

https://dspace.org/

A feasibility study for a voucher management system (VOMS) as part of intelligent network (IN) architecture for telecommunications operations in a changing customer service market

Collender, Jonathan Ffyner

2016
http://hdl.handle.net/10539/20633

Downloaded from a DSpace repository. https://dspace.org/

CHAPTER 2

VOMS FACTORS RENDERING HIGHER AVAILABILITY AND THE IMPACT OF AVAILABILITY ON THE CUSTOMER BASE

2.1 INTRODUCTION

The choice of hardware components, high availability concepts and the architectural design would constitute a system with the ability to provide a non-interruptible service offering. This chapter contains a discussion of the available literature on the research topic, and the factors that require VOMS to have a service that is constantly available are examined followed by high availability concepts that would be employed to prove the high availability of the new VOMS architecture.

Voucher Management Systems have become an important component in telecommunications networks for their PrePaid offerings. PrePaid is a method to allow all subscribers access to mobile services without a contract. Le Cadre *et al.* [10] discuss how mobile operators for a long time have regulated the mobile tariffs and the type of service offerings they have available for their subscribers. This is changing, as the telecommunications markets have become saturated and mobile operators strongly have begun to compete for market share [10].

Furthermore, Kim *et al.* [11] maintains that the demand for cheaper tariffs and offerings has become the norm and so has the need for an always-on service. Mobile operators need to ensure that the foundations of their service offerings are built upon systems that are available at all times. If these services were not reliable, customers would become disgruntled and not content with the service they would be receiving, and this could possibly lead to churn for a mobile operator. It has become important to protect the brand [11].

PrePaid services need to be available at all times, as it is vital to the business of mobile operators. VOMS, a component of PrePaid, is no different - this system has to be available to provide its services to its subscribers uninterruptedly and at all times.
Figure 2.1 captures the main elements of this chapter schematically.

