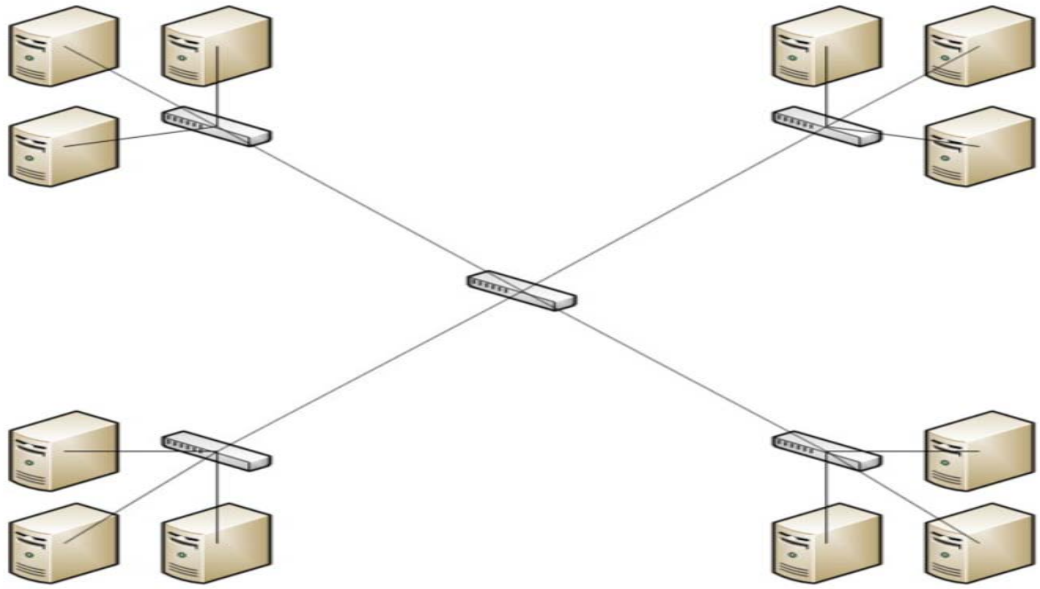

Fig 3: STAR Network Topology Diagram [68]

The star topology was historically designed for local area networks, which are typically implemented using 10Base-T or Ethernet over twisted pair technology with a central switch or hub that the nodes use to communicate to each other [68]. However, because wireless technology work better where there are predominantly low-income households and the population is geographically dispersed, we used them instead. The current blueprint replaces

the wired cables with wireless point-to-point connections between the nodes (schools) and the central hub (Digital Access Nodes). The network topology consists of two cells, which each use the STAR topology and these two cells are linked together via point-to-point connections between their DANs [1], [4]. Each DANs represents an access point of a Local Wireless Area Network (LWAN), these LWAN are what we call BI, in our current solution we have two BI, that are connected together via a point-to-point connection between their DANs. In essence, the BI model, can be described as a network that consists of LWANs connected wireless to each other via point-to-point connections.

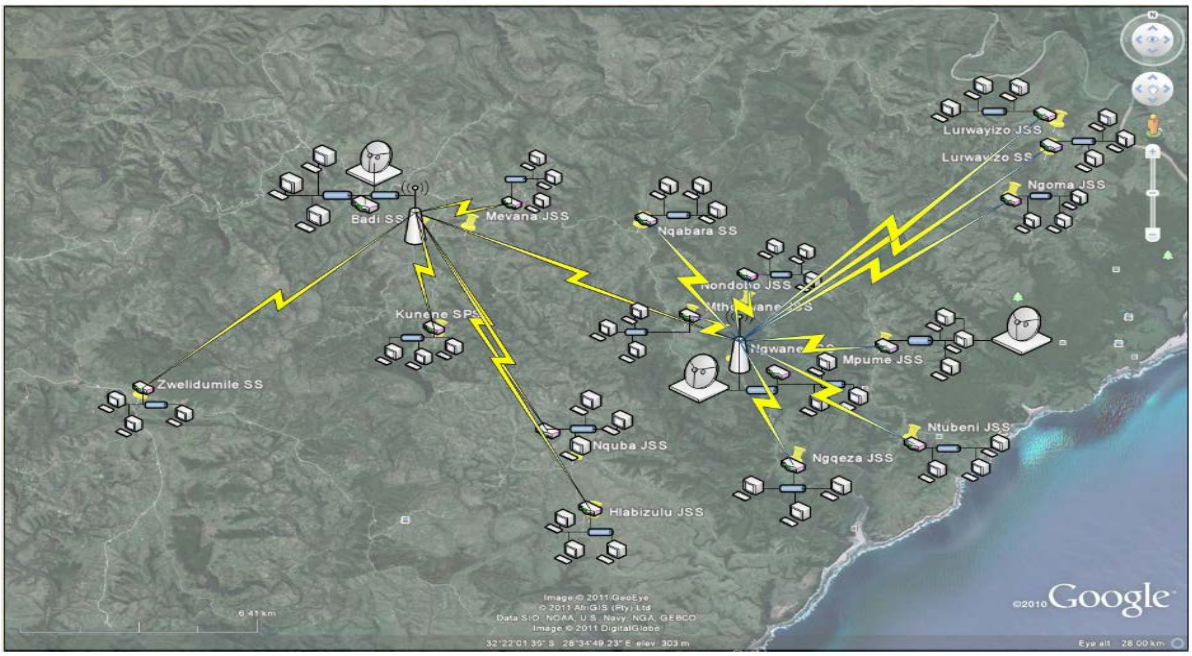

Fig 4: Geographic Layout of the BI in the SLL [19].

3.2.1 Digital Access Nodes

At the core of our topology is the digital access nodes, because it connects the LWANs and also doubles as an access point for wireless enabled devices to connect to the network [1], [4]. Therefore, it is only fitting that we discuss the underlying technologies and topology implemented on the access nodes. Earlier, we discussed the thin client architecture in the DANs, now we will discuss the network topology. In the first blueprint, a digital access node consisted of the following; a WiMAX micro base station, an omni-directional antenna connected to a core router, a layer 2 switch, a VSAT, thin clients and a wireless access point [4]. The core router receives traffic from different schools within the network and then it either routes it to the thin clients or to the Internet [1], [4].

Below, is a diagram illustrating what the typical DANs looked like before WiMAX became obsolete. The topology is still robust and applicable in current times, the only major change we will make is the technology used to setup the access node. In this case, as mentioned already, we will replace WiMAX devices with Wi-Fi devices.

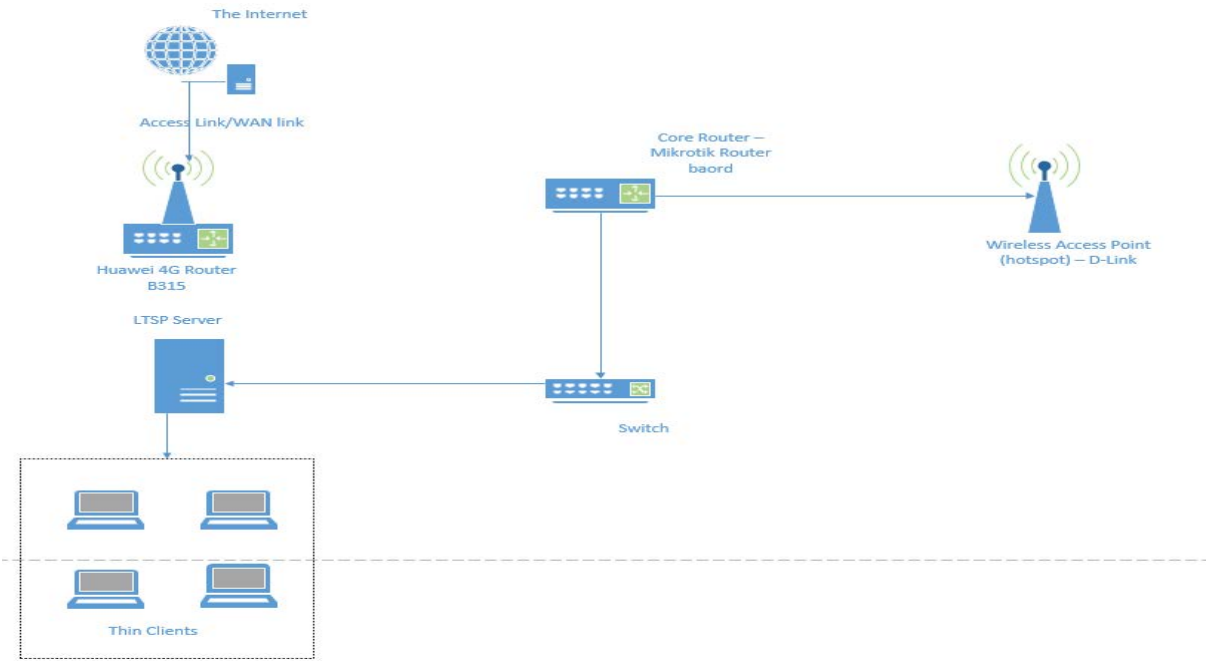

Fig 5: Network Topology of a digital access node

In the following section, we will now compare the STAR topology to the MESH topology. Our comparison will mainly be based on case studies. This will ensure that we are consistent with our emphasis on only analysing ICT projects in relation to their context. The comparisons of the two topologies from a general perspective does not make for an interesting discussion because we would repeat well known differences between the two topologies at a conceptual level. Using the case studies will allow us to factor in the nuances of these different projects (SLL and Zenzeleni). Lastly, we will discuss Khula Tech Solutions, which is a SME in Grahamstown/Makhanda, that uses the same topology as the one used in the SLL network solution. The idea behind including this case study is to illustrate how our blueprint (designed more than 10 years ago) is still stable being adopted in recent times and can also be applied in different use cases.

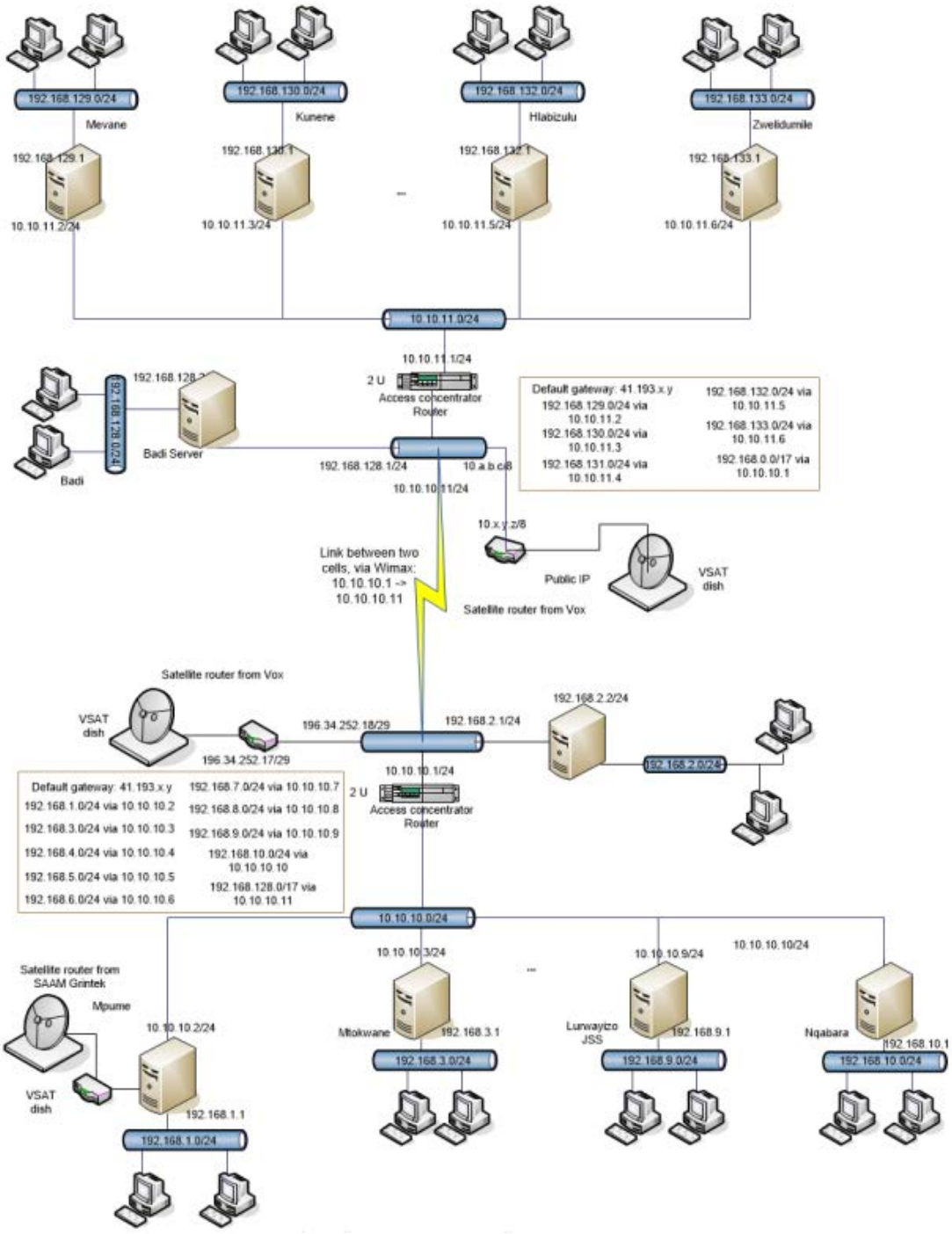

Fig 6: Network Diagram of SLL [1]

3.2.2 Comparing the MESH and the STAR topology as a last-mile internet solution

The general differences between MESH and STAR topology are well documented in the existing literature, therefore, it would be redundant for us to restate well known facts about the two. Instead, we will focus our attention on case studies in South Africa where each of these topologies were implemented in the last mile or in rural communities. This way, we ground our comparison in the context so that we factor in certain caveats that might be overlooked when simply doing a comparison on a conceptual level. The first case study is a partnership between VillageTelco, University of the Western Cape and Mankosi community in South Africa – Zenzeleni Network, the second case study is about a Small to Medium-sized Enterprise (SME) in Grahamstown called Khula Tech, the Grahamstown local community and a local school called Ntsika. The main emphasis in these case studies will be on the technologies and the topologies used to fully realize the solutions. Lastly, we will discuss Software Defined Networks (SDN), which is an approach to network management that vastly differs from the two traditional approaches we are currently comparing.

3.2.2.1 Case Study: - Zenzeleni Network by VillageTelco, Mankosi – South Africa

The University of Western Cape and the Mankosi community partnered in 2012 to create a model for a sustainable implementation of a bottom-up mesh network solutions in South African rural communities [24]. Mankosi is an impoverished community located in the Eastern Cape, where the majority of the population has no incomes other than remittances from family members or pensions and child allowances [24], [25], [70]. In addition, the community is divided into 12 villages that are geographically spread across 30km, as shown by the diagram below:

3.2.2.2 Network Topology and Technology

The telecommunications infrastructure in Mankosi village uses village telco, which is a mesh infrastructure that aims to provide easy to use and scalable VoIP over meshes WLAN (Wireless Local Area Network) telephone infrastructure [24], [25]. The major difference between mesh and star topology is that with the mesh topology there is no access point or central hub that routes traffic between the nodes[23]–[25]. Instead, every node within the network can communicate directly to each other via 2.4Ghz Wi-Fi frequency [23]–[25]. The main benefit of using this topology is that if one of the nodes malfunctions, it has no impact

on the rest of the network, in other words, it has no single point of failure [23]–[25]. Each node in the network is a mesh potato (MP) that operates in both infrastructure and ad hoc mode, the former allows the node to behave either as a client to receive internet access or as an access point to allow Wi-Fi enabled devices to connect to it. The latter, enables it to interact with other mesh potatoes. Each MP has an analogue telephone installed with it, which can be leverage for both voice calls and internet access [23]–[25]. The telephones are also linked to the nano station that links a mesh island to another. In essence, nano stations are repeaters, which are used to bridge two mesh clouds to form a larger network. A mesh cloud is a similar concept to the BI mentioned earlier, it is simply a subset of a larger network that connects to the rest of the network via a nano station, just like how the BI or LWANs connect to other BI via a point-to-point link between the DANs. Mesh potatoes have a few benefits, they are available locally, they have low power consumption and are designed for easy installation and maintenance. The diagram below is a more detailed view of the mesh potatoes network topology [23]–[25].

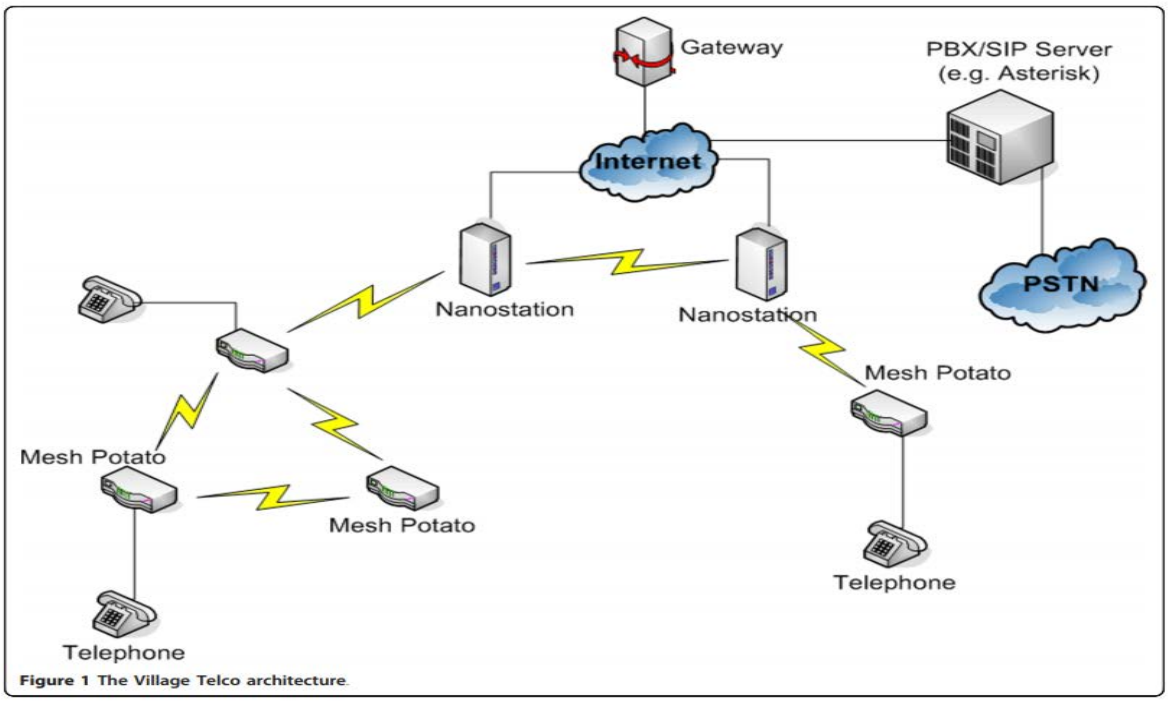

Fig 7: Mesh Network Topology [24]

Now that we have described both topologies in the form of case studies, we will now discuss the similarities and the differences. The network solution in Dwesa consists of BI, which are Local Wireless Area Networks using a star topology and are joined together via a point-to-

point connection between their DANs. On the other hand, the network solution in Mankosi, is mesh clouds which are in essence, Wireless Mesh Networks that are linked together via nano stations that act as repeaters to extend the range of the network. At a high level of abstraction, these networks are built on the same foundations of micro-networks i.e., BI or mesh clouds, linked together via a point-to-point connection (nano stations or digital access nodes) together to form a larger network. However, the implementation is what varies significantly. On one hand, the BI model requires more thorough network planning with carefully chosen antenna configurations and IP addressing structures to engineer high-quality links with good throughput [18]. The second approach makes use of a sporadic type of network in which community members join the network using easy-to-use, scalable hardware communicating using an agreed mesh routing protocol.

The BI model has two main advantages over the mesh clouds; The first one is that it uses standard industry infrastructure that has great market support, consequently, we can leverage pre-existing infrastructure to increase their coverage. An example of this is our recent switch to using Vodacom as the model's ISP, this means that we simply need to point our antennas to the nearest Vodacom base station and the whole community will have access to LTE/4G at reasonable speeds. The later sections will include experiments illustrating how effective this solution is. The second advantage the BI have is not technical, but is related to the business context of the network solutions. This will be discussed towards the end of the chapter, as it relates more to the application layer of the SLL solution than it does to the network/topology layer.

3.2.2.3 Case Study: - Khula Tech Solutions, Eastern Cape, Makhanda

Khula Tech solutions is a Wireless Internet Service Provider (WISP) founded in 2014 with the aim to provide customers with affordable and reliable internet access, telecommunications and Cloud hosting solutions. Unlike the typical ISP, WISP use fixed wireless connections to deliver Internet in the last mile. The benefits of this approach include the fact that is cheaper to deploy the infrastructure as compared to the traditional wired networks. Khula Tech solutions uses the same network topology as the BI with the key difference being the environment. The importance of this case study is to illustrate the reusability of the BI in different environments, in this case, semi-urban environments like Makhanda/Grahamstown.

In the introduction, we stated that the SLL network solution consists of three layers; the network layer, the computer infrastructure layer and the application layer. In this next section, we shift the discussion from the network layer to the application layer. This is the layer where the business model of the network is implemented. As will be discussed further, a sustainable business model will ensure that the network solutions have longevity. In addition, we will compare the two business models from Zenzeleni and SLL and then conclude.

3.2.2.4 Business Model

Long term sustainability is a key factor to a project's success in ICT4D; therefore, it is important to put in place measures that ensure that the solution is sustainable. The overarching issue, in our context, is the digital divide, and it is clear that in the context of South Africa, it is partly due to socio-economic factors. This means that the sustainability of the Zenzeleni and SLL network solutions is largely depended on how financially viable they are. By financial viability we mean, the initial costs to invest and the costs of maintenance. Wireless technology fare better in these environments i.e., low-income, geographically dispersed and remote environments, because they have a low initial cost of investment, they are easy to deploy and due to economies of scale, the devices or technologies needed are fairly cheap in relation to others. When we reduce these initial costs of investments, the projects become more attractive and sustainable for small micro enterprises (SMEs), therefore, it makes it easier for them to fund the projects. The other component of financial viability is costs of maintenance. This brings us to another reason why wireless technologies are the most viable option in this context. Wireless technologies are very easy to maintain and do not need highly specialized skills to operate. Therefore, it means the skills required are easily transferable to project champions during community engagement. This means the network solution can be maintained internally at lower cost. However, we still need to generate some revenue to ensure that at least we break even from a cost and revenue analysis or we generate some profit in the best-case scenario. Both Mankosi (Zenzeleni) and Dwesa (BI) have proposed different business models and in this section, we will discuss the differences and the sustainability of the two.

3.2.2.5 TeleWeaver: the business model

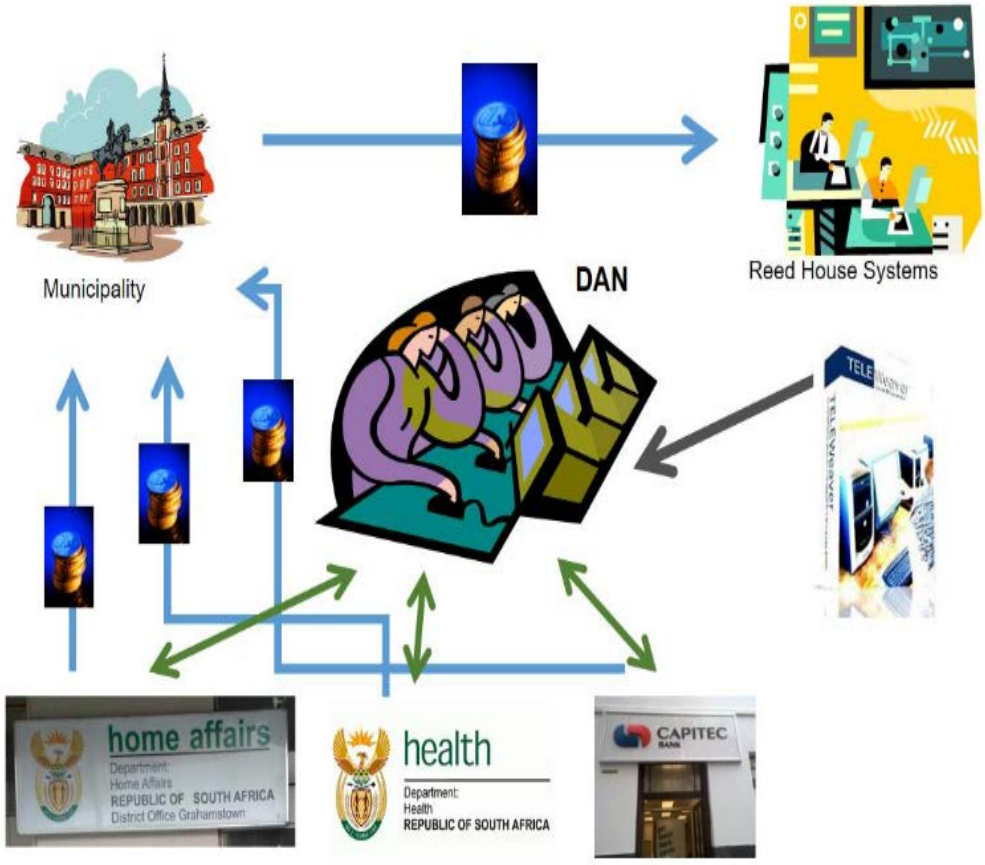

Fig 8: The teleweaver business model [3]

TeleWeaver is an application integration platform, served by the BI, with the purpose of providing useful services to both the communities and the government as a way to generate revenue to maintain the infrastructure [3], [4]. It was important to develop a model that was not dependent on the existence of valuable productions or tourists' attractions, as these are generally not available things in these communities [3], [4], [19]. As a result, a more general revenue model was implemented, that could guarantee that the ICT infrastructure was supported.

We established a now well adapted Ad-based revenue model, which entails creating ads for a specific service on high-traffic channels. For our context, we needed to find advertisers, which are external entities that are interested in trying to reach the members of this

community. These included, government departments and their units and subunits, which provide public service delivery to the citizens [1], [3], [4]. The effectiveness of this solution as a whole, lies in the fact that it leverages pre-existing relationships between community members for local buy-in of the project and relationships between the government and the citizens for monetization and the long-term sustainability of the project [3], [4].

In this context, the high-traffic channels are the channels the government uses for public service delivery and this includes information dissemination [3], [4]. If the channel is made accessible to deliver messages from the Government, the money spent over other channels can be diverted to this new channel. The model is sustainable because it is aligning with the larger objectives of bridging the digital divide, through leveraging ICT to provide effective channels of communication between citizens and the government. In addition, it propels the movement to eGovernment providing the groundwork needed to accelerate its arrival [3], [4].

3.2.2.6 Zenzeleni Business Model

The revenue generated in Zenzeleni will come from two main sources: the rate per minute to call within the network and for the charging of the mobile phones [23], [25]. The timetable for making phones calls is between 09:00 and 17:00 per day with a cost of R20 per hour of calls and the cost of charging a phone will be R3 per phone [23], [25]. According to a study done in the area, these two streams of income will produce enough to breakeven, which means operational expenditure and revenue will be the same.

It is worth noting that a key difference between the two business models is the fact that the model in Dwesa generates income from external entities through a well-adapted ad-revenue system and the model in Zenzeleni depends on income expenditure for it to be sustainable [23]. The model used in Dwesa is a more viable option because it leverages pre-existing Government communication channels and monetizes them. Public service delivery is inherently part of government functions, whether it's through campaigns or raising awareness to community members during a pandemic. Therefore, it is a more sustainable long-term solution.

Now that we have set the foundation and have established that the current blueprint is ideal/requires no changes. We now move on to the next section, which focuses on the

implementation of the network. The choices we make on the physical layer or the technology level should complement our blueprint design.

3.2.3 Software Defined Networks

SDN has received a lot of attention in recent years as a means of addressing long-standing issues in networking [71]. The two main principles for software-defined networks are separating the software that controls the network from the devices that implement it and abstracting the network hardware to packet processing functions instead of a fixed set of narrow features [38], [71]. For example, the hardware can be configured to implement a variety of network algorithms, such as shortest-path routing or traffic engineering. In general, it is possible to implement any of these applications in a traditional network but the process would be tedious; the programmer would have to design new distributed protocols and also handle practical issues [71]. The reason is that third-party programs cannot easily control traditional routers/switches. Decoupling the software from the hardware makes scaling the network relatively easy because the network can evolve without having to change the underlying hardware. By providing global visibility of the network state, SDNs make it easier to express network algorithms, which makes the network adaptable in real-time and to different environments [38], [71].

OpenFlow is a standard specification, which defines the features switches must provide, as well as an application interface the network controllers can use to communicate to the switches. OpenFlow specifies that a switch maintain a forwarding tables that contains priority rules, a pattern to distinguish the different sets of packets and actions that describe the handling of packets [71], [72].

Rural area networks are highly variable, diverse environments that are difficult to manage. However, SDNs offer a principled approach to managing rural wireless networks, providing simpler and more efficient business models [38], [71], [72]. By decoupling the control and the data plane, rural network operators can separate the two core operational tasks, which are construction network infrastructure and the configuration of that infrastructure [71]. This is particularly useful for the BI model in Dwesa, because both NMU and Rhodes University researchers are capable of troubleshooting any network configuration issues remotely while local technicians can handle the less sophisticated hardware problems.

In conclusion, it is clear that decoupling the tasks of infrastructure construction and network configuration will simplify the operation and sustainability of wireless networks in rural areas [38], [71]. SDNs have the potential of addressing long-standing problems concerning community networks in rural areas by allowing network experts to create a homogenous set of policies and tools to manage rural wireless networks despite the diverse nature of their environments. However, in the context of the BI model, this solution is met with two main barriers. Currently, the BI model does not have enough support from the government to subsidise the procurement of expensive equipment needed. In addition, to operate SDNs efficiently an understanding of the full network stack is important; therefore, the project needs more funding to either train local technicians to become specialists, or hire specialists, and both of these options are currently not affordable [38].

3.3 BI technology review

Several technologies could be used to implement the model, including optical fibre, terrestrial wireless and space based wireless systems such as VSAT. In theory, optical fibre is the ideal choice, it attains the highest speeds, it degrades less than wireless with distance and it provides very stable connections, because of being less susceptible to interference. However, in practice, one has to consider the context, which includes environmental factors like population density and financial viability[73]. Wireless technologies might fare better where there are predominantly low-income households and the population is geographically dispersed. In addition, due to the mass production of wireless devices and more and more sophisticated antennas (with MIMO capabilities, for example), the wireless industry has made dramatic advances in terms of throughput and reliability [34]. Therefore, the choice to use wireless technologies in real world implementations still makes sense. (That said, it is worth noting that fibre connectivity has gained some momentum in South African townships over the past ten years. In 2019, Gauteng MEC promised free internet services to townships in the province. In addition, Telkom and Vumatel have both engaged in a race to provide fibre to townships [74], [75]. The growing movement to connect townships with fibre has already given rise to competitive pricing. For example, Vumatel announced a project to provide Internet speeds of up to 100Mbps for less than R100 per month in Alexandra Township, as difficult as it might be to consider this more than commercial hype [76].)

3.3.1 LAN Portion of the BI

As for the specific wireless technology, WiMAX seemed very promising in 2005 for the LAN portion of the BI, but GSM was quickly becoming the dominant (mobile) wireless standard worldwide, with more than 3 billion global customers as of February 2010. Therefore, the next natural progression for the market to 4G was through LTE instead of through WiMAX, because it provided a much easier upgrade for their GSM/UMTS/HSPA-based networks and subscribers [77], [78]. Outdoor, long-distance Wi-Fi is the obvious replacement for the fixed wireless niche for which WiMAX was created initially. As WiMAX, Wi-Fi has the important characteristics of being a technology easy to deploy physically with only basic skills and without requiring investments beyond the capabilities of micro and small enterprise. As such, replacing WiMAX in the BI with Wi-Fi maintains the important characteristics of the initial WiMAX installation: to be a grass-roots technology. The economies of scale because of mass production is there for Wi-Fi and the technology have been tried and proven successful [78], [79].

3.3.2 WAN Part of the BI

As said above, the WAN part of the BI was implemented through VSATs. This was forced by the location of the Siyakhula Living Lab (no telephone network and very little mobile network available there when the project started) as well as the data rate limitations of the GSM mobile networks at the time. The situation has changed and reasonable fixed wireless uplinks to the Internet can be provided by mobile network operators at a competitive price. So, fixed wireless connections to mobile networks operators could be used instead of VSATs

3.4 Methodology

In the previous sections, we discussed the topology and the technology needed to realize the BI solution. Now, we will pay attention to the methodology used to implement the solution in the most effective manner. Network solutions are context sensitive; each environment has different needs and limitations that need to be factored in before implementing a solution. The original implementation suggested by the government was a top-down implementation plan codenamed ‘SA Connect’. The plan failed to produce results in areas where it was expected to make the most impact, the rural areas. In this section, we will discuss the top-

down approach proposed by the government through the 'SA Connect' plan and then propose the living lab methodology.

3.4.1 Bottom-Up Approach

The bottom-up method is a developmental approach that promotes distributed architectures. The advantages of this approach include the removal of a central point of failure which subsequently then remove bottlenecks. A bottom-up approach is resource driven and detail oriented and that makes it a favourable approach when dealing with problems that are context sensitive [17], [80], [81]. However, it has its own disadvantages; the project could lack cohesion because of the decision-making made at multiple levels within the project. In addition, the decision-making process is made more complicated because it is difficult to reach a consensus when dealing with diverse groups of people [80].

Guifi.net is an example of a project that used the bottom-up approach. The guifi.net is a bottom up, citizenship-driven project with the aim of creating free, open and neutral telecommunications network [82]. The project is a result of collaboration between four groups of participants: i) volunteers in the community ii) experts in telecommunications, demand and service supply iii) customers in need of affordable means to connect to the internet iv) public administrators [11]. These participants are the stakeholders of the project; therefore, their participation in the decision-making is crucial, especially for the sustainability of the project. In addition, network participants have the right to use their preferred means of connectivity to the network, which means each participant has the freedom to purchase equipment they know they can afford [11].

While this approach is ideal in the case of Guifi.net, the same does not hold true in certain communities, e.g., the Dwesa Community in South Africa. In Dwesa, the majority cannot afford to purchase the equipment. In addition, the Dwesa community does not have pre-existing infrastructure they can leverage when building their own community network. This means that there is a need for the government to intervene at some level and contribute to the initial investment [[49]7].

3.4.2 The Living Lab Approach

The living lab methodology emerged from the shortcomings of the traditional top-down approach and its inability to adapt to the different complexities of the varying projects [6].

Fundamentally, this approach is a flexible and adaptive, user driven approach that focuses on learning through experimentation. This methodology has the following components: i) beneficiary participation ii) flexible phased implementation iii) learning from experience iv) institutional support v) programme management.

In our context, the beneficiaries include Rhodes University, our industry partners, government and the Dwesa community members [5]. As mentioned above, the Living lab methodology emphasizes learning based on the outcome of failures, this entails phased implementation, with each new phase using the failures of the last phase to make better decisions. In fact, this research is laying the foundation for our next implementation phase. In conclusion, projects implemented using this methodology technically do not fail, they go through iterations of implementation.

3.4.3 Reviewing the SA Connect Plan

After announcing it in 2013, at the beginning of 2015 the Government launched a plan to connect multiple state entities at multimegabit speeds, to improve service delivery and foster economic growth. The plan was codenamed 'SA Connect'. The bulk of the state entities to be connected were schools.

The plan has not been a great success, however, in the areas on which it was expected to have the highest impact, poor rural and peri-urban areas. This was certainly true in the Eastern Cape, a large and poor province of South Africa. A number of reasons are behind it, but a fundamental one is this: the design and deployment of any ICT project are context-sensitive endeavours and by taking a traditional top-down approach, as used by the plan, the implementers missed important aspects that come with different environments [2], [17], [22]. A second reason, still linked to the top-down approach, was the lack of true engagement with the stakeholders, to harness their energy and creativity in terms of network deployment and operation. As a result of the very long time of implementation from the launch, the plan itself is now seen as obsolete and there have been calls to the Government to shelve the plan as is, because of being 'insufficient' [83].

The blueprint of the network component of the Siyakhula Living Lab, built through the living lab methodology, working directly with users in a real shared physical space, and now updated from a technological point of view - as reported in this paper- can be an important

contribution to at least one component of the Government plan, the most difficult one, the access part. The model behind the blueprint is of a grass-roots nature, emphasizing the collaboration between experts and users/beneficiaries, which makes it a more grounded, scalable and adaptable solution at national level. Because of the technology choices, the blueprint allows direct participation of micro and small businesses within the communities targeted by the network deployment. Based on the blueprint, each community can specialize the solution to their specific needs while achieving the strategic goals set by the Government.

The analogy made above constitutes part of the reasons why various studies insist on a bottom-up approach when working on an ICT4D project. Another example that illustrates this point is the 'SA Connect' plan that was initiated by the South African government, in an attempt to join state entities [3]. The decision to make Telkom, the former telecommunications incumbent in South Africa, the lead implementer hinted at a centralized point of control. Moreover, this suggested that the project would use the traditional top down approach [3]. Two years after the project had been announced it was clear that there was very little progress made in the rural communities where it was expected to make the most changes. Not only does the idea of the 'lead implementer' introduce bottleneck problems it also creates a single point of failure within the project [3], [16]. This illustrates communication infrastructure does not only differ from country to country but also from within different parts of the same country. The infrastructural needs vary within each tier of the economy; therefore, a uniform infrastructure is not possible because each tier has different needs and requirements [2], [3], [16].

3.5 Conclusion

The chapter reviewed the 'SA Connect' plan and blueprint for ICT networks deployment in poor communities, developed within the Siyakhula Living Lab over the years and suggested technological updates where necessary. The reviewed blueprint might, hopefully, become a component of a revised broadband plan by the Government. In the next chapter, we will focus on another integral component of the network model: the cache.

4 Caching in the Broadband Island

4.1 Introduction - Ngwane Lab Experiment

An experiment was done at one of the schools using a Huawei LTE CPE B315 modem (with Vodacom as its Internet Service Provider (ISP)). At the beginning of the experiment an LTSP server was setup to provide 25 thin clients for students and there was approximately 30 GB of data. In addition, repeated internet speed tests were done and the upload and download speed was an average of 10 Mbps and 30 Mbps respectively, with a latency of 60ms. To get the experiments started, the students got an instruction to do research on Nelson Mandela. Based on the nature of their research, students mainly visited his Wikipedia page, the nelsommandela.org website or any other related biography webpages. However, the majority of the students visited nelsonmandela.org and news outlets. The experiment was stopped after 30 minutes and each student was requested to log out of the network. The experiment revealed that the students consumed about 4GB of data in 30 minutes, which is not sustainable given that the monthly data cap is 30GB.

The strong presence of HTTPS, as we will discuss later in this chapter, coupled with the fast internet bandwidth in the experiment are clearly the reason for excessive data consumption. It is clear that the lack of an effective proxy cache is detrimental to the sustainability of the BI model. The solution we had to this problem was twofold and it is discussed across two chapters, this chapter and the next one. For this chapter, we address the impact of HTTPS on caching and how we can configure Squid to safely cache encrypted traffic.

4.2 The Importance of network caching in the BI model

A web cache or proxy server is a network entity that satisfies web requests on behalf of a web server. A proxy server was developed for two primary purposes, which are to reduce the load on the servers and to improve the user's experience when browsing the web. In general, the classification of user experience is as follows; page load time and data usage [84]. A proxy server stores frequently accessed content locally which means when a user requests for one of these resources, they are served directly from the cache and there is no need for them to be transferred over the WAN link to the internet. [84], [85].

In a community or institutional network, the probability that the individuals within the network will visit similar webpages is typically high. By definition, a community network is distinguished by lively interaction and engagement on issues of common concern. Hence, there is a high probability that the users within the community will need access similar resources such as discussion forums [86]. In rural areas and poor peri-urban, these community networks usually have to adhere to major financial constraints due to the fact that they cater to low-income households. As a result, financial viability is a key success factor when setting up a community network in these areas. On this account, the network cache In SLL (Dwesa) is an integral part of the BI model because it ensures that the BI model is financially viable. [3], [86]. In addition, the proxy server in Dwesa allows network administrators to enforce network quotas and monitor traffic from individuals within the network [85], [87]. Since users share the access link, this form of accountability is important especially when trying to regulate data usage per individual. With that said, it is clear that the rate of data consumption in SLL contributes significantly to the user experience. Although it is worth noting that this does not mean that page load time or latency is insignificant in the overall picture, but is rather an issue of a lower priority in a context where data usage is a bigger concern.

4.3 Caching when using HTTPS

In the previous section, we mentioned how web content caching technologies, particularly Squid, which is the caching proxy in us in SLL, have been affected by HTTPS. In this section, we describe the solution we implemented to work around this issue.

As expected, Squid was adapted by its developers to allow network administrators to reintroduce caching while still leveraging most of the benefits from HTTPS [66], [85]. The developers first attempt at adapting squid resulted in the Squid-in-the-middle SSL bump feature, which was eventually replaced by the peek-n-splice feature in later versions of Squid – version 3.5 onwards [85]. As stated earlier, the introduction of HTTPS was largely driven by the personalization of the web through Web2.0, hence, the greater demand for security [66], [88], [89]. HTTPS provides both the client and the server the means to authenticate each other. In essence, HTTPS assures the client that they are communicating or sending personal information to the right endpoint/server. Now, based on this premise, the Squid adaptation works as follows. Squid inspects the authentication messages between the client and the server during the TCP and TLS handshakes and then duplicates them to mimic the server to

the client and vice versa. Essentially, Squid becomes the man-in-the-middle for the authentication process between the client and the server, but, both the client and the server are unaware. This ability to mimic both the client and the server is the cornerstone of this feature because we still retain our ability to use end-to-end authentication while leveraging the benefits of an in-network proxy cache, at least if the local network and cache can be trusted, as we discuss next.

There are obvious limitations with this approach, with the main one being the fact that both the client and the server possibly trust the middle-man – Squid, to maintain the integrity of the content. The clients have to fully trust the middle box to communicate with the target server on behalf of them in a secure manner [66]. Establishing trust between the server and the client is a core feature of HTTPS, therefore, in this context, it is an issue that has to be addressed. A few solutions have been provided as a workaround for this issue, these include using secure hash checksums in the content or using encrypted video caching for secure video delivery [66], [88]. In SLL, squid is within a private trustworthy network, which means it can only be accessed by authenticated users. In addition, it is behind a NAT firewall, which means only trusted users have access to the networks content. To be clear, this does not eliminate the possibility of Squid being compromised, it simply reduces the attack vector by limiting it to individuals who have access to the private network.

We observed that there are two types of websites. The first type can be exemplified by Wikipedia, the second by a bank [66], [90]. The second type, usually require the user to provide sensitive information. The majority of the sites of the second type provide content that is dynamically generated based on the user and hence they are not usefully cacheable [66], [90]. Given the fact we have not entirely eliminated the possibility of Squid being compromised, the potential risk associated with inspecting sensitive traffic e.g., banks, outweighs the benefits. And reasonably, users are more concerned about security on banking websites than they are concerned about reducing bandwidth consumption. Therefore, it makes sense to only intercept traffic going to sites of the first type, which we will classify as ‘public’. This feature does not guarantee data confidentiality, naturally it isn’t the goal at all [90]. Lack of perfect confidentiality is an unavoidable consequence of caching because any proxy must be able to tell if two or more clients have requested for the same file, which violates the requirements for confidentiality [90]. This means that the network administrators together with the users within the network have to reach a consensus on which domains to consider private and which ones to consider public. Once a consensus is reached, Squid will

be configured to intercept public traffic and ignore private traffic. With this approach, the only traffic that is cached is the traffic that does not contain any sensitive data from the users. This it removes the need for users to trust Squid.

4.3.1 Terminology

Peek – This is the process in which squid parses the TLS Client Hello and the TLS Server Hello and extracts the SNI and the server certificate [85].

Bump – squid will establish a SSL connection with the server using the clients SNI and establish a SSL connection with the client using an auto-generated certificate to mimic the server [85].

Splice – a tunnel through squid will be established between the client and the server, without squid decoding the connection. In other words, the client and the server will exchange data directly [85].

Callout Sequence - The sequence of checks Squid does after successfully parsing the request header from the client. The sequence checks for cache directives, Type of Service (ToS) marking and bump directives, to name a few [85].

The SSL connection establishment rules are simple: if it the traffic is classified as public, bump it; if it is private, splice it [85]. When the connection satisfies the bump conditions, squid will handle the request on behalf of the client and will have the ability to decrypt and cache the content retrieved from the server [85]. If the connection satisfies the splice condition, a TCP connection between the client and the server is made without decoding the connection. In other words, the client and server exchange the data with no proxy in between. In the following section, we will provide details on how squid was configured to cache HTTPS traffic.

4.4 Configuring Squid to bump HTTPS traffic

The general configuration for Squid is straightforward and has been thoroughly documented on the Squid website [85]. For this section, my main emphasis is on the specific configuration options needed to enable Squid to intercept HTTPS, so configuring squid with SSL support, create a self-signing certificate and import that certificate on the client.

Installing squid with SLL support

The first step is installing Squid with support for HTTPS filtering and SSL inspection. Fortunately, there is a precompiled version of it on the diladele repository [91]. While compiling Squid manually would have been worthwhile, it is not possible because currently, there is no online repo yet for the version of Ubuntu we are using (version 20). In fact, the version on the diladele repository is actually designed for Ubuntu 18 [91]. The installation is straightforward and has been well articulated on the documentation provided by the repo. As with any repo, the first step will be to add a public a public key before you can add/clone the repository to the local machine or server.

Generate Dynamic Self-Signed Certificate

We will use OpenSSL to generate and sign certificates that will be used by Squid in TLS ServerHello handshake process. We do that with the following command:

```
openssl req -new -newkey rsa:4096 -sha256 -days 3650 -nodes -x509 -
keyout myca.pem -out myca.pem
```

Fig 9: OpenSSL command to generate a certificate for Squid

This command will generate both a key and a certificate in Privacy-Enhanced Mail (PEM) format. Once the certificate has been generated, we then import it into our browser and put it together with the group of other trusted certificates. To avoid a mismatch in the site domain name and the Squid certificate we created above, we use the DynamicSslCert feature, which generates site certificates that match the requested domain name.

For future certificates, we then initialize an SSL database file where we store all future certificates that will be generated by the server.

4.4.1 Configure Squid

After preparing the environment for Squid, the next step is to configure Squid to use the environmental resources we have provided. Earlier, we added the SSL bump directive and generated a dynamic self-signing certificate. The next step is to configure Squid to use these resources as demonstrated below.