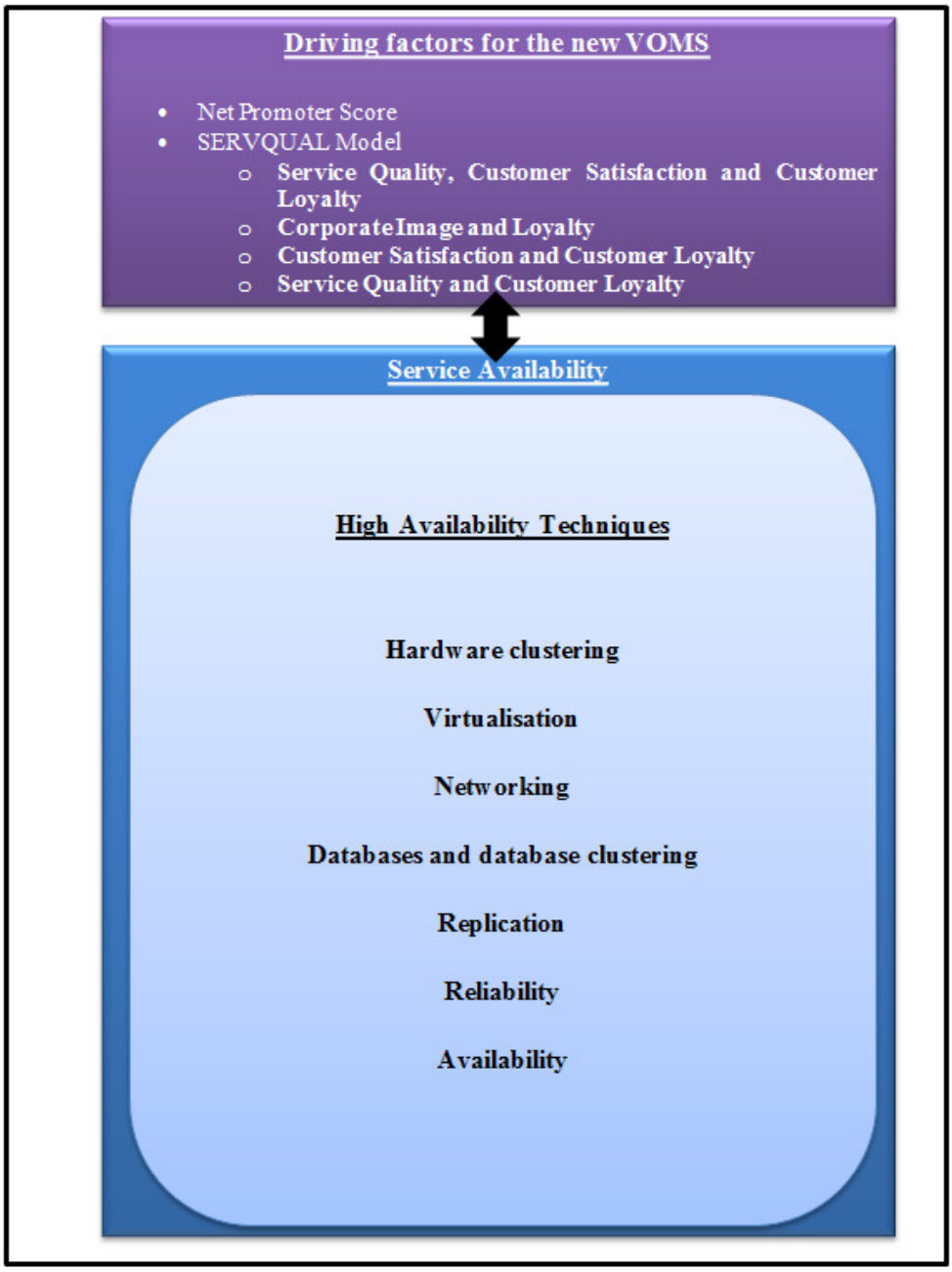


FIGURE 2.1: A diagrammatic overview of the different aspects that will be discussed

In this chapter two main factors will be identified, which constitute the driving forces for a re-evaluation of VOMS, and which are the reasons why VOMS should be designed with a higher availability, namely

1) Customers' loyalty and their demand for services having higher availability
    - The net promoter score, and
    - The SERVQUAL model.

2) The PrePaid offering
    - Components and techniques that promote higher service availability to enhance the PrePaid service offering.

2.2 PREPAID OFFERING

A prepaid system allows consumers to be more in control of their accounts than a contract does. The subscriber only pays for the minutes that are used and are not tied to a lengthy contract. Prepaid subscribers are not credit vetted and consumers who may have a tarnished credit history will not be blocked from owning and using a mobile phone. With several mobile operators in the market today, consumers have options to choose the best pricing plans amongst them.

Prepaid services as discussed by Scanlon [12], penetrated emerging markets, and as from September 2006, those services accounted for 62.5% of the global net growth. This is equivalent to around 1.62 billion customers who are all using prepaid as opposed to contract. In Figure 2.2 the graph indicates the percentage of subscribers making use of prepaid services. The prepaid system is lucrative in developing markets, specifically because of its low denominations for recharging. Further approaches also have been made to reach niche customer sectors, for example the popular prepaid offering according to which the subscribers are contracted for up to a specific amount, and when the amount is depleted, they switch over and replenish their contract by making use of a prepaid system. These customers are called hybrid subscribers, and the contract is known as a PrePaid Contract Hybrid (PCH). This boils down to being a contract subscriber with the benefit of

a prepaid usage. Customers making use of this system have control over their accounts and cannot incur large bills at the end of a month [12].

Figure 2.2 depicts a graph of the percentage of prepaid subscriptions.

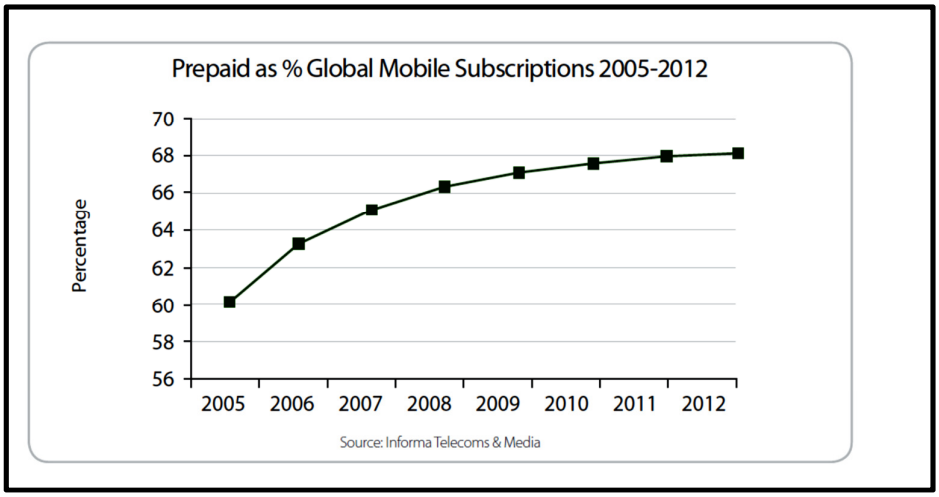


FIGURE 2.2: Graph indicating prepaid customers as a percentage of global mobile subscriptions 2005 – 2012, taken from [14]