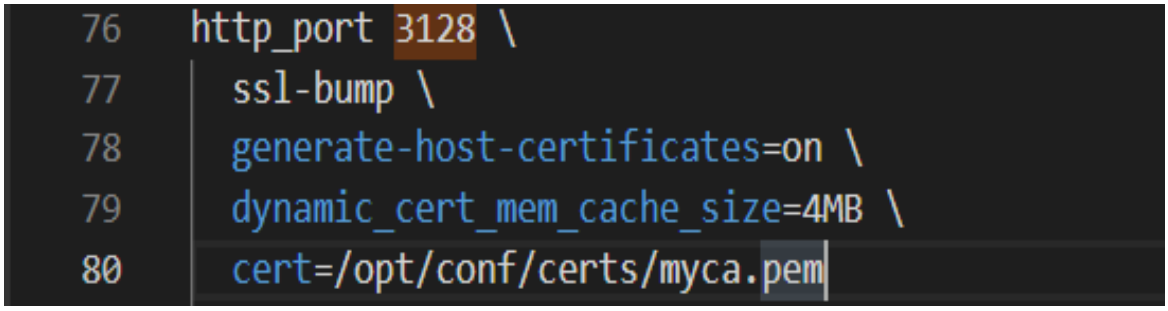

Fig 10: Port Configuration and SSL-Bump mode options

In the snippet above, we instruct Squid to use the SSL-bump directive for SSL inspection and HTTPS interception. Also, we have indicated in the configuration (line 78) that we will be using dynamic self-signed certificates by giving the "generate-host-certificates" directive the value "on". Since we will generate certificates as per request, we allocate 4MB of memory cache to Squid for generated certificates, which will be enough to store approximately 1000 certificates. Lastly, we provide a path to the root certificate that Squid will use.

The port number determines where Squid will listen for client requests, for this configuration, we used the default port number 3128 as show on line 76 in Fig 3. The network port identifies the application or service running on a given IP address and this is very useful when many services are running on the same device. This is best illustrated further down in this chapter when we configure Pi-Hole (an ad-blocking software) to work together with Squid.

4.4.2 SSL Bump Peek and Splice Processing Steps

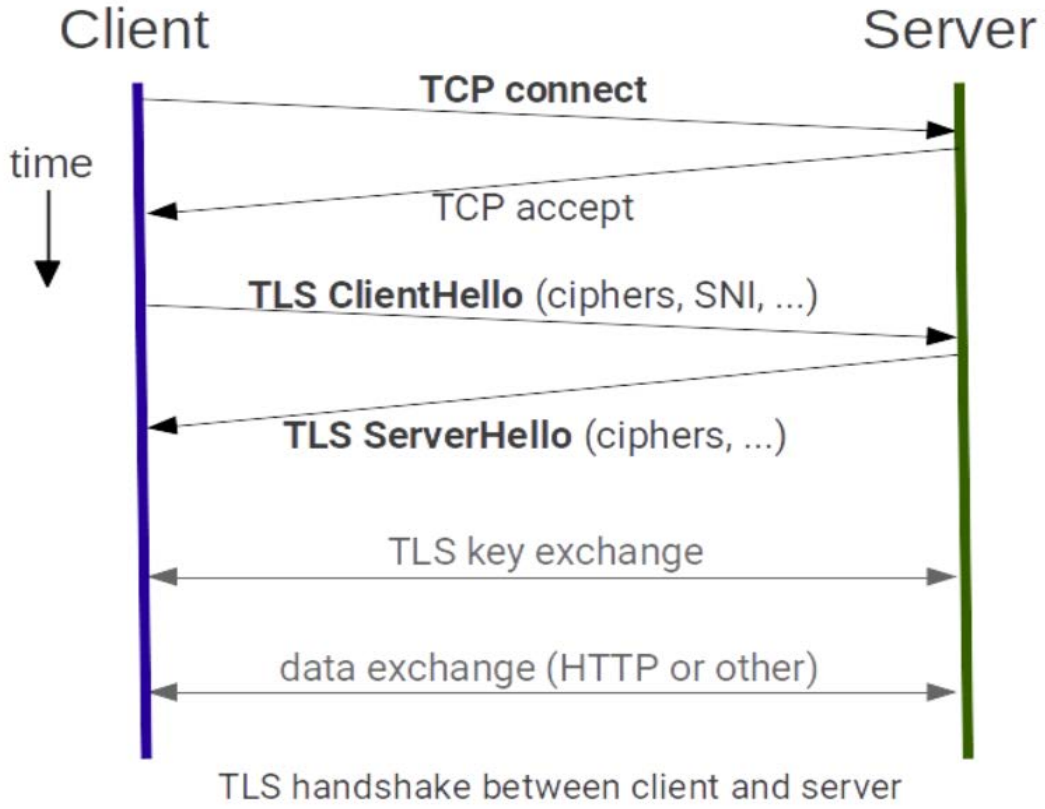

Fig 11: TLS handshake between client and server [85]

With the aid of the Fig 18, I will provide a simple overview of how the peek and splice feature works. The peek and splice feature looks at the TLS Client Hello message and SNI information, and sends an identical or similar Client Hello Message to the server[85]. When the server responds, Squid inspects the TLS Server Hello message and sends a duplicate message to the client. In essence, Squid inspects traffic coming from the client and then uses the information it has retrieved from the inspection to mimic the client to the server. Upon receiving the TLS Client Hello message, the server has to respond with a TLS Server Hello message as required TLS/SSL handshaking protocol. During this process, the flow between the server and the client is duplicated, with the first flow being between the client and the server and the second flow being between the proxy and the server[85]. There are three main steps that characterizes this process, they'll be discussed at length in the following sections.

4.4.2.1 Step 1

The TCP handshake process is initiated by the client with a TCP Client Hello Message which is then intercepted by Squid. Once the message has been requested, Squid collect client information from it and creates a fake CONNECT request [85]. The CONNECT request will go through the callout sequence, which is a sequence of checks and adjustments Squid applies to most HTTP requests before contacting the server [85]. The sequence has 12 item checklists, but for this specific configuration we will focus on the *ssl_bump* directive, which is the 11th item on the checklist. On the SSL bump directive, Squid will evaluate all SSL bump directives below and then perform the first matching action:

```
ssl_bump splice localhost
ssl_bump splice ssl_skip_bump
# peek on SslBump1 step
ssl_bump peek step1 all
# force bump (decryption) on SSL request header match
ssl_bump bump ssl_force_bump
# just tunnel (no decryption) based on whitelisting (domains, ips, src/dest, user-agent)
#ssl_bump splice "www.nelsonmandela.org"
ssl_bump splice ssl_exclude_domains
ssl_bump splice ssl_exclude_ips
ssl_bump splice ssl_exclude_useraddr
ssl_bump splice ssl_exclude_useragent
# To disable decryption (bump) uncomment line "ssl_bump splice all" and comment "sspl_bump bump all"
#ssl_bump splice all
ssl_bump bump all
```

Fig 12: Squid configuration file – SSL bump rules

This first step is always performed and the CONNECT request(s) are logged in the access.log file. Squid logs each connect request in the access log as shown in the image below.

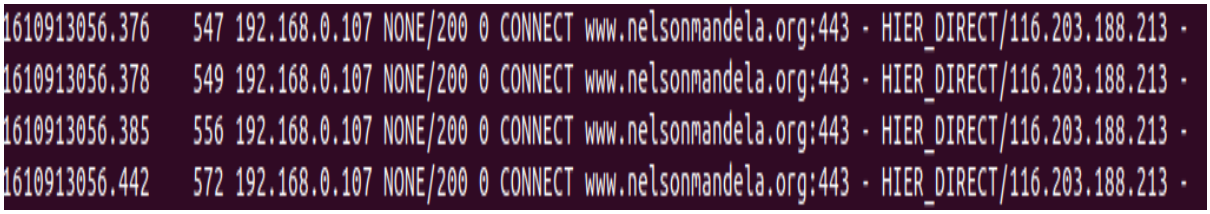

Fig 13: Fake CONNECT request made by Squid in the access log file

4.4.2.2 Step 2

After a TCP connection has been established, the TLS handshake is initiated. Squid intercepts the Client Hello message from the TLS handshake and creates a duplicate CONNECT request to the server using the retrieved information, the information includes the clients Server Name Indication (SNI) if it is available [85]. The SNI is a protocol that allows the client to indicate the hostname it is attempting to connect to. Next, Squid adjusts the previous CONNECT request from step 1, with updated information retrieved from the TLS Client Hello message and then it duplicates the CONNECT request. In accordance to Squid configuration, the CONNECT request goes through the callout sequence again and evaluates the directives in fig 4.

4.4.2.3 Step 3

After the server receives the TLS Client Hello, it replies with a TLS Server Hello message. This message will include information like the TLS server certificate and information about the server [85]. The proxy will use this information to generate a self-signed certificate with information mirroring the server. This certificate will be signed by the root CA certificate we generated earlier [85]. Again, Squid will evaluate the SSL bump rules and perform the first matching action.

4.4.2.4 Blacklisting and Whitelisting domains

As mentioned earlier, data confidentiality is the trade-off we are making when we implement this solution, and the justification is the fact that our cache will consist of mainly public information that does not require privacy, such as material from newspapers or Wikipedia. This means that we want to avoid intercepting or bumping traffic that contains sensitive information or that is blacklisted. So the next step is to compile a list of websites that will either be categorized as private or public. Private domains will be blacklisted, while public domains will be whitelisted. To enforce this policy, Squid provides the necessary tools in the

form of access control lists (ACLs). These lists filter users based on lP addresses, hostnames and request methods. Once the lists are created, they are then combined with rules or directives that enforce a particular action from Squid.

```
31  acl step1 at_step SslBump1
32  acl step2 at_step SslBump2
33  acl step3 at_step SslBump3
34  acl ssl_skip_bump req_header X-SSL-Bump -i skip
35  acl ssl_force_bump req_header X-SSL-Bump -i force
36  acl ssl_exclude_domains   ssl::server_name "/opt/conf/squid/ssl/exclude/domains.conf"
37  acl ssl_exclude_ips       dst              "/opt/conf/squid/ssl/exclude/ips.conf"
38  acl ssl_exclude_useraddr  src              "/opt/conf/squid/ssl/exclude/useraddr.conf"
39  acl ssl_exclude_useragent browser -i       "/opt/conf/squid/ssl/exclude/useragent.conf"
40  # cache acl
41  acl cache_exclude_domainaddr dst           "/opt/conf/squid/cache/exclude/domainaddr.conf"
42  acl cache_exclude_domainname dstdomain     "/opt/conf/squid/cache/exclude/domainname.conf"
43  acl cache_exclude_useraddr   src           "/opt/conf/squid/cache/exclude/useraddr.conf"
44  acl cache_exclude_useragent  browser -i    "/opt/conf/squid/cache/exclude/useragent.conf"
45  # send_hit acl
46  acl cache_exclude_contenttype rep_mime_type "/opt/conf/squid/cache/exclude/contenttype.conf"
```

Fig 14: ACL for SLL bump directive

To illustrate this, I will use a banking domain as the private website and an educational website as the public website. This example is meant to illustrate how the proxy will work when deployed in SLL. To determine whether the configuration is working, we will use the Site Identify button (a padlock) that appears in the address bar to the left of the web address. If we are on a private website, the site identity should match the domain but if the website is public, the site identity should match the one provided by our own server certificate. The Site Identity button includes details about the server certificate, the website owner and the encryption algorithm being used.

4.4.2.5 Blacklisted Domains

Banking websites are part of the websites that we would prefer not to intercept and bump. When you look at the image below, it is clear that Squid was not the middle-man because the site identity matches what the server is supposed to provide

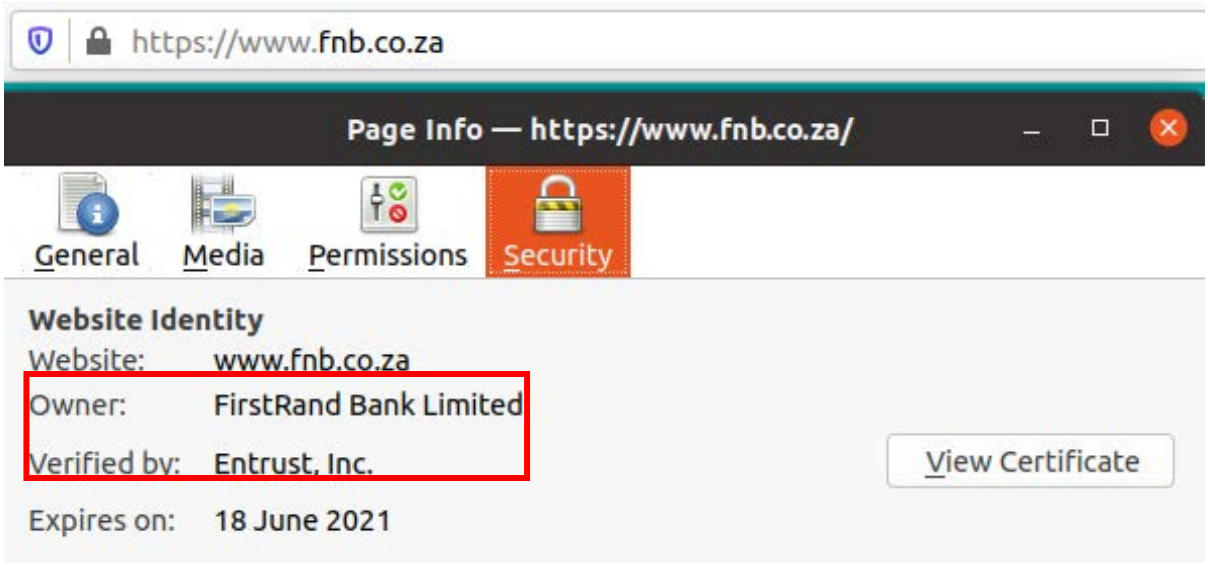

Fig 15: Security information provided by website after SSL Splice

4.4.2.6 Whitelisted Domains

The Rhodes University is a public website that contains information about the university that anyone can view with no need to provide any private information. As you can see below, we bumped the connection to the site and it is revealed by the site identity and the owner of the certificate.

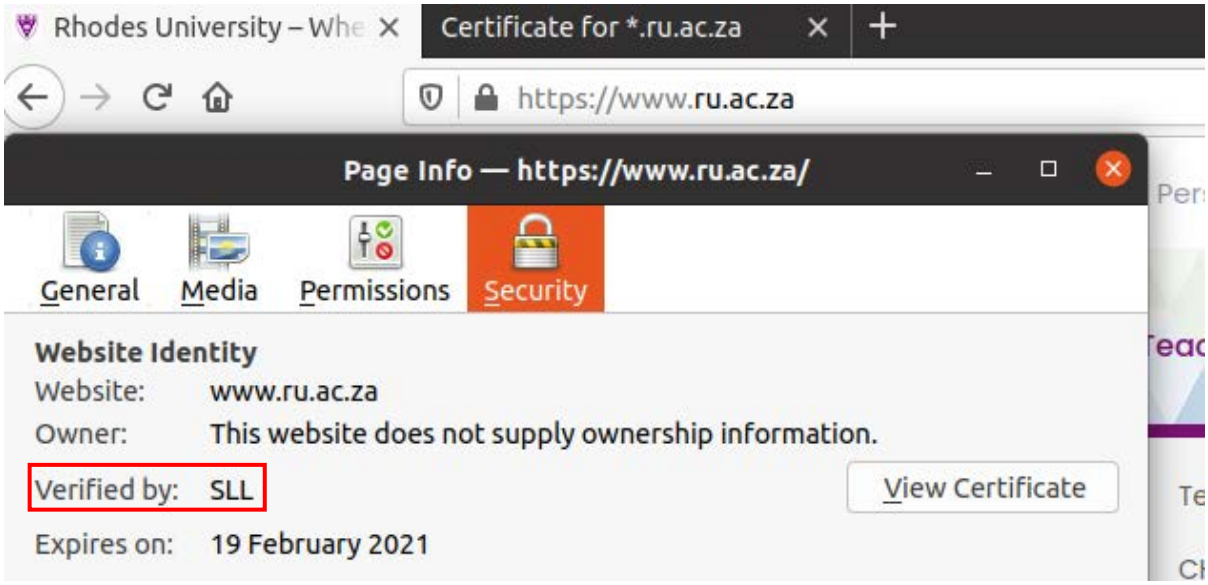

Fig 16: Security information provided by Bumped website

4.4.3 Squid Performance Evaluation

In the last section, the main focus was on understanding the underlying Squid architecture that enables it to observe, intercept and cache HTTPS traffic between the client and the server. In this section we shift our focus to its performance evaluation. To get an accurate picture of how Squid is performing, we need a performance metrics to measure it. In addition, we want to simulate the type of environment Squid will be used in so that our analysis is well-grounded in the context.

Cache Hit Rate = A cache hit is when the content requested by the client is successfully served from the cache

Cache Hit Rate = Total requests served from the cache * 100 / Total number of requests

Cache Miss Rate = A cache miss is an instance were content is search for in the cache and is not found

Cache Miss Rate = Number of cache misses * 100 / Total number of requests

4.4.3.1 Data Collection

The goal of this experiment was to measure how effective Squid was when configured to cache encrypted traffic. This experiment was performed in a controlled environment that simulates how it will work when deployed in SSL. The environment variable include, a Linux operating system – Ubuntu version 20 Long Term Support (LTS), Firefox web browser with browser cache switched off and a computer with a 64-bit architecture. The experiment was performed by a single user over a period of 30 minutes. The user will visit 3 sites, ru.ac.za, khanacademy.org and nelsonmandela.org. As the experiment was being performed, we automatically generated reports using the report generator SARG (Squid Analysis Report Generator) to obtain a comprehensive report on the metrics we mentioned earlier.

4.4.3.2 Evaluation of data

As stated, we used Squid Analysis Report Generator to generate a well-formatted summary of Squids access logs. This tool summarizes the access logs in a readable manner and provides a “cache-out” column for all the cache misses and a “in-cache” column for all content that was served from the cache.

ACCESSED SITE	CONNECT	BYTES	%BYTES	IN-CACHE-OUT		ELAPSED TIME	MILLISEC	%TIME
www.nelsonmandela.org	319	85.30M	36.81%	15.20%	84.80%	00:02:16	136,339	8.73%
www.ru.ac.za	496	45.87M	19.80%	41.06%	58.94%	00:01:24	84,593	5.41%
cms-tc.pbskids.org	348	12.76M	5.51%	65.11%	34.89%	00:01:11	71,257	4.56%
dl.acm.org	247	10.61M	4.58%	46.16%	53.84%	00:00:32	32,199	2.06%
r2---sn-woc7ln7r.googlevideo.com	56	7.21M	3.11%	0.00%	100.00%	00:00:21	21,694	1.39%
www.dailymail.co.uk	141	6.27M	2.71%	60.27%	39.73%	00:00:06	6,139	0.39%
www.coolmathgames.com	398	5.54M	2.39%	78.13%	21.87%	00:00:04	4,985	0.32%
www.youtube.com	141	5.54M	2.39%	26.13%	73.87%	00:00:21	21,171	1.35%
cdn.kastatic.org	281	5.22M	2.26%	0.00%	100.00%	00:00:03	3,798	0.24%
r2---sn-4g5e6nss.googlevideo.com	48	4.53M	1.96%	0.00%	100.00%	00:00:36	36,932	2.36%
i.dailymail.co.uk	269	3.44M	1.49%	61.22%	38.78%	00:00:04	4,425	0.28%
pbs.twimg.com	63	3.37M	1.46%	66.70%	33.30%	00:00:00	877	0.06%
i.ytimg.com	91	2.74M	1.19%	16.28%	83.72%	00:00:19	19,375	1.24%
edition.cnn.com	70	2.72M	1.18%	49.95%	50.05%	00:00:00	914	0.06%
translate.googleapis.com	144	2.53M	1.09%	86.45%	13.55%	00:00:05	5,416	0.35%
cdn.cnn.com	75	2.39M	1.03%	51.51%	48.49%	00:00:01	1,223	0.08%
imasdk.googleapis.com	15	2.09M	0.90%	27.67%	72.33%	00:00:06	6,327	0.40%
r1---sn-4g5ednsy.googlevideo.com	27	1.78M	0.77%	0.00%	100.00%	00:00:27	27,468	1.76%
use.typekit.net	61	1.76M	0.76%	78.95%	21.05%	00:00:01	1,132	0.07%
fonts.gstatic.com	100	1.50M	0.65%	70.69%	29.31%	00:00:01	1,029	0.07%
www.gstatic.com	95	1.34M	0.58%	66.43%	33.57%	00:00:01	1,646	0.11%
incoming.telemetry.mozilla.org	156	661.91K	0.29%	100.00%	0.00%	00:00:00	0	0.00%
a.fsdn.com	27	644.06K	0.28%	0.00%	100.00%	00:00:02	2,546	0.16%
edition.i.cdn.cnn.com	26	613.21K	0.26%	50.08%	49.92%	00:00:00	207	0.01%
creative.dailymail.co.uk	5	569.96K	0.25%	60.00%	40.00%	00:00:00	691	0.04%
scripts.dailymail.co.uk	25	506.93K	0.22%	60.01%	39.99%	00:00:01	1,884	0.12%
apis.google.com	14	466.73K	0.20%	83.55%	16.45%	00:00:00	320	0.02%
easylist-downloads.adblockplus.org	2	440.77K	0.19%	0.00%	100.00%	00:00:01	1,273	0.08%
js.stripe.com	18	335.90K	0.14%	83.33%	16.67%	00:00:01	1,200	0.08%
content-signature-2.cdn.mozilla.net	55	329.91K	0.14%	94.54%	5.46%	00:00:01	1,406	0.09%

Fig 17: SARG reports showing content fetched from the internet vs content served from the cache

The data for single user (localhost) was collected and summarized in Fig 17. There are very few sites the user visited and the majority of the sites logged in the report are background scripts and ads fetched in the background. For the accuracy of the results, the browser's cache

was disabled to prevent local caching. Each of the sites above were accessed 3 times by the user as a way to warm up the cache before using it [92].

The cache hit rate for ru.ac.za, khanacademy.org and nelsonmandela.org was 40%, 0% and 15% respectively. The average cache hit ratio in Fig 17 is approximately 45% on all 32 web requests. This means that from all the content requested by the user, less than half of it was served from the cache. An efficient cache maximises the number of cache hits and minimises the misses, which results in better resource utilization [93]–[95]. A cache hit rate between 80% and 90% more accurately represents an efficient cache [93]–[95]. If this is true, then our cache was inefficient. However, there are a number of caveats which concern the manner in which we assess these results. As we mentioned in the introduction, the nature of modern web traffic has evolved greatly and it has to be factored in to give an accurate account of Squid's performance.

Improving our understanding of the underlying nature of modern web traffic was necessary for us to interpret these results more accurately. Once we properly understood the nature of the web traffic, we could configure Squid accordingly and improve our performance evaluation metrics [52], [96]. A website today fetches content not only from the origin server, but from servers used by web content providers, advertising agencies and Content Distribution Networks (CDNs) [52], [96]. Furthermore, websites have moved from simply hosting text and images to varying content types, rich media and videos [52], [96]. This complexity entails more difficulty configuring proxy caches because each server has different caching rules for its content. Therefore, to accurately evaluate the effectiveness of our configuration, our cache needs to be configured in a very generic manner that will enable it to override the caching requests from each server. In addition, the metrics we used above had to be based on the number of cacheable objects returned from the server.

$$\text{Cache Hit Rate} = \frac{\text{Total requests served from the cache}}{\text{of requests}} * 100 \quad * \text{Number of cacheable content / Total number}$$

$$\text{Cache Miss Rate} = \frac{\text{Number of cache misses}}{} * 100 \quad * \text{Number of cacheable content / Total number of requests}$$

4.4.4 Understanding web complexity

There are two main metrics we used to determine the characteristics of a webpage, page level characteristics and network level characteristics. Page level characteristics include the

percentage of dynamic content a webpage has as compared to static content and the type of content the client is receiving from the server. Further, it includes the percentage of cacheable content a website fetches on page load. On the other hand, network level characteristics determine the number of non-origin servers it retrieves its content from. This includes CDNs. In addition, network level characteristics involve examining HTTP cache headers to determine the cache directives provided by the web server. In this chapter, the main focus is on network level characteristics to determine the percentage of cacheable content the website is fetching each time a user visits it. The WebrequestApi was designed specifically for that operation and we used it to monitor web traffic between the client and the server.

There are various states of the web request/response cycle. For our investigation, we pay more attention to the state after we have received response headers from the server. The goal is to inspect the response object from the server as a way to determine whether the cache-header directives align with our configuration. In addition, this script will have a counter that will count the number of cacheable and non-cacheable objects received from the server, based on the server directives.

```
var counter = {
    cacheable: 0,
    notCacheable: 0
}
chrome.webRequest.onHeadersReceived.addListener(
    function add(details){
        var responseHeaders = details.responseHeaders;
        for(var i = 0; i < responseHeaders.length; i++){

            if(responseHeaders[i].name.includes("cache-control") && responseHeaders[i].value.includes("no-cache") ){
                console.log(responseHeaders[i]);
                counter.notCacheable++;
            }
            else if(responseHeaders[i].name.includes("cache-control") && responseHeaders[i].value.includes("private")){
                console.log(responseHeaders[i]);
                counter.notCacheable++;
            }

           else  if (responseHeaders[i].name.includes("cache-control") && !responseHeaders[i].value.includes("no-cache"))
            {
                console.log(responseHeaders[i]);
                counter.cacheable++;
            }

        }
    }, {types: ["sub_frame", "script", "image", "xmlhttprequest", "other"],
    urls: ["<all_urls>"]},
    ["blocking","responseHeaders"]
);
```

Fig 18: WebrequestApi used to inspect HTTP response headers from the server

Fig 19 was a typical response object from a server. In this case, it was quite obvious that it was an ad based on the “g.doubleclick.net” domain contained in the URL. The main thing to pay attention to here is the fact that the object has cache-control directives instructing the browser and the proxy server not to cache it. Cache-Control is an HTTP header used to specific caching policies from both the client requests and the server responses

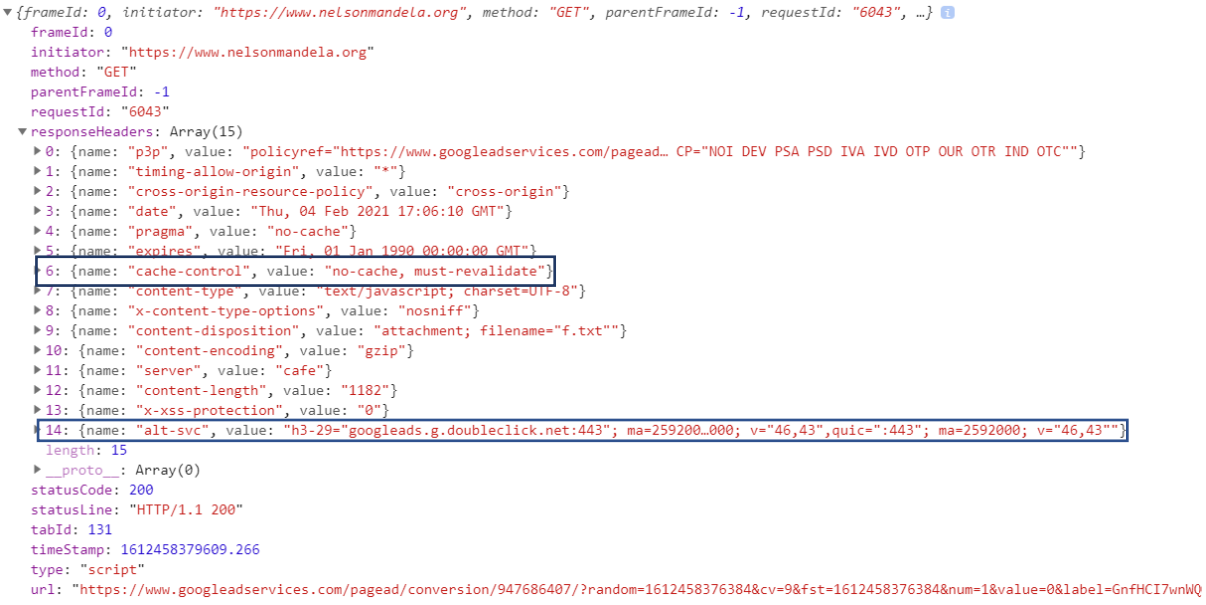

Fig 19: A response object from the server intercepted by the background script

Fig 19 was an example of a response object received from the nelsonmandela.org domain. The two main things to take note of, are the cache-control HTTP response header and the URL. The cache control has a no-cache directive for Squid, which means we should not cache the content associated with this URL. Secondly, the URL is from google ad services, which simply demonstrates the fact that when the user loads a webpage, the web site fetches its resources from multiple domains, including from ad services. In this case, the advert is not cacheable for clear reasons, each advert is usually unique to the user, and therefore it counts as dynamically generated content. Using the script the script in Fig 18, we then capture every response from the server and put them into two categories, cacheable and non-cacheable. The results from running the script in Fig 18:

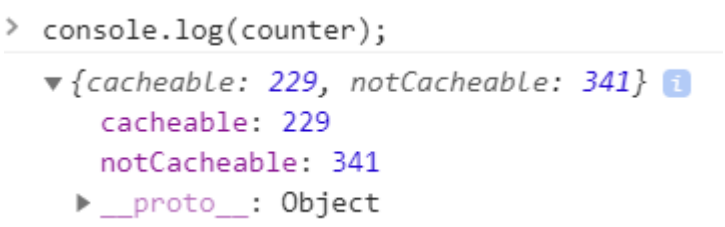

Fig 20: Cacheable vs non-cacheable content on nelsonmandela.org domain

It is clear that for the nelsonmandela.org domain, the majority of the content requested for was triggered by ad service scripts running in the background. As a result, the majority of the content on that website was not cacheable as illustrated above. Therefore, to improve our caching, we need to address the issue of the non-cacheable ads and also override the cache directives seen in the cache-control header. We needed configure Squid in a manner generic enough to cater for many different cases and also be able to override the server directives. The issue with the ads was be addressed, as discussed in the section. For now we will address the issue with the cache-directives through changing of the configuration of Squid.

4.4.4.1 Refresh Patterns

Refresh patterns determine what is saved and served from the cache. Ideally, we would prefer to use the directives provided by the web servers, but as demonstrated above, some web server have caching directives that contradict our requirements. Therefore, we need refresh patterns to override those directives to increase bandwidth savings. The benefit of using refresh patterns is that we create another layer of abstraction which allows us to not focus on the underlying complexity of the web traffic, but just focus more on which content we want to cache. In other words, the page complexity is not taken consideration and all pages follow the same caching criteria defined by Squid, despite their different caching directives.

The usage pattern of the refresh_pattern directive is defined by Squid as shown below:

refresh_pattern [-i] regex min percentage max [options]

[-i]: The regular expressions (regex) are case-sensitive by default, the –I option is to make them case insensitive [85].

Regex: The directive uses regular expressions to describe the type of object it will be applied to. The regular expression can describe a URL, a file type or a file extension [85].

Min (in minutes): This is the minimum time an object without an explicit expiring time should be considered fresh [85].

Percent: this is the percentage of the objects age [85].

Max: This is the maximum time objects without an expiring date will be considered fresh [85].

Options: The options include ignore-no-store and ignore-private, this means that if the webserver has any of these directives in the cache-header Squid will ignore them and cache the object or content regardless [85].

For this experiment, we used the configuration settings specified below. In this experiment, we focus again on Rhodes University , Khan Academy, website and nelsonmandela.org. The configuration file below is a low-level representation of the objectives we have for Squid. In summary, we want Squid to reduce bandwidth consumption and that entails blocking excessive media i.e., images and videos and limiting the need to fetch resources over the network. In the configuration below, Squid is given the directives to cache all media regardless of the cache directives from the server. Khan Academy has the majority of its content served by a Content Delivery Network (CDN), therefore, to improve the results, we added the domain name to the refresh patterns below.

```
152 refresh_pattern ^ftp: 1440 20% 10080
153 refresh_pattern ^gopher: 1440 0% 1440
154 refresh_pattern -i cdn.kastatic.org/.* 10080 90% 43200 override-expire ignore-no-cache ignore-no-store ignore-private
155 refresh_pattern -i www.ru.ac.za/.* 10080 90% 43200 override-expire ignore-no-cache ignore-no-store ignore-private
156 refresh_pattern -i khanacademy.org/.* 10080 90% 43200 override-expire ignore-no-cache ignore-no-store ignore-private
157 refresh_pattern -i www.nelsonmandela.org/.* 10080 90% 43200 override-expire ignore-no-cache ignore-no-store ignore-private
158 refresh_pattern -i .(gif|png|jpg|jpeg|ico)$ 10080 90% 43200 override-expire ignore-no-cache ignore-no-store ignore-private
159 refresh_pattern -i .(iso|avi|wav|mp3|mp4|mpeg|swf|flv|x-flv)$ 43200 90% 432000 override-expire ignore-no-cache ignore-no-store ignore-
    private
160 refresh_pattern -i .(deb|rpm|exe|zip|tar|tgz|ram|rar|bin|ppt|doc|tiff)$ 10080 90% 43200 override-expire ignore-no-cache ignore-no-store
    ignore-private
161 refresh_pattern -i .index.(html|htm)$ 0 40% 10080 override-expire ignore-no-cache ignore-no-store ignore-private
162 refresh_pattern -i .(html|htm|css|js)$ 1440 40% 40320 override-expire ignore-no-cache ignore-no-store ignore-private
163 refresh_pattern . 0 40% 40320
```

Fig 21: Refresh patterns configurations for the experiment

Now, to test the effectiveness of the configuration, we repeated the experiment done earlier with the same environmental conditions but different configuration. As expected, the cache hit ratio increased for the sites added in the configuration file. To be more precise, the hit rate doubled for each website. In the first experiment, Rhodes University, Khan Academy Nelson Mandela and had 40%, 0% and 15% cache hit rates respectively. After introducing the "refresh_pattern" directive, the cache hit ratio for the 3 sites changed to 80%, 88% and 30%. While the outcome of the results is certainly desirable because they align with our objectives, there are some drawbacks that we need to be cognisant of. In most cases, the web server has a better understanding of the content it is serving. What that means is the web master has a better idea about which resources should be cached or should not be cached. Therefore, when

we override the HTTP-cache directives set by the web server, we run the risk of serving stale content. In addition, a more generic cache policy means the majority of web content will be stored and that means the network administrator needs to allocate enough memory to Squid. These are the trade-offs we make to have a higher cache hit rate.

ACCESSED SITE	CONNECT	BYTES	%BYTES	IN-CACHE-OUT		ELAPSED TIME	MILLISEC	%TIME
www.nelsonmandela.org	735	190.35M	49.75%	34.24%	65.76%	00:04:07	247,789	16.91%
cdn.kastatic.org	5.67K	97.96M	25.60%	53.89%	46.11%	00:01:06	66,058	4.51%
www.ru.ac.za	844	47.91M	12.52%	80.25%	19.75%	00:00:14	14,708	1.00%
survey.g.doubleclick.net	79	10.63M	2.78%	52.73%	47.27%	00:00:18	18,612	1.27%
safebrowsing.googleapis.com	7	5.68M	1.48%	0.00%	100.00%	00:00:03	3,497	0.24%
translate.googleapis.com	286	5.01M	1.31%	71.39%	28.61%	00:00:12	12,452	0.85%
www.googletagmanager.com	103	4.22M	1.11%	0.10%	99.90%	00:00:26	26,474	1.81%
use.typekit.net	66	3.28M	0.86%	66.42%	33.58%	00:00:01	1,235	0.08%
www.khanacademy.org	513	3.19M	0.84%	25.36%	74.64%	00:03:00	180,808	12.34%
www.google-analytics.com	276	2.09M	0.55%	62.54%	37.46%	00:00:22	22,278	1.52%
fonts.gstatic.com	136	1.93M	0.51%	55.50%	44.50%	00:00:02	2,273	0.16%
www.gstatic.com	127	1.56M	0.41%	59.40%	40.60%	00:00:01	1,365	0.09%
help.univention.com	66	1.29M	0.34%	0.00%	100.00%	00:02:19	139,265	9.51%
popup.popupsmart.com	29	915.89K	0.24%	48.59%	51.41%	00:00:00	342	0.02%
www.youtube.com	25	849.65K	0.22%	0.00%	100.00%	00:00:02	2,314	0.16%
www.linux.com	24	654.95K	0.17%	0.00%	100.00%	00:00:01	1,297	0.09%
www.mozilla.org	26	596.96K	0.16%	0.00%	100.00%	00:00:00	645	0.04%
feeds.flowsa.net	30	516.42K	0.13%	60.01%	39.99%	00:00:00	637	0.04%
ssl.google-analytics.com	34	319.86K	0.08%	78.81%	21.19%	00:00:00	717	0.05%
apis.google.com	12	310.78K	0.08%	60.85%	39.15%	00:00:00	814	0.06%
cdn.sstatic.net	17	310.21K	0.08%	0.00%	100.00%	00:00:00	177	0.01%
192.168.0.107	86	305.48K	0.08%	66.19%	33.81%	00:00:00	89	0.01%
player.ex.co	1	180.31K	0.05%	0.00%	100.00%	00:00:00	362	0.02%
translate.google.com	51	156.86K	0.04%	74.44%	25.56%	00:00:10	10,351	0.71%
ssl.gstatic.com	5	154.53K	0.04%	34.01%	65.99%	00:00:00	160	0.01%
fonts.googleapis.com	91	148.43K	0.04%	0.00%	100.00%	00:00:19	19,093	1.30%
googleads.g.doubleclick.net	140	145.93K	0.04%	0.00%	100.00%	00:00:27	27,052	1.85%

Fig 22: SARG reports after adding refresh patterns to Squid configuration file

On the other hand, the Nelson Mandela website still has a significantly low cache hit rate and the reason for this is clear, the website has too many background scripts for ad services running. In the next section, we address the problem of excessive ads when working with limited bandwidth.

4.4.5 Blocking Unwanted Ads

After implementing a working solution for the problem of caching in the presence of HTTPS, there still an outstanding issue, which is the issue regarding excessive advertisements during browsing. Online advertising is a major business model for businesses offering free services on the internet [97]. Often that is the only way content providers can generate revenue because these ads allow them to monetize free content. However, these ads have become intrusive and consume a lot of data for the user. An experiment in Ngwane revealed the impact of excessive ads on bandwidth consumption, especially auto play video ads, this is why users have started installing ad blockers. According to a study done by

Enders, ad content accounted for between 18% and 79% of data usage depending on the site you are viewing [98]. In addition, the study shows that about 6% to 68% extra page weight is caused by scripts that were not central to the core of the web page [98]. Below, are the results of the small-scale experiment they conducted. In this experiment, researchers requested 8 pages from popular publishing websites and compared the data usage with an ad blocker, without an ad blocker and with JavaScript disabled. Based on the results above, it is clear that ads and background JavaScript scripts have contributed to the surge in data consumption in recent years.

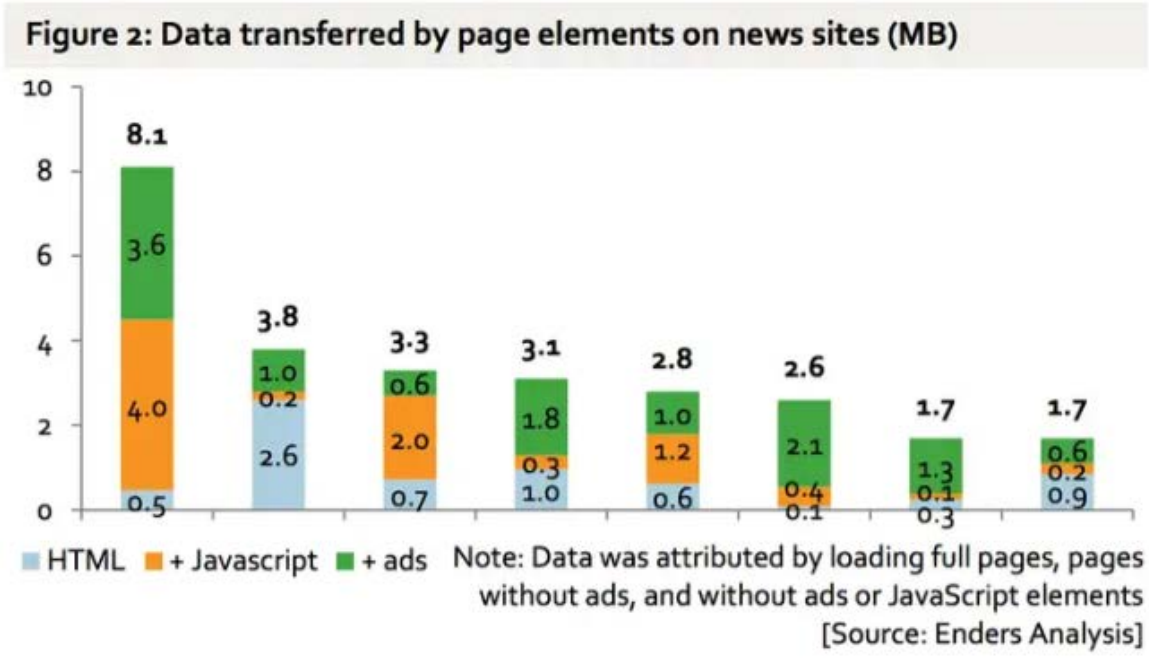

Fig 23: The results from the experiment done by Enders [98]

As a result of the rapid changes in ICT that include fast wired and wireless connections, an increase in resources such as bandwidth and processing power and smartphones, advertising companies do not have the same limitations as before. Advertising companies have access to auto play ads that apparently generate 20 to 50 times more revenue than traditional display ads [99]. With these changes taken into consideration, it is evident why auto play ads have been are an appealing option for ad service businesses. According to Google, 63% of users installed an ad blocker because of too many ads and 48% installed ad blockers because of "annoying ads" [100]. From a technical perspective, "annoying ads" translates to bad user experience. In this context, bad user experience can be a result of slow page load time,

excessive data usage or both with ads. Since we put a lot of emphasis on user experience in our model in SLL, we needed to find a solution to mitigate this problem and improve user experience.

4.4.5.1 Ad-Blockers

For the majority of this chapter, we have explored mainly the negative implications of the evolution of the web. However, for this particular issue, we observe how this evolution has a "self-correcting" nature to it. By this I mean that for every issue that arises with the development of new technology, a solution or two come up. In this case, the solution to excessive ads is ad blockers. An ad blocker is a content filtering application. Technically, the ad-blocker does not "block" ads, but just like Squid, it relies on simple lists (ACLs) that determine which content to download or block. These filter lists are usually maintained by third part communities that are not affiliated with ad-blockers or ad companies.

4.4.5.2 Pi Hole

Pi Hole is a network-ad blocker, which acts as a DNS sinkhole and is intended for use on a private network [101]. A DNS sinkhole is used to provide false DNS information to systems and can allow attackers to redirect users to a malicious destination. However, in this case, we use that feature for non-malicious purposes – to redirect ads, trackers and adware software to non-routable addresses [101]. When we install Pi Hole, it places itself between the device and the upstream DNS server and blocks out or redirects any requests to known ads sources. The advantages of this setup come from the fact that Pi Hole works at the network level. This means that we do not need to install it on every device that is on the network [101]. In addition, we eliminate the need to deal with compatibility issues across multiple devices because unlike traditional ad blocking applications, it does not work at the OS/Application level.

Earlier in the chapter, we described the potential security risks we introduced with our decision to cache HTTPS using the SSL bump feature from Squid. Given the fact that malicious users also use ads as an attack agent, the decision to use Pi Hole also effectively helps us eliminate that potential threat. Most importantly, Pi Hole reduces bandwidth consumption from ads and improves the overall network performance. As you can see from Fig 31, over 50% of ads where blocked before they were downloaded. For the administrator,

Pi hole has an in-built web server that provide an easy to use web interface to monitor various stats on ad blocking.

Fig 24: Pi Hole Admin Dashboard

4.4.5.3 Installing Pi hole

Pi Hole provided a one-step automated install that is activated using the following command:

```
curl -sSL https://install.pi-hole.net | bash
```

Once the command had been executed, Pi Hole's smart installer asked the network administrator a few questions and then it installed.

4.4.5.4 Configuring Pi Hole to work with Squid

Pi Hole and Squid are different services/applications; therefore, they work on different network ports. The network ports identify different applications and allow a computer to run multiple services at the same time. The default port for Pi-Hole is port 53. The default configuration of Squid uses the local machines DNS server for DNS resolution; therefore, it follows that with the installation of Pi Hole – which becomes the machines new DNS server, Squid will also start using Pi Hole as its DNS server. In some instances, the network administrator might have to manually configure the local machines DNS server to point at Pi Hole and that can be done using the following command:

```
sudo nano /etc/resolv.conf
```

Once the configuration file is open, the administrator must add the IP address of Pi Hole at the top of the list to make it the first priority.

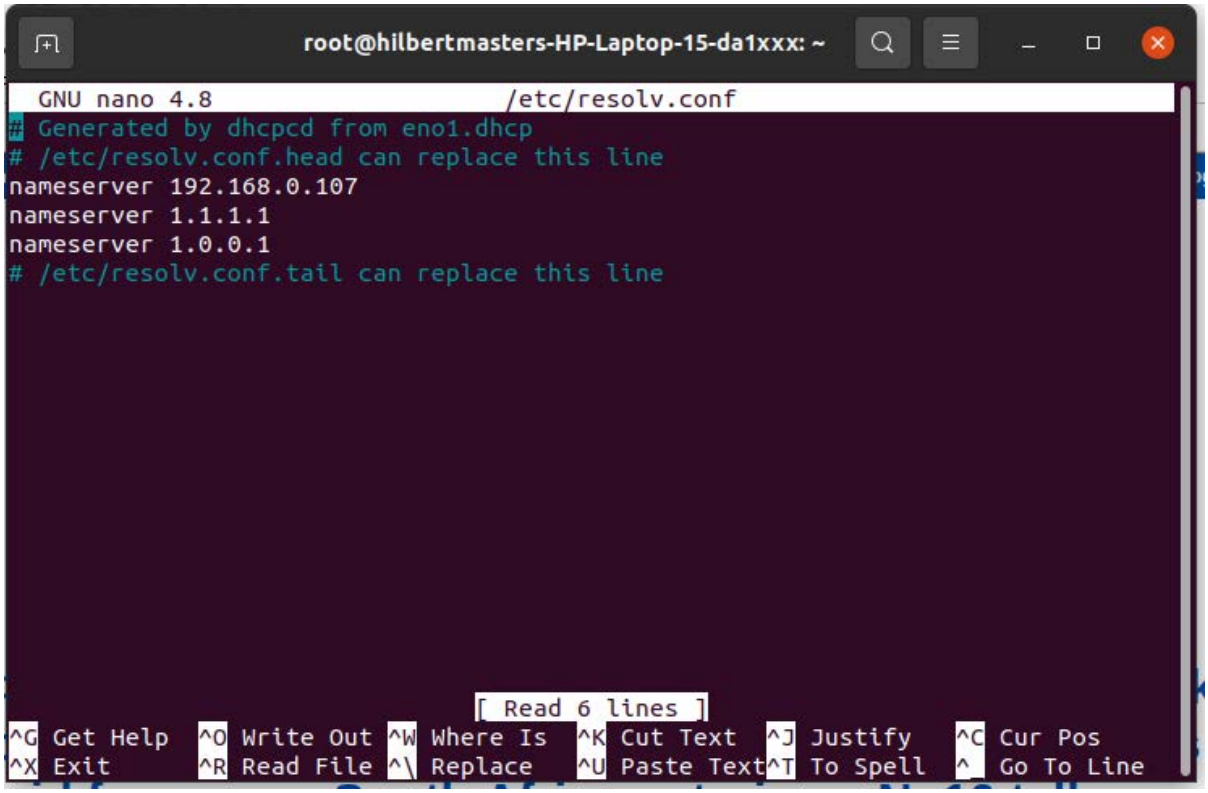

Fig 25: Configuring the local hosts name server to Pi Hole

Lastly, we have to add Pi Holes port as part of the "safe ports" in the configuration file for Squid as done below:

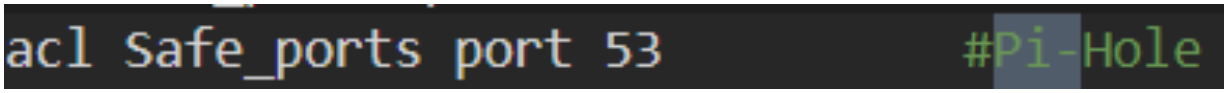

Fig 26: Configuring safe ports

4.4.6 The controversy around ad blockers

The vast majority of websites on the internet are maintained by online advertising. From tiny blogs to established publishing websites, they rely on ad revenue to operate. That is why ad blocking is controversial [102]. On one end, the users want to save data and improve user experience, on the other, websites need revenue to maintain and provide the services to the user. Data from Page Fair states that there has been a steady increase in adoption of ad blockers over the past decade, from 2010 we had approximately 21 million users and now we have more than 181 million users [102]. As a result, websites have pushed back against ad-blockers by placing "bait content" on their pages. The purpose of "bait content" is to detect ad blockers and restrict the user from viewing the website unless they disable their ad blocker as show by Fig 17.

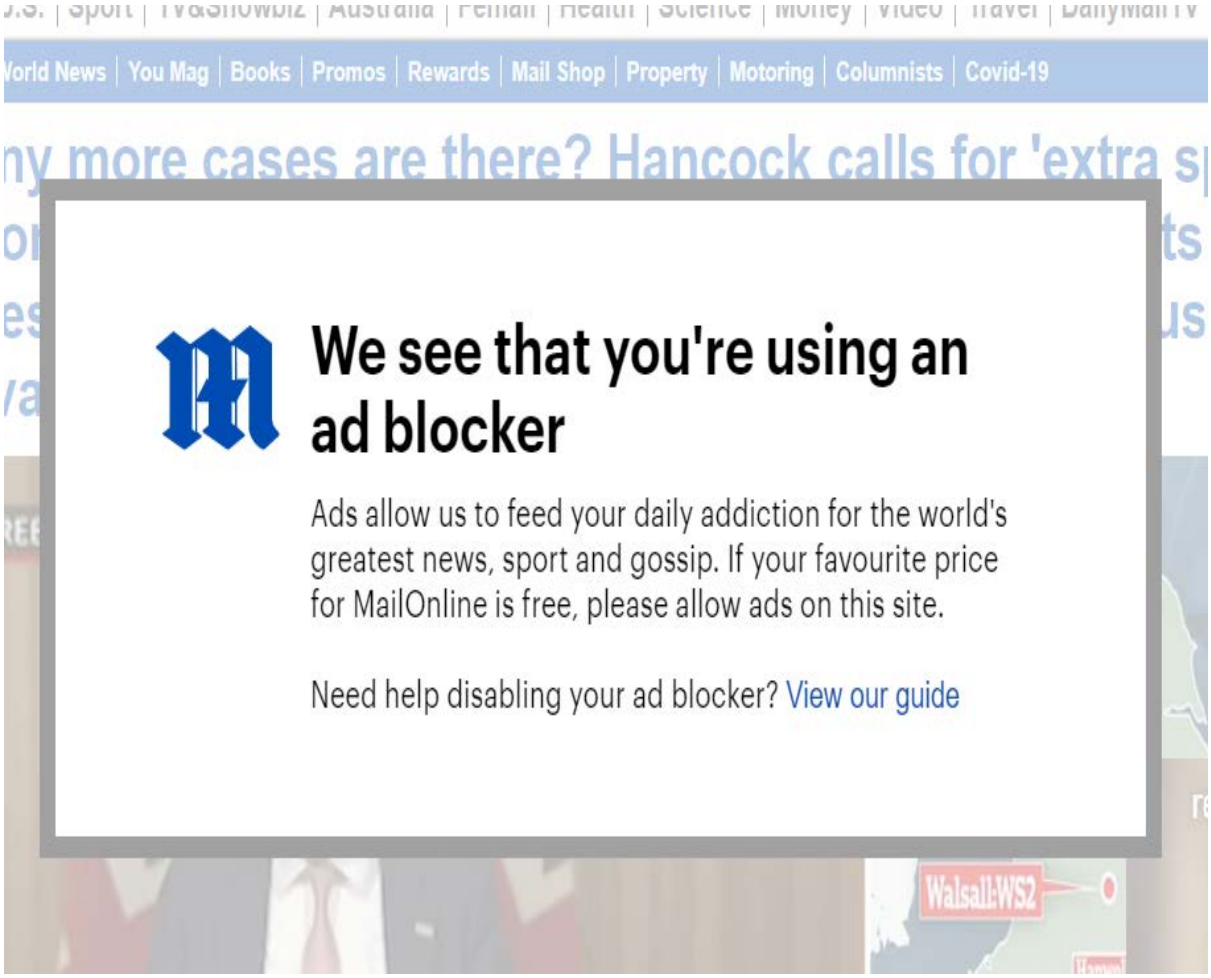

Fig 27: Ad-blocker detection

Pi Hole can circumvent this issue. In fact, you see it with the daily mail website example below, where I used Pi Hole instead of the traditional ad blocker.

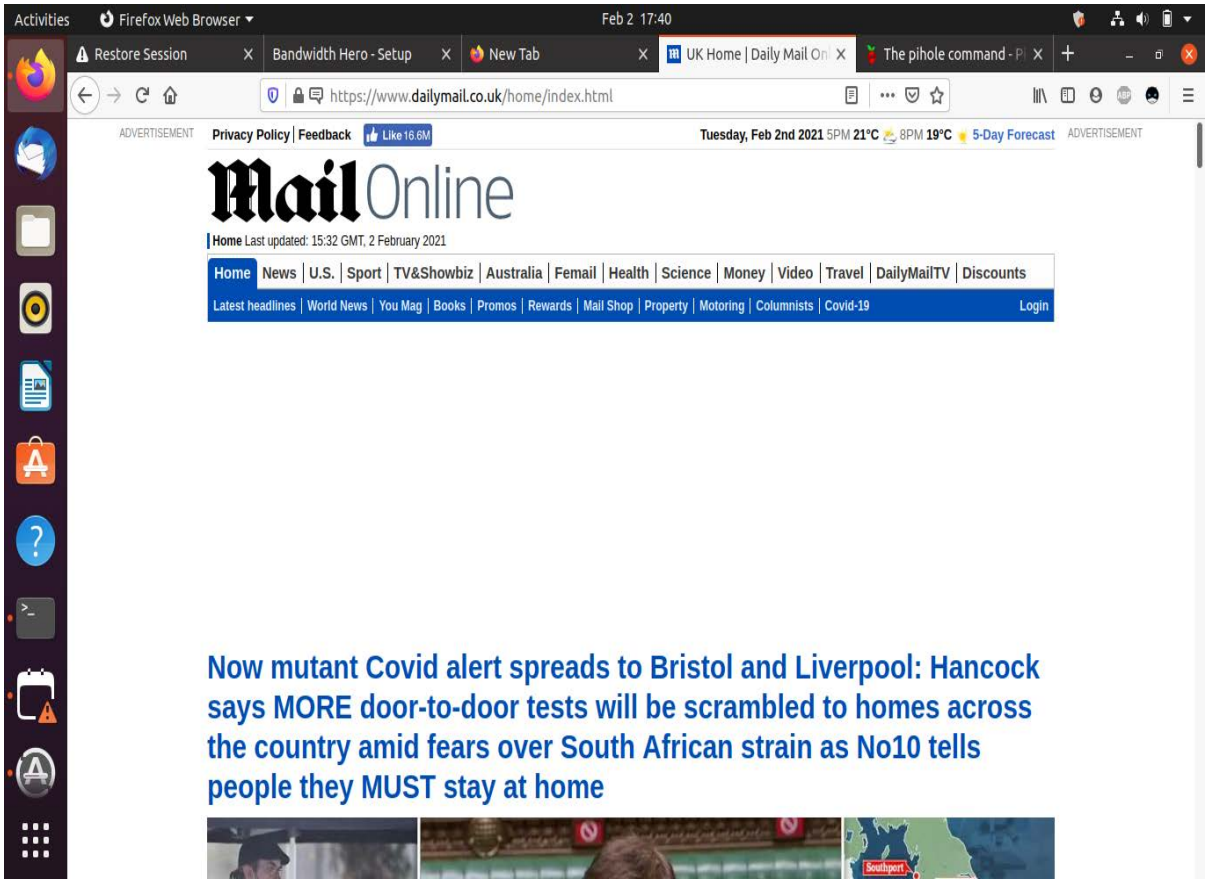

Fig 28: Ads blocked using Pi Hole

This is one of the reasons why Pi Hole is a better option than traditional ad blockers.

4.5 Conclusion

In this chapter we address three main issues; caching encrypted traffic, adjusting caching performance metrics to account for modern web complexity and filtering out excessive ads to improve user experience. For the first issue – allowing Squid to intercept encrypted traffic. We acknowledge the security risks that are introduced by such an approach and we provide justification on the basis of our specific deployment context. HTTP-cache directives to investigate and overwrite them reasonably. Lastly, we addressed the issue with advertisement services and their impact on user experience.

5 Improving User Experience of Web Browsing in the Broadband Island model

In the previous chapter, we stated that page complexity can be broken down into two metrics; content and service level characteristics. In this chapter, we put more emphasis on how the page level characteristics can be used to improve user experience. Page level characteristics comprise page weight and page load time. The various studies within this domain focus more on page load time. In the context of limited data, low bandwidth setting, however we will put more emphasis on page weight.

5.1 Introduction

Over the last 10 years web browsing is one of the main internet services has substantially evolved. We have seen webpages evolve from simple text content from one server to complex dynamic web pages being served from multiple domains [103], [104]. The increase in page complexity is partly due to the introduction of Web2.0 and HTTP/2, which have dramatically influenced the nature of the content and how it is delivered. Web2.0 has resulted in more personalized dynamic web sites e.g., Social Media websites. In addition, we have seen an increase of the user's bandwidth over the years. According to Nielsen's Law of Internet bandwidth, the User's bandwidth grows by 50% per year and there is evidence that shows that the law fits data collected from 1983 to 2019 [105]. With the increase in resources, such as user bandwidth, the introduction of more efficient content delivery protocols and technology, the demand for rich content and more functionality on Web sites has risen. Consequently, page complexity has had to increase consistently to meet user expectations.

Of course, there is anecdotal evidence of users being frustrated by the high page load times and increased bandwidth consumption [95]. To an extent, this outcome is a result of conflicting expectations. In general, the heavier and more functional a webpage is, the higher its Page Load Time (PLT) because we then factor in computational overhead and the bandwidth bottleneck [95]. In some instances, web masters have resorted to rendering web

pages on the server and then simply serving the client with a fully rendered webpage. However, this also limits the number of requests servers can handle, depending on how much computation must be done per request. However, it is worth noting that these issues mentioned have had solutions or partial solutions, such as edge caching, reverse caching, and Server Push (an HTTP/2 feature).

HTTP/2 was recently standardized to optimize the Web by promising faster Page Load Times (PLT) as compared to the standard HTTP/1.1. In addition, the introduction of edge caching and CDNs, has allowed content providers to ensure that they provide fast delivery of their content by serving them from servers that are geographically closer to the end user. Therefore, the increase in page complexity has been coupled with improvements to handle the ever-increasing demands from the end user. However, these solutions fare well only in high resource environments where users have access to high bandwidth and are willing to pay extra for sophisticated infrastructure, faster internet, and have uncapped connections.

In low resource communities, on the other hand, with limited or no access to high bandwidth, uncapped links, the rate of data consumptions impacts user experience more significantly. To be clear, this does not imply that either environment would not have issues arising from PLT or general high data consumption. This only highlights the types of trade-offs likely to be taken in both environments based on the available resources.

The problems mentioned above could be observed during one of our experiments in Ngwane. The data usage from a simple lab experiment reflected that the rate of data consumption was not sustainable. Based on HTTP Archive, we have seen significant increase in the average page weight from the time that we initially setup the BI network [58]. Images and videos are the biggest contributors to this increase. To relate this to what we mentioned earlier, in a low resource environment like Dwesa, where there are limited financial resources, the rate of data consumption does not only impact user experience, but it also determines whether the solution is viable in the long run. Since it has already been established that images and videos contribute significantly to the problem, in this chapter we propose a browser extension that will reduce the page weight of each webpage by blocking the download of unnecessary images and videos

5.2 Page Weight and User Experience

Page weight refers to the total size of a web page including all resources used to create the page. Fig 36 shows in practise what was said earlier: images and videos are responsible for the page weight increase over the past decade.

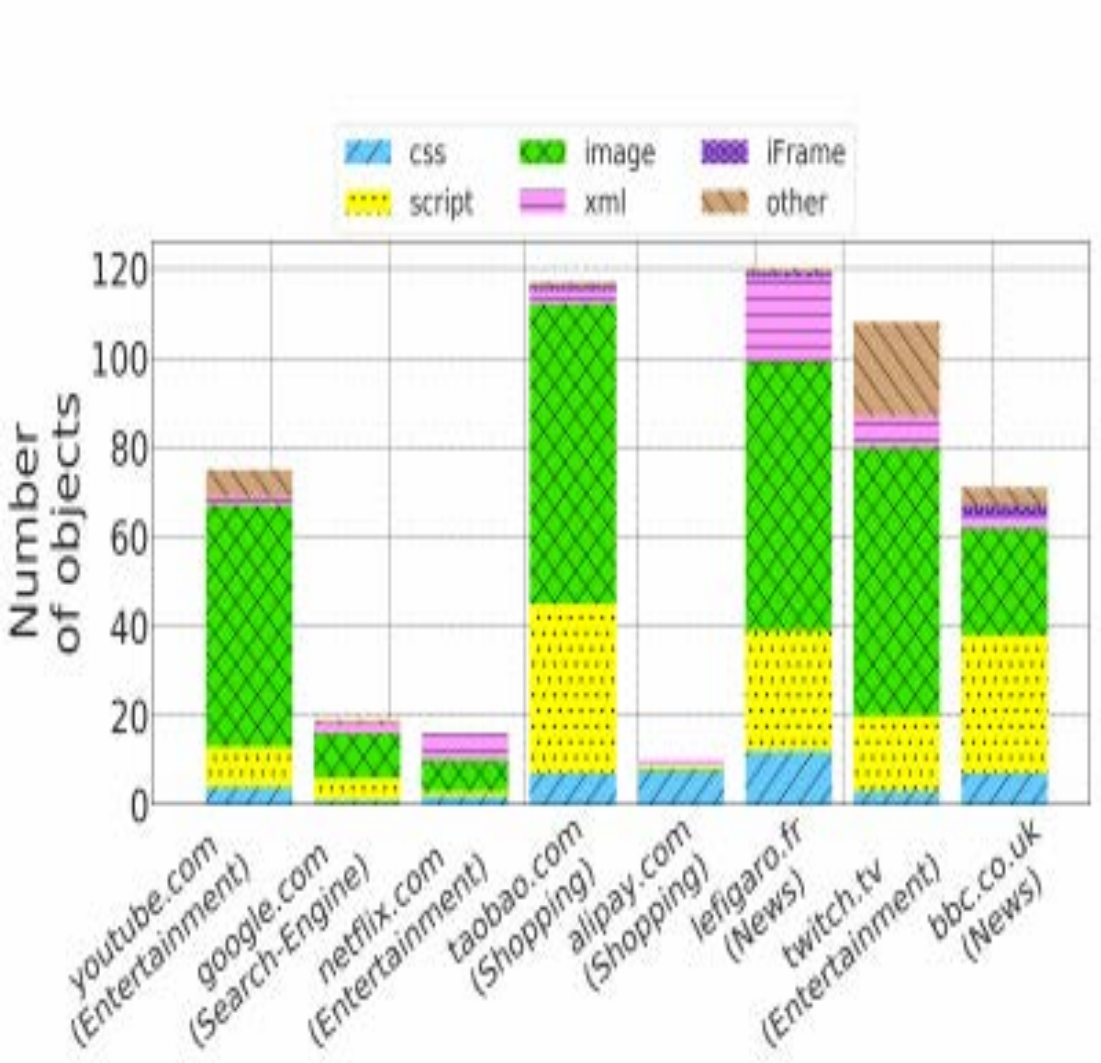

Fig 29: Type of resource contributing to the page weight [103]

To discuss the solution we proposed, we need to explain the architecture of a browser, including extensions.

5.3 Browser Architecture

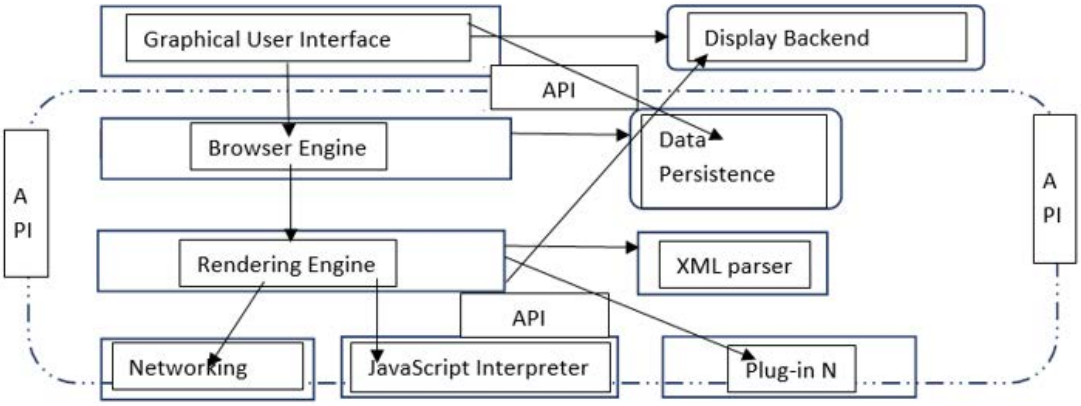

Fig 30: The browser architecture [106]

5.3.1 Graphical User Interface (GUI)

The user interface is the top bar in a browser and that is where the user control for the user lies. The user interface comprises a space to type in a URL, refresh buttons and back/forward buttons. In addition, the user interface displays all active extensions [106].

5.3.2 Browser Engine

The browser engine is the bridge between the rendering engine and the user interface. Based on the inputs from the user, it queries and manipulates the rendering engine [106].

5.3.3 Rendering Engine

The rendering engine starts by getting contents of the requested document from the networking layer and then it goes through this four-step process:

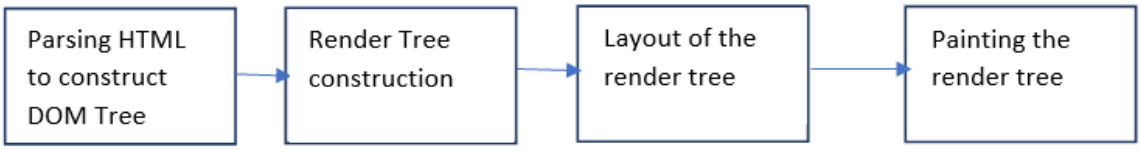

Fig 31: A Flowchart of how the rendering process happens within a browser [106]

The render engine parses through the HTML document looking for HTML symbols and related attributes so that it can convert them into DOM nodes that can be traversed using JavaScript. The engine will also parse through CSS files (inline, embedded and external) to

create a cascading style sheets object model (CSSOM). The styling information and the visual instructions from the HTML will create another tree called the render tree [106] [107] .

The layout process determines where each node will appear on the screen. Finally, the browser engine will traverse through the render tree and paint it below the GUI using the UI backend layer/Display backend [106] [107].

5.3.4 Networking

The networking components handles HTTP requests and all aspects of internet communications. In addition, it implements a cache of retrieved documents to minimize network congestion

5.3.4.1 The Request/Response Cycle

The web is a series of requests and responses between clients and servers. A client is a software component that requests for services readily made available by the server. Normally, a client is a web browser but, in some instances, they can be API's making requests to another server. A client generates a text string containing specifications of the resources it needs from the server, this string is called an HTTP request. This request is containing a URL, which is a string that specifies the resource the client wants from the server [108] [109].

In response to the HTTP request, the server sends an HTTP response. The response contains a status code, which notifies the user whether the request has been successful or not and a message body, which is typically an HTML page that the browser can render. In addition, the HTTP response also contains a header with the metadata related to the response [108].

5.4 Plug-in

5.4.1 Browser Extensions

A browser extension is a typical example of a plug in. A plug in is a software component that extends the features of an existing program without the need to modify its codebase. This allows programmers to implement new features while maintaining the same behaviours. The browser extensions can take advantage of the same APIs that the browser can use on the web page and extend the features of the browser with its own set of APIs. Browser extensions

enable third-party developers to create abilities which extend the browser without the risk of them modifying the source code [106].

5.5 Data Persistence

This is a support for storage mechanisms such as local Storage, indexedDB and the file system. It is a small database created on the local drive of the computer [106].

5.6 How it all works together

Below is a holistic view of the browser architecture that details how the 6 components mentioned in the previous sections all work together to load a webpage.

First, the user types in the user interface to the URL of the webpage or resource they need. The browser engine uses the network process to create an HTTP GET request with the URL of the desired resource or web page. The client side does a DNS Lookup to find the IP address of the host containing the resource specified in the URL [107], [110]. When the host server locates the resource, a HTTP response message is sent back to the browser. If the returned resource is a web page, the browsers render engine parses through the HTML file. Parsing means that the browser will derive the structure or layout of the webpage by processing the various components contained in the response. The HTML file contains the text content of the web page, the structure of the web page and links to other resources (URLs) [107]. The URLs appear in HTML tags for resources needed by the page and are automatically fetched and they appear in hyperlinks where they are fetched on click. As the render engine is parsing the HTML file, it communicates through IPC with the network component to fetch the other resources the web page needs. These resources include CSS (sets the rules for the structure of the page), JavaScript (describes the behaviour of the webpage) and media content which includes images and videos. After the browser parses through the HTML file, CSS files and decodes all media needed for the page, it renders the web page and displays it to the user. In addition, it creates an internal data structure called the document object model[107].

5.7 Browser Extension in SLL

In the previous chapter, we described an experiment in Ngwane and concluded that the strong presence of HTTPS and the fast internet bandwidth have resulted in the BI model being unsustainable. We then implemented a network-level solution to ensure that Squid can cache HTTPS traffic. Considering the fact that the majority of web traffic in recent times is images and videos, in this chapter, we will configure the browser to control the number of media objects requested by the user at a given time. Given the outline of the browser architecture mentioned above, the ideal solution would be to design a plug-in or a browser extension that will stop the render engine from requesting any images or videos linked to the webpage. It is understood that the modern-day browser already has the ability to block images prebuilt into its core software, however, it blocks all images and does not give the user the ability to choose which images they'd like to view. The proposed extension will block all images and give the user the ability to download images based on their preferences, in addition, the extension will modify request headers to ensure that server push is deactivated.

5.7.1 SLL Browser Extension

The SSL extension consists of different, but cohesive, components. These components include, a background script, content script, UI elements and a JavaScript file that maintains its state globally and across different tabs. Based on the description of the SSL extension, this section will use a holistic approach to the analysis of the browser extension in order to emphasize how the constituent components of the extension correlate and are the sum of the system.

5.7.1.1 Manifest file

A browser's extensions architecture heavily relies on its functionality. Therefore, it is important to provide the browser with descriptive data about the files, scripts and permissions the extension will use upon installation [111]. The manifest file fulfils this role by providing metadata of the extension. This metadata includes information that specifies the most important files and the capabilities the extension might use.

```
{
    "manifest_version": 1,
    "name": "SLL Image Blocker",
    "version": "1",

    "description": "A plain text description",

    "background": {
        "scripts": ["jquery-3.4.1.min.js", "background.js"],
        "persistent": true
    },
  "content_scripts" : [
     {
         "matches": [
             "<all_urls>"
         ],
         "js": ["jquery-3.4.1.min.js", "content.js", "state.js"],
         "css": ["content.css"]

     }
 ] ,
 "browser_action" : {
     "default_icon": "images/icons8-lock-50.png",
     "default_title": "Image Blocker",
     "default_popup": "user.html"
 },
 "content_security_policy": "script-
src 'self' https://kit.fontawesome.com/4b9051e741.js; object-src 'self';"
}
```

Fig 32: Manifest file code snippet

The code displayed above or the manifest file contains metadata, which is data that gives useful information about the browser extension. From a simpler perspective, the manifest file is simply a JSON object consisting of attribute—value pairs that describe the architecture browser extension. The manifest file depicted above references the different components of the browser extension, which are content scripts, background scripts and UI elements. At the very top of the file, there is the general description of the extension, this includes the name, the version, and the description.

Fig 33: Permissions to use APIs, block URLs and use chromes storage

An important attribute about this extension is its permissions attribute. As shown above, the extension declares permissions to access the following chrome API's, web Request, webrequestblocking and storage. Declaring permissions in an extension helps to limit the damage if hackers or malware compromises the extension. Rather than initially giving an extension all privileges and then revoking them later, google chrome requires that the extension has very little privileges initially and the burden is on the developer to make sure that they declare permissions as per need [60], [112] [111]. This measure limits the path a hacker or malware might follow to compromise the user's data, i.e. it reduces the attack vector of the software. To determine the permissions needed for this extension, a clear outline of the purpose of the extension was stated, which is to ensure that the browser does not fetch and download images resources when rendering. To achieve this, the browser extension needs access to the network component of the browser extension, which is the component responsible for making all HTTP requests [60], [112]. In addition, the extension is required to work among all hosts that are on the web, therefore, an explicit declaration for that is included. The declaration value uses a wildcard (*) to indicate that the extension will require permissions from all hosts within the web. Lastly, the extension requests permission to access the storage API and the reason is to allow the extension to maintain its state and statistics across multiple browser tabs [60]. In the following section, the paper will discuss the different components of the browser extension and how they are all integrated together. First, I will start off with an abstract view of the system as a way to highlight the core functions of the system. This means I will first discuss the UI and its components because a user interface is technically the most abstract component of any system. In my understanding, the UI provides a gateway for the user to interact with the system with no need to understand the underlying complex system fully. In other words, the UI is the black box of the system.

Therefore, starting with describing it will allow me to give a brief overview of the extension before expanding on the details.

5.7.2 UI Elements

The browser extension is expected to be used by various users, both technical and non-technical users, therefore, the user interface design has to be simple enough for the average user to understand but also offer complex enough information for the practitioners to use. As a rule, a browser extension must only have one clearly defined objective. The SLL extension has one objective, which is to block images and videos to allow the user to save data. To be clear, the SLL extension is not simply just blocking the image from being loaded on the user's webpage but it's ensuring that the images or videos are not transferred over the network or access link. This is a key point because the extension is offering an alternative to the proxy cache. Based on the description mentioned earlier, the proxy cache mainly operates on the network layer, therefore, it is important that the browser extension operates on that layer as well to provide similar results. Upon blocking the images and videos, the browser extension will provide statistics to the user regarding the potential data saved and the total number of images blocked.

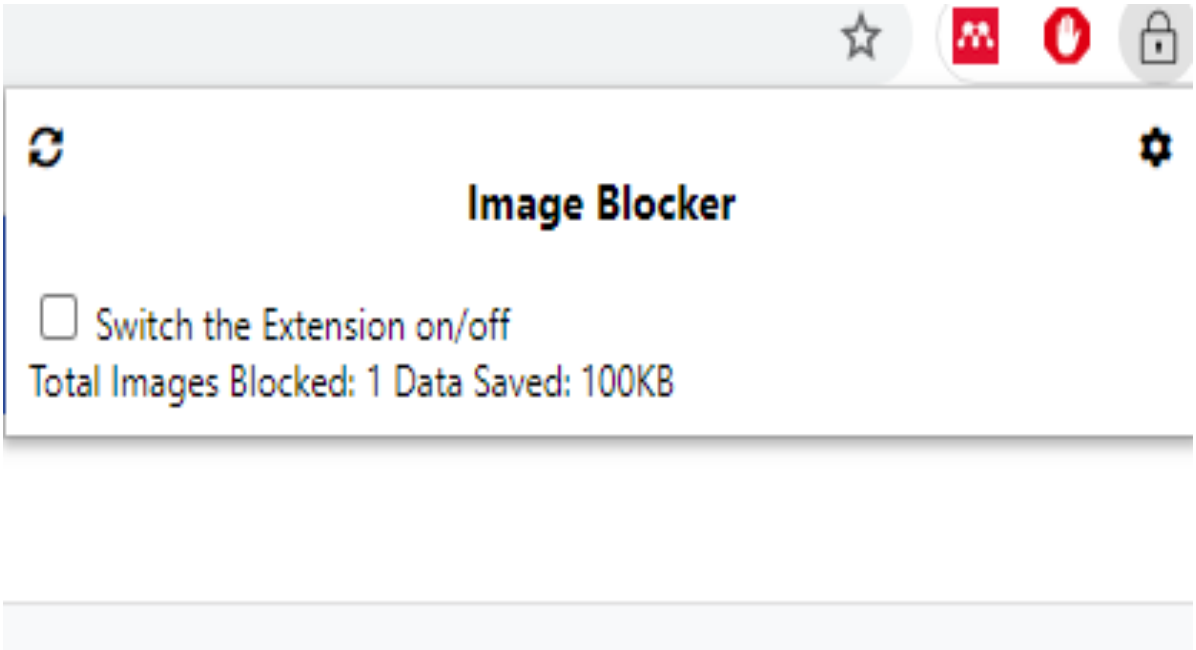

Fig 34: User Interface of the Browser Extension

As illustrated above, the status of the browser extensions provides the user with regular feedback on the state of the system. Much of the design principles used for the user interface are based on Nielsen's 10 heuristics. For example, the choice to use the cog to represent

systems settings and the circular arrows as the refresh button was to leverage the familiarity that comes with these icons as they are widely used in the majority of interfaces. Another example is the minimalistic design is to ensure that all information displayed on the UI is relevant, therefore, allowing the user to spend less time understanding the extension. Lastly, is the visibility of the system status, this is to ensure that the user is informed about what is going on.

5.7.3 Background Scripts

```
var allowMedia = []
var extState;
var blockedMedia = 0;
var totalImages = 0
chrome.storage.sync.set({ "extState": true }, function () {
})
chrome.webRequest.onBeforeSendHeaders.addListener(
  function cancelImageRequest (details) {
    /* The function below checks if the url strings contains any of the extensions stored in the variable 'imageExtensions'.
       If the URL contains the file extension, it returns true, else it returns false
    */
    function removeImages (url) {
      var imageExtensions = 'jpeg,jpg,image,png,gif,mp4,mov,mkv,flv,f4v,avi,avchd,wmv,webm,webp'.split(
        ','
      )
      var hasImage = false
      imageExtensions.forEach(extension => {
        if (url.includes(extension)) {
          hasImage = true
        }
      })
      return hasImage
    }
    var cancelRequest = removeImages(details.url)
    if (extState == false) {
      return { cancel: false }
    } else {
      /* If the browser has added more parameters, a new string that excludes those parameters is created, else the string is used as it is */
      if(details.url.indexOf('?') > 0){
        var newUrl = details.url.substring(0, details.url.indexOf('?'));
        if (allowMedia.indexOf(newUrl) == -1) {
          if (details.type == 'image') {
            blockedMedia++;
            return { cancel: true }
```

```
        }
        else{
          if(cancelRequest){
            blockedMedia++;
          }
          return {cancel: cancelRequest}
        }

      } else if (allowMedia.indexOf(newUrl) > -1) {
        return { cancel: false }
      }
    }
    else {
      if (allowMedia.indexOf(details.url) == -1) {
        if (details.type == 'image') {
          blockedMedia++;
          return { cancel: true }
        }
        else{
          if(cancelRequest){
            blockedMedia++;
          }
          return {cancel: cancelRequest}
        }
      } else if (allowMedia.indexOf(details.url) > -1) {
        return { cancel: false }
      }
    }
  }
 },
 {
  types: ['sub_frame', 'script', 'image', 'xmlhttprequest', 'other'],
  urls: ['<all_urls>']
 },
 ['blocking', 'requestHeaders']
)
chrome.runtime.onMessage.addListener(function (request, sender, sendResponse) {
 var url = request.urlMessage
 var index = allowMedia.indexOf(url)
 if (index > -1) {
  sendResponse({ urlStatus: 'Url has already been added' })
 } else {
  if(url.indexOf('?') > 0){
   var newUrl = url.substring(0, url.indexOf('?'));
   allowMedia.push(newUrl);
      $.ajax({
        url: url,
        type: 'GET',
```

```
        success: function(response){
            console.log(response);
      }
    });
   }
   else {
    allowMedia.push(url);
    $.ajax({
      url: url,
      type: 'GET',
      success: function(response){
          console.log(response);
      }
    });
   }
   sendResponse({ urlStatus: 'Url Added!' })
  }
  return true
})
chrome.runtime.onMessage.addListener(function (request, sender, sendResponse) {
  var checkState
  chrome.storage.sync.get("extState", function (result) {
    checkState = result.extState
  });
  chrome.storage.sync.set({ "extState": request.status }, function (result) {
    sendResponse({ status: `Extension status: ${checkState}` });
  });
  return true;
});
```

Fig 35: Background script for SLL image blocker extension

Based on the code snippet above, an extension is an event-based program that either enhances or modifies the browsing experience [60]. In this case, the events that trigger the background script are based on the network component of the browser extension. This is important to mention because as mentioned before, the browser extension for the SLL is providing an alternative to the proxy cache. The proxy cache will enhance user experience by reducing page load time and allowing the user to save data by storing remote content locally. This is all done in on the network layer, therefore, it makes sense that the browser extension will be listening to the request/response cycle of the web. This will enable it to reduce page load time and save data by cancelling a request objects retrieving media objects and reducing the page weight. Based on the description of the request/response cycle, the browser is simply a client

that makes requests on behalf of the user and displays them below the GUI. The browser provides the extension with the necessary API's it needs to access the network component and below is a diagram that summarizes the events the browser extension can listen to.

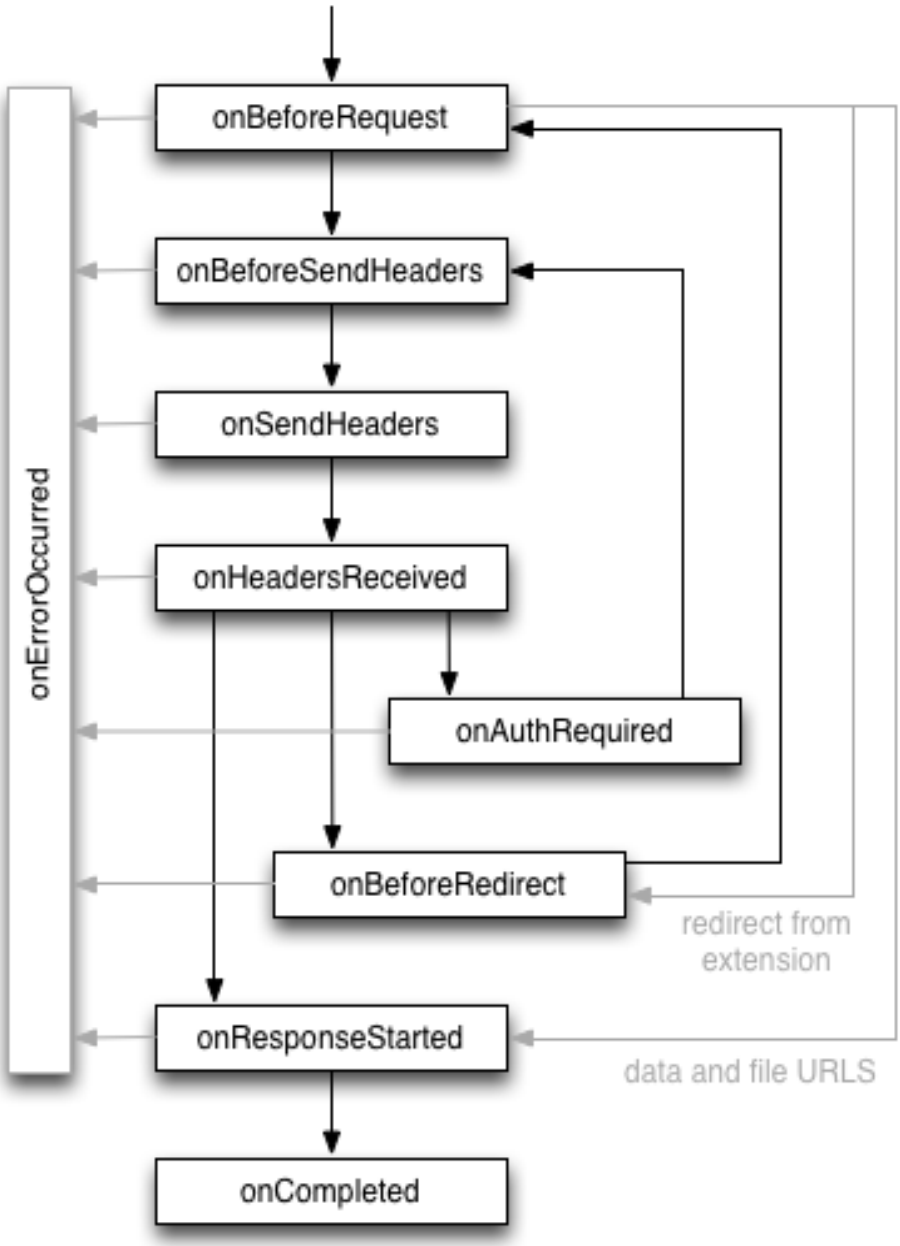

Fig 36: Event types that can be listened to and handled by the webrequestApi [111], [113]

Since the purpose of the SLL extension is to provide similar benefits to the proxy cache, a lot of the design decisions will be based on architecture of a proxy cache. For example, the proxy stores remote content locally to ensure that the expensive access link is used sparingly, therefore, it makes sense that the browser extension would listen to network events before

they are sent over the network. With that said, the SLL extension will be listening for the onBeforeRequest to allow it to filter out any URL's that are fetching images before the images are already transferred over the access link. This will URL filter policy that will cancel any request that has a URL containing an image extension or if the request type is an image. According to HTTP archive, the most common image extension types on the web have the following file extensions: - jpeg, webp, ico, svg. In addition, the policy will also cancel requests with a MIME type containing image [60], [63].

On the other, the background scripts estimate the total data saved during each browsing session using the following equation:

Estimated data saved = Average size of an image x number of images blocked

This is an estimate because it is not possible to know the size of the image unless it is transferred over the network. The average size of the image is based on the data provided by HTTP Archive that states that as of 2020 the average size of an image is 973.4 KB [63].

5.7.3.1 Content Script

Context scripts work in the context of the webpage; therefore, they are treated as if they are not part of the main extension [60] [61] [114]. The content script of the SLL extension is will make changes to the webpages the user visits and it will listen to events and pass messages to the background extension. The goal of the extension is not only to block all images but to also allow the user to pick specific images they would like to view. When the web page is initially loaded, the content script identifies every image tag on the DOM of the webpage and then appends a button node to the parent node of the image. When clicked, this button will enable the user to download the image related to it.

Below is the code written for the content script:

```
var blockedImages = $('.blocked');
chrome.storage.sync.set({ "imageCount": blockedImages.length}, function () {
  console.log(`Number of Images: ${blockedImages.length}`);
})
  }
function addCss(){
  var imagesBlocked = $(".blocked");
  for(var i = 0; i < imagesBlocked.length; i++){
    var url = imagesBlocked[i].src;
    var otherImages = imagesBlocked[i].srcset;
```

```
    var srcSet = imagesBlocked[i].dataset.srcset;
    var downloadButton = document.createElement("button");
    downloadButton.innerHTML = "Download Image"
    downloadButton.id = $(imagesBlocked[i]).id;
    downloadButton.className = "downloadButton";
    var imageLink = document.createElement("a");
    if(url.includes("data:")){

    imageLink.href = imagesBlocked[i].dataset.srcMedium;
    }
    else {
      imageLink.href = url;
    }
    if(typeof srcSet != "undefined" || otherImages.length > 0){
      if(typeof srcSet == "undefined"){
        imageLink.id = otherImages;
      }
      else{
        imageLink.dataset.srcset = srcSet;
      }

    }
    downloadButton.appendChild(imageLink);
    var link = walkTheDom($(imagesBlocked[i]), 10,$(imagesBlocked[i]));
    link.parent().parent().append(downloadButton);
  }
}
function openTab(src){
  const image_window = window.open("", "_blank")
  image_window.document.write(`
        <html>
          <head>
          </head>
          <body>
            <img src=${src} alt="Example" height="50%" width="50%">
        </html>
  `);
 }
function addButton () {
  countImages();
  addCss();
     $('.downloadButton,img').on('click', function () {
       var tagName = $(this)[0].tagName;
       if(tagName == 'BUTTON'){
        var button = $(this)[0];
        var image = $(button).children('a');
        var img = $(image)[0];
        if(typeof img.dataset.srcset != "undefined"){
```

```
          var dataset = img.dataset.srcset.split(',');
          dataset.forEach(src => {
            chrome.runtime.sendMessage({urlMessage: src}, function(response) {
              console.log(response.urlStatus);
            });
          });

        }
        else if(img.id.length != 0) {
          var srcset = img.id.split(',');
          srcset.forEach(src => {
            chrome.runtime.sendMessage({urlMessage: src}, function(response) {
              console.log(response.urlStatus);
            });
          })
        }
        chrome.runtime.sendMessage({urlMessage: img.href}, function(response)
{
          console.log(response.urlStatus);
          img.load(img.href);
        });
       openTab(img.href); //checking
       }
       else {
        var img = $(this)[0];
        chrome.runtime.sendMessage({urlMessage: img.src}, function(response) {
          console.log(response.urlStatus);
          img.load(img.src);
        });
       openTab(img.src);
       }

    });
}

$(window).ready(addButton())
```

Fig 37: Content script for image blocker

Since the content script run in the context of the webpage, it uses a message passing API to communicate with the rest of the extension. This communication is done using an API for one-time requests or in certain instances long-lived connections. The content script will need to communicate with the background script throughout the duration of the users browsing session. This is because the extension will be continuously updating tab statistics for the user and also listening for click events in the case where the user needs to download an image. That is why the content script uses long-lived connections that allow a longer conversation

between the background script and the content script. To establish this connection, both the content and the background script are given a port number that is used for sending and receiving messages throughout the duration of the user's browsing session. As mentioned earlier, the browser extension will provide the user with useful information regarding data saved and images blocked and that will be achieved by allowing communication across different components of the extension.

5.7.3.2 How it all works together:

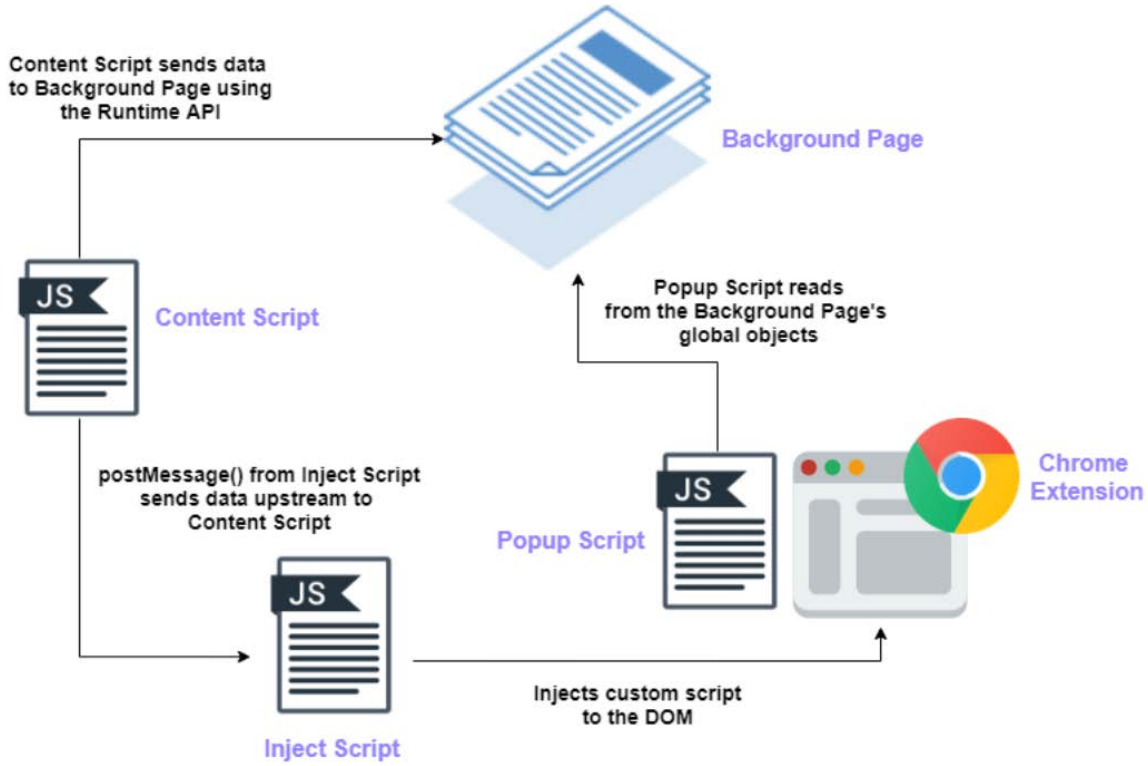

Fig 38: How the different components of the extension work together [115]

The diagram above shows how the different components of the browser extension all work together and communicate for the common goal.

When the page is loaded, the script executes code that traverses through the DOM and adds a download button to next to all blocked images or within the same parent node.

Fig 39: An example of how a website will look after images are blocked

To ensure that the extension works properly, the download button needs to be embedded within close proximity to the image on the DOM and not within a link tag. The Figs 40 and 41 below are from two different websites and their purpose is to show how images are usually embedded in a link or nested within many HTML tags. The depth of the image is based on the number of ancestor nodes it has before getting to the link it's embedded in or the number of ancestors the image has before the <HEAD> tag. As illustrated by the two images, how far an image is nested in other HTML tags varies across different websites and to ensure that there is consistent formatting throughout the assumption will be that each image has a maximum of 10 ancestors before it reaches the <HEAD> tag. The reason why this is important is that if the image tag has a parent or grandparent element that is a link tag, adding the button as a sibling of the image will result in the button itself being rendered as a link. As a result, it interferes with what the button is supposed to do, which is allow a user to view a specific image in a separate tab.

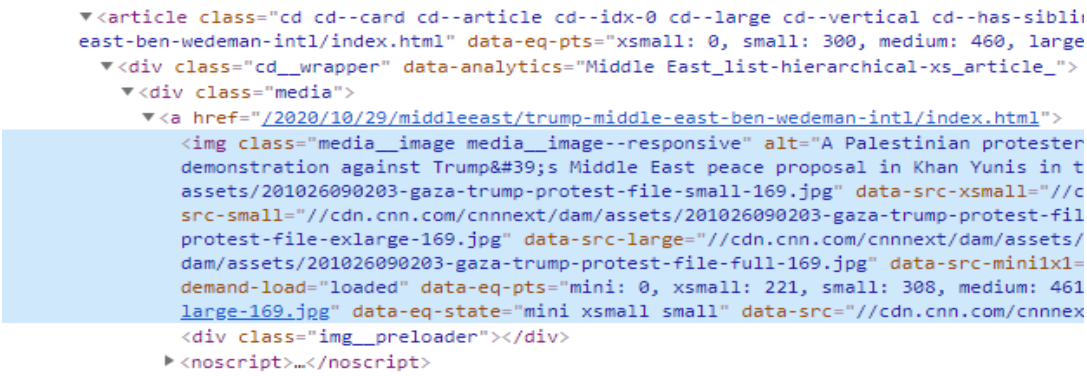

Fig 40: The DOM elements of an image embedded in a link

Fig 41: The DOM elements of an image not deeply embedded or embedded in a link

Below is a snippet of the code used to traverse through the DOM of the various websites and locate where the button tag will be added. A DOM is a recursive data structure, which why a recursive function is used to traverse through it in the figure below. This function has single purpose, to make sure that the image tag is not embedded in a link, if the image tag is embedded in a link, the function returns the node of that link. If the image is not embedded in a link the function will simply return the starting node. However, in certain instances, the image is not nested in a link and it is located low in the DOM tree and this will affect the performance of the extension. This is because recursive functions use a lot of memory since with each call, values are pushed onto the memory stack. Therefore, there is a relationship between performance and the depth of a node on the DOM tree. To solve this, the function has a maximum depth of 10 nodes. This means that once the recursive function has been called 10 times and it still hasn't either come across the <HEAD> tag or the link tag, the assumption is that the image is not nested in a link and it can return the start node. Lastly, the function takes in an argument called startNode and this is there to help the function remember where it started from because the other node argument is being modified.

```
function checkLink(node, depth, startNode){
```

```
  while(node[0]){
    if(depth == 0){
      return startNode;
    }
    if(node[0].nodeName == 'HEAD'){
      return startNode;
    }
    if(node[0].nodeName == 'A'){
      return node;
    }else{
      depth = depth - 1;
      node = node.parent();
      checkLink(node);
    }
  }
  return node;
}
```

Fig 42: Code snippet for checking if an image is embedded in a link

The diagram below represents a use case for when the user loads the extension and attempts to view an image. In this section, the content scripts functions will be broken down in 3 steps.

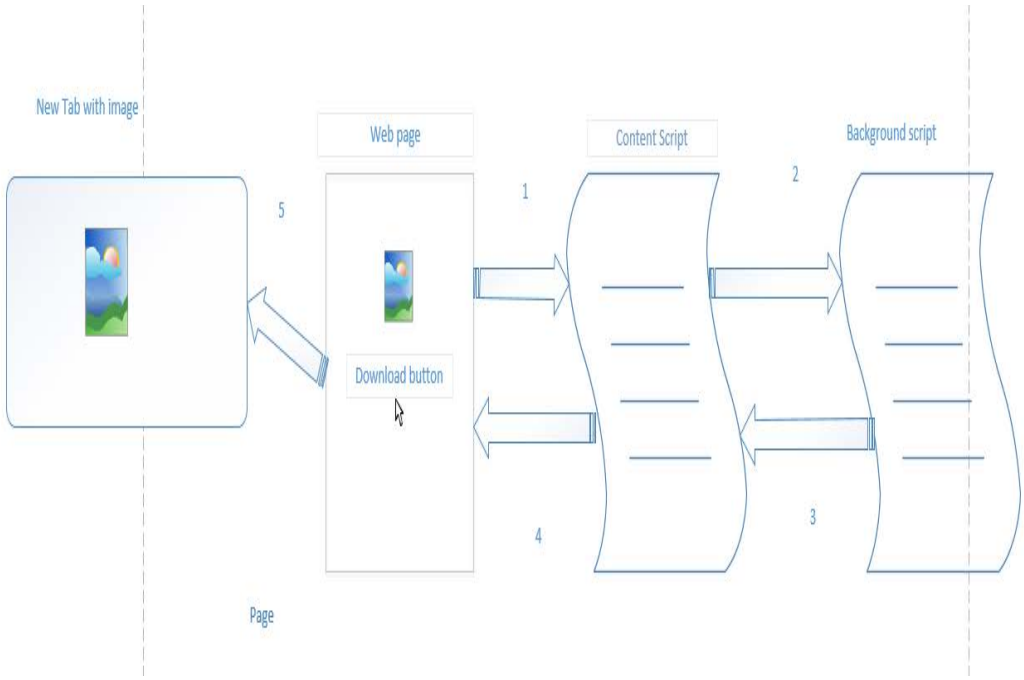

Fig 43: A sequence diagram showing events that take place when a user clicks the download button

When the user clicks on a specific download button, an on click event is triggered in the content script, this event takes the URL of the image stored in a link embedded in the button

and sends it as part of a message object to the background script. The code snippet below illustrates how the first step takes place

```
   $('.downloadButton,img').on('click', function () {
        var tagName = $(this)[0].tagName;
        if(tagName == 'BUTTON'){
         var button = $(this)[0];
         var image = $(button).children('a');
         var img = $(image)[0];
         if(typeof img.dataset.srcset != "undefined"){
           var dataset = img.dataset.srcset.split(',');
           dataset.forEach(src => {
             chrome.runtime.sendMessage({urlMessage: src}, function(response) {
               console.log(response.urlStatus);
             });
           });

         }
         else if(img.id.length != 0) {
           var srcset = img.id.split(',');
           srcset.forEach(src => {
             chrome.runtime.sendMessage({urlMessage: src}, function(response) {
               console.log(response.urlStatus);
             });
           })
         }
         chrome.runtime.sendMessage({urlMessage: img.href}, function(response)
{
           console.log(response.urlStatus);
         });
         openTab(img.href);
        }
        else {
         var img = $(this)[0];
         chrome.runtime.sendMessage({urlMessage: img.src}, function(response) {
           console.log(response.urlStatus);
         });
         openTab(img.src);
        }

     });
}
```

Fig 44: Code snippet for sending messages between browser extension components

The background script has an event listener that listens for a message from the content script and then passes that message to an event handler [62]. This event handler adds the URL to an

array with whitelisted URLs. The whitelisted URLs do not have to be scanned for images or videos, this is to allow users to download particular images while others remain blocked. When the URL is added, the background script sends a response message to the content script alerting it that either the image has been added or it was already added before. Below is a snippet of the event handler in the background script.

```
chrome.runtime.onMessage.addListener(function (request, sender, sendResponse)
{
  var url = request.urlMessage
  var index = allowMedia.indexOf(url)
  if (index > -1) {
    sendResponse({ urlStatus: 'Url has already been added' })
  } else {
    if(url.indexOf('?') > 0){
      var newUrl = url.substring(0, url.indexOf('?'));
      allowMedia.push(newUrl);
      $.ajax({
        url: url,
        type: 'GET',
        success: function(response){
            console.log(response);
        }
     });
    }
    else {
      allowMedia.push(url);
      $.ajax({
        url: url,
        type: 'GET',
        success: function(response){
            console.log(response);
        }
     });
    }
    sendResponse({ urlStatus: 'Url Added!' })
  }
  return true
})
```

Fig 45: Code snippet for background script listener

Finally, the content script creates a new tab with an image tag containing the URL of the image so that the browser can render it.

```
function openTab(src){

 const image_window = window.open("", "_blank")
 image_window.document.write(`
       <html>
         <head>
         </head>
         <body>
           <img src=${src} alt="Example" height="50%" width="50%">
       </html>
 `);
}
```

Fig 46: Opening new tab after download button has been clicked

5.8 Testing Extension Functionality

Independent variable: Disabled/Enabled images in the browser

Dependent variable: Data Usage

Controlled variables: Browser cache (Always must be cleared at the beginning of each experiment), the experiments will all be done on the same website and each experiment will be done in Google Chrome.

Website Name	Website Url	Data Transferred before	Data transferred	Data Saved	Data Saved
Khan Academy	khanacademy.org/	1600	299	1301	81,3
Code Academy	codecademy.com	1300	832	468	36,0
Apex Learning	apexlearning.com/	13400	4700	8700	64,9
Mobymax	mobymax.com/	2700	1300	1400	51,9
ABCya	abcya.com/	2200	1800	400	18,2
Achieve 3000	achieve3000.com/	1600	1021	579	36,2
Cool Math	coolmathgames.com/	2100	1500	600	28,6
PBS kids	pbskids.org/	5100	1300	3800	74,5
Hooda Math	hoodamath.com/	1100	384	716	65,1
Average Data Saved (%)	50,7				
Average Data Saved (KB)	1996				
Average Data Saved (MB)	2,00				

Fig 47 A summary of the results from the experiment with the browser extension

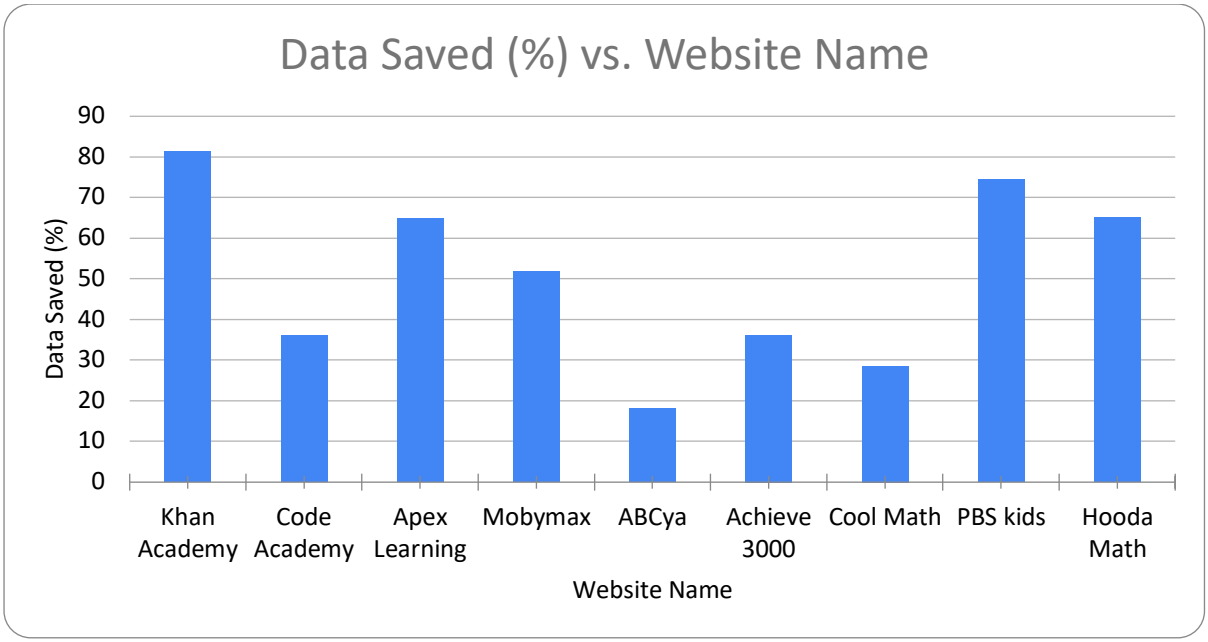

Fig 48: The results shown in Fig 41 in the form of a chart

As illustrated on table 2, the average data saved per webpage is 50.7%. These results confirm that the browser extension is more effective than the browsers in-built image blocking function and this is mainly because the extension also blocks videos. Due to the blocked images and videos, the page weight of the web pages is reduced, as a result, less data is consumed.

5.9 UX Testing of the SLL Browser Extension

Usability refers to the quality of a user's experience when interacting with a system. The aim of this testing it to access the effectiveness and efficiency of this browser extension. The criteria for the testing are accessibility, functionality and user interface. The user interface of the extension has a very simple design which comprises of a single button that allows the user to either enable or disable the extension with ease. The extension automatically reloads the webpage when the state of the system changes i.e., the user starts or stops the extension and this improved the user experience for the users as they were provided with visual cues whenever they interacted with the system.

Porting the extension from Google Chrome to Firefox and Microsoft Edge is a straightforward process because the Web Extension APIs are designed to support cross browser compatibility. This is because most of the Web APIs support call back functions under the chrome namespace. For this reason, the extension is a lightweight and easy to deploy software which is a crucial faction in this context. The browser extension improves

user experience and this can be broadly put into two categories; page load time and data usage. Google has indicated that page load time or page speed is used as a ranking factor when it comes to user experience.

5.10 Conclusion

In conclusion, we have established that the significant increase in images and videos in recent years has negatively impact the user experience when browsing the internet. In cases where the users have enough resources i.e., unlimited bandwidth, high-end devices and high disposable income the effects of this increase are not prominent. However, in a resource setting i.e., low-bandwidth and lower disposable income, these changes significantly impact the user experience. We state that user experience in general, is determined by the page load time and the rate of data consumption. We then proposed that by reducing the page weight, we can positively affect user experience by reducing the data usage and increasing the page load time. Of course, one could argue that lack of images does not necessarily make for an enticing browsing experience, however, it is not an uncommon thing to make such trade-offs when resources are limited. In this instance, we argue that users in a low-bandwidth setting could benefit more from browsing the internet with no images and videos if it means they do not have to incur the cost of constantly purchasing data at an unsustainable rate. We also added a feature on the browser extension that allows the user to selectively download specific images they want to view as they are browsing. The idea to add this feature was as a result of reviewing pre-existing image blocker extensions on google chrome and realizing that the majority of them did not have this feature. We performed some usability lab experiments to determine the extent to which blocking images would affect user experience during a browsing session. The results of these tests confirmed that a lack of images did not negatively impact a user's browsing experience, especially when they still have the option to download specific images based on their needs.

Finally, it is worth noting that the browser extension solution is not limited only to the SLL but can be used in different contexts depending on the needs of various users. The next section is a reflection on this project and we will summarize the key lessons learnt from doing this research.

6 Conclusion

6.1 Summary of Research

This research work sought to determine whether the BI network blueprint from the SLL is still a viable solution to the meet the needs of the marginalized communities in South Africa. This was prompted by the changes in ICT over the past decade that resulted in one of our core link layer technologies – WiMAX, becoming obsolete due to lack of market support. Our viability test involved investigating all three components of the SLL network solution, which consist of the network topology, network technology or infrastructure and methodology (or approach). From the onset, we had established that the use of WiMAX as our backhaul technology was due for an update. The next logical step for us was to experiment with Wi-Fi, which has become a suitable replacement due to the availability of an unlicensed frequency spectrum and the enhancements of antennas over the years. In addition, outdoor Wi-Fi, from an industry perspective, has gained traction as the go to, easy-to-deploy wireless technology for last-mile connection by WISPs. The researchers did field and lab experiments to document quantitative data on the performance of Outdoor Wi-Fi as a last-mile technology.

The viability tests for the network topology included comparisons between the STAR and the MESH network topology. Both these topologies have been studied thoroughly across existing literature and the difference between them are well documented. Therefore, instead of reiterating well-established the advantages and disadvantages of the two types of topologies, in this work, we used various case studies that implemented these topologies as a way to put more emphasis on what makes the various approaches different in practise. Given the premise, the network solution provided by the VillageTelco initiative, in Zenzeleni, was an ideal benchmark to compare our network solution. For starters, both the SLL and VillageTelco have a similar objective, building low-cost community networks in marginalized communities to provide them with affordable internet connectivity. The network solution provided by the VillageTelco initiative, uses the MESH network topology and relies on the availability of community telephone network hardware. With the SLL approach, the end-users simply needs a CPE to establish a connection to the network. Secondly, the SLL topology scales easily to a national level because the hardware used to implement the solution compatible with the majority of devices used in the industry. Overall,

we established that the traditional STAR topology with a LAN/WAN distinction, was still viable in recent times as it met both financial and technical constraints imposed by our context. In addition, since the inception of the SLL, we have seen a wide adoption of similar topologies from WISPs such Khula Tech – a small medium enterprise that aims to provide affordable and reliable internet access to disadvantaged communities of the Eastern Cape. This reinforced our conviction that the SLL network topology is still a viable solution.

The application component in the SLL network solution consisted of value added services, such as web caching, implemented via squid. After performing viability tests on our this component, it was apparent that our in-network proxy - Squid – is no longer effective, given the switch to HTTPS, the strong presence of advertising, and the data weight of current web pages. We mitigated the issue by leveraging Squid's 'bumping' and splicing features; deploying a browser extension to make picture download optional; and using Pi-Hole, a DNS sinkhole to block ads. As it should be expected, our results showed that by blocking images and videos, we decrease the page weight and manage data usage, which improves user experience. When using the browser extension, the users have the option to download an image if they need to. It should be noted that the browser extension can be used outside the BI.

We also did a review of the SLL methodology - which was used to build the blueprint of the network component of SLL. We focused on the dichotomy of the top down and the bottom up approach. We used case studies as a reference point for our comparisons. We reflected on the initial top-down approach proposed by the South African government through an initiative to connect the unconnected using a plan codenamed the 'SA Connect'. To contrast this, the study presents case studies of projects implemented using a bottom up approach. We established that a bottom-up approach solves some of the problems that are introduced by the traditional top-down approach, such as a single point of failure but it also introduces another set of issues. These issues include the fact that it is more difficult to produce a homogeneous solution at a national scale and they rely too much on internal funding, which is usually not possible in our context. The Living Lab methodology on the other hand, is a hybrid of the two, embraces co-creation with the user and encourages community engagement among the researchers, the industry, the government and the beneficiaries. Importantly, this approach takes an iterative or long-term approach to the implementation of network solutions ensuring that these projects get the much-needed support until they're able to sustain themselves.

6.2 Goals and Objectives revisited

In the first chapter, the research reported two main objectives summarized below:

1. Reviewing the blueprint of network component of the Siyakhula Living Labs project
    a. Reviewing the network topology of the SLL network solution
    b. Reviewing the technology used to implement the SLL network solution
    c. Reviewing the methodology used to design and implement the SLL network blueprint
2. Proposing an updated blueprint based on the review done in the first objective

The research managed to achieve both objectives and provided a satisfying answer to the overarching research question – is the BI still a viable solution in recent times ? The short answer is yes, with the exception of certain components of the network layer solution. For the first objective, we did viability tests on the three components of the network layer. The three main components of the network solution – as stated above, are the network topology, network technology or infrastructure and the application and the methodology or implementation approach. On each component we performed viability tests – which are experiments that allow us to

1. Determine whether component or parts of the component are compatible with current technological trends.
2. Compare existing solution or hardware with alternative hardware used in similar case studies
3. Recommend a suitable update based on the results obtained from 1. And 2.

The reviews of each component where both qualitative and quantitative and where all documented in experiments. Each of the three components underwent viability tests and each of the tests either proposed changes/updates or determined that non where necessary.

However, it is worth noting that in some cases, the researchers could not preform adequate field experiments due to the limitations caused by the pandemic. In those instances, we reviewed a component in lab environments that closely resembled field environments.

Lastly, based on the findings from the first objective, the research paper provided an updated blueprint of the BI network.

6.3 Research Contributions

The findings from this work are important because they might contribute to a much larger goal, which is, preparing South Africa for the inevitable arrival of eGovernment. As stated earlier, these are predicated on a country's network readiness; otherwise, ICT will only exacerbate pre-existing socio-economic disparities. With this blueprint, we aim to leverage the benefits of ICT to provide solutions that will transform public service delivery by the government by creating better channels of communication between its citizens, businesses and government entities.

It is also worth mentioning that during the course of the research, we managed to reconnect one of the schools in SLL, to the internet. New hardware such as a Huawei Router and a new wireless AP were introduced to the pre-existing infrastructure to improve connectivity and coverage. The Huawei Router/Modem is an easy to deploy plug n play device that took minimal effort to setup and managed to leverage pre-existing infrastructure at the school and connected easily to the Vodacom cell tower at reasonable speeds. In addition, another school – Ntsika Primary School, in Grahamstown, Makhanda had its connection restored after experiencing technical difficulties. Upon reconnecting the school to the internet, we provided documentation in the form of a network topology diagram of their network solution and we also documented the various ports uses on the switches to connect to the internet.

Lastly, the work reported in this research has resulted in a published article at Supporting International Research Collaboration for Socio-Economic Impact (IST-Africa) and a submission to Southern Africa Telecommunication Networks and Applications Conference (SATNAC).

6.4 Reflections

In this section, we focus on the lessons learnt during the review and in analysis work of community networks in marginalized communities. A core feature of the SLL methodology, is reflection, and this is the case because we take an iterative and long-term approach to the implementation of our solutions, first discuss the trade-offs we had to make based, on our

constraints. Then, we will discuss the importance of documentation and knowledge transfer through community engagement Lastly, we will discuss the importance of leveraging pre-existing relationships within communities or between entities as a way to amplify the impact of ICT in marginalized communities.

Earlier in the thesis, we discussed the three main issues we faced in order to deal with the long-standing issues of the digital divide. The three issues were: accessibility, availability and affordability. As later discovered in our experiments, affordability was a recurring constraint in our decision making during our review of the broadband island. It was through our understanding of this, constraints that we were able to ensure that we propose sustainable and viable solutions to the community. Our first significant trade off was the one between security and data consumption in chapter 4. While we did acknowledge the fact that HTTPS introduced the much-needed security in a more personalized web era, it came at the cost of one of our core in-network services – caching second instance in which we had to make a trade off was when we had to block images and videos with the aim to improve user experience by managing data consumption. These two scenarios illustrate the importance of grounding our work in the context as it is what informed our decisions whenever we needed to make a trade-off.

A second lesson we learnt during this review is the importance of project champions and documentation. During the course of the project, I had to troubleshoot internet connectivity issues at a school in Makhanda, Ntsika Secondary School. Normally, troubleshooting is supposed to be a fairly straight-forward process but we ran into a lot of difficulties due to lack of documentation. For starters, we were the third set of network technicians to work on their network and with no documentation to serve as a paper trail of all changes made previously, we had to do a lot of guess work. Fortunately, we were accompanied by two students-interns who had been working closely with the technicians since the inception of the network. They managed to provide us with vital information regarding the design decisions made by their service provider. By the time we finished working at their school, we provided them with a detailed diagram with their network topology and also suggested that any further changes made to the network be documented in the future.

This was a clear example of why local buy-in when implementing an ICT4D project is important. The knowledge transfer that happened between us and the two student-interns eventually helped us better understand the network and fix it. These local champions can be

very useful for another reason: they will be properly motivated to eventually have the technical capabilities to manage the network with no external help. Because they are also beneficiaries of healthy the network solution, it is in their best interest to ensure that everything is well maintained. Of course, this experience made us recognise the importance of documentation.

Finally, an important perspective we would like to emphasize when dealing with community networks is that community networks do well when researchers and experts leverage pre-existing infrastructure and social networks within the community. For example, in the BI, SLL takes advantage of the fact that schools are generally focal points within a community. In general, this is where the more technically adept demographic of a population is likely to exists. By choosing to setup our core nodes i.e., Ngwane and Mpume at these locations, we can take advantage of pre-existing infrastructure like laboratories and reliable power supply. In addition, schools are generally central to the majority of the population, therefore, we improve the accessibility and the availability of the network to the majority of the population.

While we are still discussing the issues of leveraging pre-existing relationships, it is worth noting that we took a similar approach with our business model for the SLL. Governments are primarily supposed to provide services to the citizens. Therefore, it comes as no surprise that in we already had pre-existing communication channels between the government and the Dwesa Community. Instead of trying to establish a new channel of communication, with the SLL network solution, we simply leverage the existing one and charge an amount to the government whenever it uses the channel to reach these communities. The benefit of this is the fact that there is already a pre-existing budget for national campaigns or service delivery targeted towards the4se communities, so receiving the funding will not be difficult

6.5 Future Work Recommendations

The of the research is an updated blueprint for the BI model in Dwesa based on the traditional network approach. As for future work, we recommend to do more research into software defined networks. Although we have stated in this work that they are currently not immediately a viable solution, they are bound to become viable at a certain point in the future.

One aspect of the software defined networks is their flexibility and capacity to cope easily with changing workloads and requirements. This aspect is a good match to the characteristics of poor areas.

Secondly, the work in this research is centred around using wireless technologies to implement BI model. For future work, we recommend to do more research into introducing fibre to rural and peri-urban communities. The arrival of fibre in Grahamstown via Khula Tech is one of many signs that fibre might have a stronger presence in the future.

References

[1] A. Terzoli, I. Siebörger, and S. Gumbo, 'Community " Broadband Islands " for digital government access in rural South Africa', presented at the Proceedings of the 17th European Conference on Digital Government, 2017.

[2] V. Gunasekaran and F. C. Harmantzis, 'Emerging wireless technologies for developing countries', *Technology in Society*, vol. 29, no. 1, pp. 23–42, 2007, doi: 10.1016/j.techsoc.2006.10.001.

[3] A. Terzoli, I. Siebörger, M. Tsietsi, and S. Gumbo, 'Digital Inclusion : A model for e-Infrastructure and e- Services in Developing Countries', in *Lecture Notes of the Institute for Computer Sciences, Social- Informatics and Telecommunications Engineering.*, 2018, pp. 85–98.

[4] Gumbo, Sibukele, Terzoli, Alfredo, and Mamello, Thinyane, 'Living Labs as South Africa's enabler for ICT services creation: The Siyakhula Living Lab experience', in *Southern Africa Telecommunication Networks and Applications Conference (SATNAC) 2013*, 2013, no. September, p. 279.

[5] A. Beamish, 'Communities on-line: Community-based computer networks', Massachusetts Institute of Technology, 1995.

[6] S. Gumbo, H. Thinyane, M. Thinyane, A. Terzoli, and S. Hansen, 'Living Lab Methodology as an Approach to Innovation in ICT4D: The Siyakhula Living Lab Experience', in *IST-Africa 2012 Conference Proceedings*, 2012, p. 9.

[7] L. Philip, C. Cottrill, J. Farrington, F. Williams, and F. Ashmore, 'The digital divide: Patterns, policy and scenarios for connecting the "final few" in rural communities across Great Britain', *Journal of Rural Studies*, vol. 54, pp. 386–398, Aug. 2017, doi: 10.1016/j.jrurstud.2016.12.002.

[8] C. Gibbs, 'Court: Sprint can begin to kill WiMAX network this week', *Court: Sprint can begin to kill WiMAX network this week*, Jan. 02, 2016. https://www.fiercewireless.com/wireless/court-sprint-can-begin-to-kill-wimax-network-week (accessed Nov. 11, 2020).

[9] V. Ndou, 'E – GOVERNMENT FOR DEVELOPING COUNTRIES:OPPORTUNITIES AND CHALLENGES', *Electronic Journal of Information Systems in Developing Countries*, vol. 18, pp. 1–24, Jan. 2004.

[10] M. W. L. Fong, 'Digital Divide: The Case of Developing Countries', *Issues in Informing Science and Information Technology*, vol. 6, no. 2, Art. no. 2, 2009.

[11] O. Evans, 'Digital government: ICT and public sector management in Africa', 2019, pp. 269–286.

[12] T. Janowski, 'Digital government evolution: From transformation to contextualization', *Government Information Quarterly*, vol. 32, no. 3, pp. 221–236, Jul. 2015, doi: 10.1016/j.giq.2015.07.001.

[13] V. Gunasekaran and F. C. Harmantzis, 'Emerging wireless technologies for developing countries', *Technology in Society*, vol. 29, no. 1, pp. 23–42, Jan. 2007, doi: 10.1016/j.techsoc.2006.10.001.

[14] M. Marais, 'ICT4D and Sustainability', 2015. doi: 10.1002/9781118290743/wbiedcs038.

[15] R. W. Harris, 'How ICT4D Research Fails the Poor', *Information Technology for Development*, vol. 22, no. 1, pp. 177–192, Jan. 2016, doi: 10.1080/02681102.2015.1018115.

[16] R. Heeks, *Development Informatics Working Paper Series*, no. July. 2017. doi: 10.13140/RG.2.2.31885.90084.
[17] M. Daradkah, E. Al Jounidy, and A. Qusef, 'Top-Down vs. Bottom-Up in project management: A practical model', *ACM International Conference Proceeding Series*, 2018, doi: 10.1145/3234698.3234709.
[18] D. L. Johnson and K. Roux, 'Building rural wireless networks: Lessons learnt and future directions', *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, pp. 17–22, 2008, doi: 10.1145/1410064.1410068.
[19] A. Terzoli, I. Siebörger, and S. Gumbo, 'Community " Broadband Islands " for digital government access in rural South Africa', 2017.
[20] R. Baig, L. Dalmau, R. Roca, L. Navarro, F. Freitag, and A. Sathiaseelan, 'Making community networks economically sustainable: The Guifi.net experience', *GAIA 2016 - Proceedings of the 2016 Global Access to the Internet for All Workshop, Part of SIGCOMM 2016*, pp. 31–36, 2016, doi: 10.1145/2940157.2940163.
[21] D. Vega, L. Cerdà-Alabern, L. Navarro, and R. Meseguer, 'Topology patterns of a community network: Guifi.net', in *International Conference on Wireless and Mobile Computing, Networking and Communications*, 2012, pp. 612–619. doi: 10.1109/WiMOB.2012.6379139.
[22] D. L. Johnson and K. Roux, 'Building rural wireless networks: Lessons learnt and future directions', in *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, 2008, pp. 17–22. doi: 10.1145/1410064.1410068.
[23] C. Rey-Moreno, Z. Roro, M. J. Siya, J. Simo-Reigadas, N. J. Bidwell, and W. D. Tucker, 'Towards a Sustainable Business Model for Rural Telephony', in *Conference: Third International Workshop about Research in ICT for Human Development.*, 2012, p. 14.
[24] M. Adeyeye and P. Gardner-Stephen, 'The Village Telco project: A reliable and practical wireless mesh telephony infrastructure', *Eurasip Journal on Wireless Communications and Networking*, vol. 2011, Aug. 2011, doi: 10.1186/1687-1499-2011-78.
[25] C. Rey-Moreno, W. D. Tucker, N. J. Bidwell, Z. Roro, M. J. Siya, and J. Simo-Reigadas, 'Experiences, challenges and lessons from rolling out a rural WiFi mesh network', in *Proceedings of the 3rd ACM Symposium on Computing for Development, DEV 2013*, New York, New York, USA, 2013, pp. 1–1. doi: 10.1145/2442882.2442897.
[26] A. Sathiaseelan *et al.*, 'Towards Decentralised Resilient Community Cloud Infrastructures', doi: 10.1145/1235.
[27] A. Lertsinsrubtavee, N. Weshsuwannarugs, N. Tansakul, A. Taparugssanagorn, and K. Kanchanasut, 'Wireless edge network for sustainable rural community networks', *ANRW 2018 - Proceedings of the 2018 Applied Networking Research Workshop*, pp. 40–42, 2018, doi: 10.1145/3232755.3232782.
[28] L. Cerdà-Alabern, 'On the topology characterization of Guifi.net', in *International Conference on Wireless and Mobile Computing, Networking and Communications*, 2012, pp. 389–396. doi: 10.1109/WiMOB.2012.6379103.
[29] R. Baig, R. Roca, F. Freitag, and L. Navarro, 'Guifi.net, a crowdsourced network infrastructure held in common', *Computer Networks*, vol. 90, pp. 150–165, Oct. 2015, doi: 10.1016/j.comnet.2015.07.009.
[30] C. Ball, 'LTE and WiMax Technology and Performance Comparison Contents ':, 2007.

[31] R. Flickenger, S. Okay, E. Pietrosemoli, M. Zennaro, and C. Fonda, 'Very long distance Wi-Fi networks', *Proceedings of the ACM SIGCOMM 2008 Conference on Computer Communications -2nd ACM SIGCOMM Workshop on Networked Systems for Developing Regions, NSDR'08*, no. May 2014, pp. 1–5, 2008, doi: 10.1145/1397705.1397707.
[32] S. Jain and D. P. Agrawal, 'Wireless community networks', *Computer*, vol. 36, no. 8, pp. 90–92, 2003, doi: 10.1109/MC.2003.1220588.
[33] S. Aust, R. V. Prasad, and I. G. M. M. Niemegeers, 'Outdoor Long-Range WLANs: A Lesson for IEEE 802.11ah', *IEEE Communications Surveys and Tutorials*, vol. 17, no. 3, pp. 1761–1775, 2015, doi: 10.1109/COMST.2015.2429311.
[34] P. Bhagwat, B. Raman, and D. Sanghi, 'Turning 802.11 inside-out', *Computer Communication Review*, vol. 34, no. 1, pp. 33–38, 2004, doi: 10.1145/972374.972381.
[35] R. Flickenger, S. Okay, E. Pietrosemoli, M. Zennaro, and C. Fonda, 'Very long distance Wi-Fi networks', *Proceedings of the ACM SIGCOMM 2008 Conference on Computer Communications -2nd ACM SIGCOMM Workshop on Networked Systems for Developing Regions, NSDR'08*, no. May 2014, pp. 1–5, 2008, doi: 10.1145/1397705.1397707.
[36] R. Flickenger, 'Building Wireless Community Networks'.
[37] B. Y. M. Casado, N. Foster, and A. Guha, 'Abstractions for Software- Defined Networks', *Communications of the ACM 57(10):86-95*, Sep. 2014.
[38] S. Hasan, Y. Ben-David, U. C. Berkeley, A. C. Scott, E. Brewer, and S. Shenker, *Enhancing Rural Connectivity with Software Defined Networks*. 2013.
[39] J. Liew, A. W. Yeo, K. Ab, and A. Othman, 'Implementation of Wireless Networks in Rural Areas', *Work with Computing System*, no. June, pp. 282–285, 2004.
[40] L. T. Gwaka, J. May, and W. Tucker, 'Towards low-cost community networks in rural communities: The impact of context using the case study of Beitbridge, Zimbabwe', *Electronic Journal of Information Systems in Developing Countries*, vol. 84, no. 3, pp. 1–11, 2018, doi: 10.1002/isd2.12029.
[41] J. Frankenfiedl, 'Very Small Aperture Terminal (VSAT)', 2019. https://www.investopedia.com/terms/v/vsat.asp (accessed Aug. 09, 2020).
[42] B. Irwin, I. Siebörger, and D. Wells, 'Bandwidth management and monitoring for community networks', *The Southern Africa Telecommunication Networks and Applications Conference (SATNAC), Spier Estate, Stellenbosch, South Africa*, 2010.
[43] T. Bhandare, 'LTE and WiMAX Comparison', no. December, 2008.
[44] Y. Zhang, K.-S. Tan, P.-Y. Kong, J. Zheng, and M. Fujise, *IEEE 802.16 WiMAX Mesh Networking*, no. March 2014. 2006. doi: 10.1201/9781420013542.ch13.
[45] S. Banerij and Chowdhury, 'S. Banerji and R.S. Chowdhury, "Wi-Fi &WiMAX: A Comparative Study".' http://www.sciepub.com/reference/14935 (accessed Aug. 17, 2020).
[46] J. Akhtar, N. M. Sheikh, and S. A. U. N. Abbas, 'Architecture of WiFi Based Broadcast Network for Rural Community', vol. 35, no. 2, pp. 199–208, 2016.
[47] I. Aldmour, 'LTE and WiMAX : Comparison and Future Perspective', vol. 2013, no. November, pp. 360–368, 2013.
[48] S. Jain and D. P. Agrawal, 'Wireless Community Networks', pp. 90–92.
[49] J. Simo-Reigadas *et al.*, 'Sharing low-cost wireless infrastructures with telecommunications operators to bring 3G services to rural communities', *Computer Networks*, vol. 93, pp. 245–259, Dec. 2015, doi: 10.1016/j.comnet.2015.09.006.
[50] W. Konhäuser, 'Broadband wireless access solutions - Progressive challenges and potential valueof next generation mobile networks', *Wireless Personal*

Communications, vol. 37, no. 3–4, pp. 243–259, 2006, doi: 10.1007/s11277-006-9076-z.
[51] M. Butkiewicz, H. V. Madhyastha, and V. Sekar, 'Understanding Website Complexity: Measurements, Metrics, and Implications', in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, 2011, pp. 313–328. [Online]. Available: http://www.cs.ucr.edu/~harsha/web_complexity/.
[52] S. Ihm and V. S. Pai, 'Towards understanding modern Web traffic', *Performance Evaluation Review*, vol. 39, no. 1 SPEC. ISSUE, pp. 143–144, 2011, doi: 10.1145/1993744.1993797.
[53] S. Ihm and V. S. Pai, 'Towards understanding modern Web traffic', in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, 2011, vol. 39, pp. 143–144. doi: 10.1145/1993744.1993797.
[54] D. Naylor *et al.*, 'The cost of the "s" in HTTPS', *CoNEXT 2014 - Proceedings of the 2014 Conference on Emerging Networking Experiments and Technologies*, pp. 133–139, 2014, doi: 10.1145/2674005.2674991.
[55] A. Brylinski, 'Overview of HTTP / 2', 2017.
[56] T. Zimmermann, B. Wolters, O. Hohlfeld, and K. Wehrle, 'Is the web ready for HTTP/2 server push?', *CoNEXT 2018 - Proceedings of the 14th International Conference on Emerging Networking EXperiments and Technologies*, pp. 13–19, 2018, doi: 10.1145/3281411.3281434.
[57] J. P. Mulerikkal and I. Khalil, 'An architecture for distributed content delivery network', 2007, pp. 359–364.
[58] D. Robinson, *Content delivery networks: fundamentals, design, and evolution*. John Wiley & Sons, 2017.
[59] 'Alternative Options to Caching Web Traffic'. https://help.zscaler.com/zia/what-are-alternatives-caching-web-traffic (accessed Dec. 11, 2020).
[60] 'Getting Started Tutorial - Google Chrome'. https://developer.chrome.com/extensions/getstarted (accessed Nov. 15, 2020).
[61] 'Browser Extensions - Mozilla | MDN'. https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions (accessed Nov. 15, 2020).
[62] P. Mehta, 'API Availability and Messaging', in *Creating Google Chrome Extensions*, P. Mehta, Ed. Berkeley, CA: Apress, 2016, pp. 79–144. doi: 10.1007/978-1-4842-1775-7_3.
[63] 'HTTP Archive'. https://httparchive.org/ (accessed Nov. 15, 2020).
[64] A. Al-Dailami, C. Ruan, Z. Bao, and T. Zhang, 'QoS3: Secure Caching in HTTPS Based on Fine-Grained Trust Delegation', *Security and Communication Networks*, vol. 2019, 2019, doi: 10.1155/2019/3107543.
[65] 'Introduction to QoS (Quality of Service)', *NetworkLessons.com*, Apr. 21, 2017. https://networklessons.com/cisco/ccna-routing-switching-icnd2-200-105/introduction-qos-quality-service (accessed Jul. 25, 2021).
[66] A. Al-Dailami, C. Ruan, Z. Bao, and T. Zhang, 'QoS3: Secure Caching in HTTPS Based on Fine-Grained Trust Delegation', *Security and Communication Networks*, vol. 2019, 2019, doi: 10.1155/2019/3107543.
[67] S. Africa and T. Networks, *2013 Proceedings*, no. September. 2013.
[68] 'Star Topology - an overview | ScienceDirect Topics'. https://www.sciencedirect.com/topics/computer-science/star-topology (accessed Nov. 19, 2020).
[69] A. Beamish and B. Arch, 'COMMUNITIES ON-LINE ':, vol. 0, 1995.

[70] Z. Roro, C. Rey-Moreno, W. D. Tucker, and M. J. Siya, *Socio-economic aspects of voice-over-IP technology in rural SA*. Telkom, 2012. Accessed: Apr. 15, 2021. [Online]. Available: http://repository.uwc.ac.za/xmlui/handle/10566/594
[71] B. Y. M. Casado, N. Foster, and A. Guha, 'Abstractions for Software- Defined Networks'.
[72] A. D. Ferguson, A. Guha, C. Liang, R. Fonseca, and S. Krishnamurthi, 'Hierarchical policies for software defined networks', *HotSDN'12 - Proceedings of the 1st ACM International Workshop on Hot Topics in Software Defined Networks*, pp. 37–42, 2012, doi: 10.1145/2342441.2342450.
[73] M. Zhang and R. S. Wolff, 'Crossing the Digital Divide: Cost-Effective Broadband Wireless Access for Rural and Remote Areas', *IEEE Communications Magazine*, vol. 42, no. 2, pp. 99–105, 2004, doi: 10.1109/MCOM.2003.1267107.
[74] 'Soweto to become first SA township to get fibre – The Citizen'. https://citizen.co.za/news/south-africa/general/2131903/soweto-to-become-first-sa-township-to-get-fibre/ (accessed Dec. 20, 2020).
[75] 'Is Fibre Lighting-Up The Townships? - Business Media MAGS'. https://businessmediamags.co.za/business/made-in-sa/is-fibre-lighting-up-the-townships/ (accessed Dec. 20, 2020).
[76] 'How Fibre is Penetrating South Africa | Fibre | ECN'. https://www.ecn.co.za/how-fibre-is-penetrating-south-africa/ (accessed Dec. 20, 2020).
[77] 'Tales In Tech History: WiMax | Silicon UK Tech News'. https://www.silicon.co.uk/networks/tales-tech-history-wimax-227889 (accessed Nov. 22, 2020).
[78] 'Who won in the WiMAX vs LTE competition?' https://www.computaris.com/wimax-lte-competition/ (accessed Nov. 22, 2020).
[79] 'WiMax versus WiFi'. https://mybroadband.co.za/news/wireless/1748-wimax-versus-wifi.html (accessed Nov. 22, 2020).
[80] M. Daradkah, E. Al Jounidy, and A. Qusef, 'Top-Down vs. Bottom-Up in project management: A practical model', *ACM International Conference Proceeding Series*, 2018, doi: 10.1145/3234698.3234709.
[81] H. Lund and E. Sutinen, 'Contextualised ICT4D: A bottom-up approach', Jan. 2010.
[82] R. Baig, L. Dalmau, R. Roca, L. Navarro, F. Freitag, and A. Sathiaseelan, 'Making community networks economically sustainable: The Guifi.net experience', *GAIA 2016 - Proceedings of the 2016 Global Access to the Internet for All Workshop, Part of SIGCOMM 2016*, pp. 31–36, 2016, doi: 10.1145/2940157.2940163.
[83] 'Shelve "insufficient" SA Connect, develop updated broadband plan | ITWeb'. https://www.itweb.co.za/content/P3gQ2MGx1RKqnRD1?utm_source=dailyEnews_leadLink&utm_medium=email (accessed Dec. 19, 2020).
[84] 'Web Caching - an overview | ScienceDirect Topics'. https://www.sciencedirect.com/topics/computer-science/web-caching (accessed Nov. 13, 2020).
[85] 'squid : Optimising Web Delivery'. http://www.squid-cache.org/ (accessed Nov. 13, 2020).
[86] A. Beamish and B. Arch, 'COMMUNITIES ON-LINE ':, vol. 0, 1995.
[87] 'Alternative Options to Caching Web Traffic | Zscaler'. https://help.zscaler.com/zia/what-are-alternatives-caching-web-traffic (accessed Nov. 15, 2020).
[88] X. Yuan *et al.*, 'Enabling secure and efficient video delivery through encrypted in-network caching', *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 8, pp. 2077–2090, 2016, doi: 10.1109/JSAC.2016.2577301.

[89] D. Naylor *et al.*, 'The cost of the "s" in HTTPS', *CoNEXT 2014 - Proceedings of the 2014 Conference on Emerging Networking Experiments and Technologies*, pp. 133–139, 2014, doi: 10.1145/2674005.2674991.
[90] Daniel Roethlisberger, 'SSLsplit - transparent SSL/TLS interception (SSLsplit)', 2019. https://www.roe.ch/SSLsplit (accessed Dec. 27, 2020).
[91] Diladele, 'Squid For Ubuntu '. https://github.com/diladele/squid-ubuntu/commits/master (accessed Mar. 13, 2021).
[92] 'What is Cache Warming? | Section'. https://www.section.io/blog/what-is-cache-warming/ (accessed Mar. 13, 2021).
[93] 'What is Cache Hit Ratio? - Articles for Developers Building High Performance Systems'. https://blog.stackpath.com/glossary-cache-hit-ratio/ (accessed Feb. 28, 2021).
[94] 'Cache Hit Rate - an overview | ScienceDirect Topics'. https://www.sciencedirect.com/topics/computer-science/cache-hit-rate (accessed Mar. 13, 2021).
[95] 'What is a Cache Hit Ratio? | Cloudflare UK'. https://www.cloudflare.com/en-gb/learning/cdn/what-is-a-cache-hit-ratio/ (accessed Mar. 13, 2021).
[96] M. Butkiewicz, H. V Madhyastha, and V. Sekar, *Understanding Website Complexity: Measurements, Metrics, and Implications*. 2011.
[97] A. K. Singh and V. Potdar, 'Blocking online advertising - A state of the art', 2009. doi: 10.1109/ICIT.2009.4939739.
[98] 'Enders Analysis ad blocker study finds ads take up 79% of mobile data transfer'. https://www.businessinsider.com/enders-analysis-ad-blocker-study-finds-ads-take-up-79-of-mobile-data-transfer-2016-3?IR=T (accessed Mar. 14, 2021).
[99] 'Autoplay Videos Are Not Going Away. Here's How to Fight Them. - The New York Times'. https://www.nytimes.com/2018/08/01/technology/personaltech/autoplay-video-fight-them.html (accessed Mar. 14, 2021).
[100] 'Engaging with people who use ad blockers - Think with Google'. https://www.thinkwithgoogle.com/marketing-strategies/monetization-strategies/adblock-report/ (accessed Mar. 14, 2021).
[101] 'Pi-hole – Network-wide protection'. https://pi-hole.net/ (accessed Dec. 27, 2020).
[102] 'The Rise of Ad Blockers: Should Advertisers Be Panicking?(!!)'. https://www.wordstream.com/blog/ws/2015/10/02/ad-blockers (accessed Mar. 14, 2021).
[103] M. Butkiewicz, H. V. Madhyastha, and V. Sekar, *Understanding Website Complexity: Measurements, Metrics, and Implications*. 2011. [Online]. Available: http://www.cs.ucr.edu/~harsha/web_complexity/.
[104] A. Saverimoutou, B. Mathieu, and S. Vaton, 'A 6-month analysis of factors impacting web browsing quality for QoE prediction', *Computer Networks*, vol. 164, p. 106905, Dec. 2019, doi: 10.1016/j.comnet.2019.106905.
[105] W. L. in R.-B. U. Experience, 'Nielsen's Law of Internet Bandwidth', *Nielsen Norman Group*. https://www.nngroup.com/articles/law-of-bandwidth/ (accessed May 23, 2021).
[106] S. B. Matija Varga, Michal Mokrys, 'The evolution of web browser architecture', no. June, 2014.
[107] 'Inside look at modern web browser (part 1) | Web | Google Developers'. https://developers.google.com/web/updates/2018/09/inside-browser-part1 (accessed Nov. 15, 2020).
[108] 'An overview of HTTP - HTTP | MDN'. https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview (accessed Nov. 15, 2020).
[109] A. Brylinski, 'Overview of HTTP / 2', 2017.

[110] ‘How the browser renders a web page? — DOM, CSSOM, and Rendering | by Uday Hiwarale | JsPoint | Medium’. https://medium.com/jspoint/how-the-browser-renders-a-web-page-dom-cssom-and-rendering-df10531c9969 (accessed Nov. 15, 2020).
[111] P. Mehta, ‘Architecture Overview’, in *Creating Google Chrome Extensions*, P. Mehta, Ed. Berkeley, CA: Apress, 2016, pp. 35–77. doi: 10.1007/978-1-4842-1775-7_2.
[112] A. Barth, A. P. Felt, P. Saxena, and A. Boodman, ‘Protecting Browsers from Extension Vulnerabilities’, *Ndss*, vol. 147, pp. 1315–1329, 2010, doi: 10.1111/j.1365-2486.2006.01169.x.
[113] C. Google, ‘Getting Started Tutorial’. https://developer.chrome.com/extensions/getstarted (accessed Apr. 11, 2020).
[114] P. Mehta, ‘Introduction to Google Chrome Extensions’, in *Creating Google Chrome Extensions*, P. Mehta, Ed. Berkeley, CA: Apress, 2016, pp. 1–33. doi: 10.1007/978-1-4842-1775-7_1.
[115] ‘Chrome Extension Tutorial: How to Pass Messages from a Page’s Context’, Feb. 24, 2021. https://www.freecodecamp.org/news/chrome-extension-message-passing-essentials/ (accessed Jun. 06, 2021).