In a blueprint of Hewlett-Packard (HP) [13] the popularity of Prepaid in many segments of the market is discussed. Especially in segments where family incomes are low this is the preferred way of staying connected. Prepaid is particularly popular amongst children, as there is no risk of high bills being incurred. Offering credit to customers always carries with it a credit risk; however, this financial risk can be mitigated by making use of the prepaid billing model. Prepaid also is a preferred method for data, voice and video services.

The servers that form part of the services that are used for prepaid methods of payment, however, need to be reliable and have a high availability. Subscribers to the prepaid services have expectations of being able to perform balance enquiries at any time; this emphasises the importance of a reliable system that is guaranteed to be available at all times. Prepaid is a viable market and it shows constant growth, with many new subscribers joining every day. Competition among mobile providers has created a challenging market and mobile operators are compelled to change their strategies regularly to constantly adapt to provide their subscribers with the most suitable products

at the appropriate time. In the light of this, mobile operators cannot do without a system architecture that supports rapid deployment of new offerings to remain competitive [13].

According to Grzybowski [14], competition is rife, and certain mobile operators have started giving away certain of their services at a significantly reduced cost - their reasoning being to gain a greater market share. Mobile operators need to meet the demands of their customers, and as their customer base is evolving, demands are growing. To remain profitable, operators need to work smarter and become more efficient than ever before when delivering value-adding services [14].

According to Insulander [15], revenue assurance is vital in that mobile operators need to make sure that the offerings that are provided to their subscribers are validated, and to make certain that subscribers are not receiving more than they should, or, more importantly, that they indeed get what they pay for. Revenue assurance is an important ramification of a mobile operator's functions and PrePaid systems that are responsible for providing this function therefore are required to be resilient. Mobile operators are to be cautious, as system errors could cost them money due to revenue leaks and fraud. Studies suggest that it is important to set up proper revenue assurance teams that have the ability to perform checks to circumvent this. The loss of data to perform an analysis would be detrimental to a mobile operator and its subscribers. Having a resilient PrePaid system, greatly benefits the revenue assurance of a mobile operator [15].

2.3 DRIVING FACTORS FOR A NEW VOMS ARCHITECTURE

As described by Nguyen and Leblanc [16], a strong customer base is crucial to any mobile operator. How do mobile operators know whether their subscriber base is satisfied with the services they are rendering? A model was developed to determine this, called the Net Promoter Score [16].

2.3.1 The Net Promoter Score

Grisaffe [17] indicates that the Net Promoter Score (NPS) has been developed as a means of defining consumer happiness. The NPS is described as having values as low as -100 and as high as +100, meaning that the negative value denotes that all the scores are detractors and positive values indicate all are promoters. A value that is positive is seen to

be “good” and any value above 50 is considered to be excellent. There are claims of a correlation between the NPS and revenue growth [17].

2.3.2 The SERVQAUL model

The SERVQUAL model as described by Nguyen and Leblanc [16] suggests that a customer’s expectation about the performance of a mobile network and the assessment of the actual performance thereof result in the perceptions of quality. Nguyen and Leblanc [16] maintain that corporate image is affected directly by service quality and customer loyalty. The leading predictors of loyalty from the side of customers have been found to be both customer satisfaction and service quality [16].

2.3.3 Customer loyalty

Abd-El-Salam and Shawky [18] define customer loyalty as the result of customer satisfaction and the trust of customers. Customer loyalty was measured in a study undertaken by an international mobile operator. The results of the study showed that service availability had a direct effect on customer loyalty, which had an effect on consumer satisfaction and trust [18].

2.3.4 Customer satisfaction and customer loyalty

The close link among service quality, consumer satisfaction and customer loyalty has been clearly established in other studies too. A study conducted by Ali *et al.* [19], for example, found that consumer satisfaction was a natural consequence of service quality, and from that consumer loyalty ensued. The results of this study also indicated that repeated consumer purchases were a clear indicator of consumer loyalty [19].

2.3.5 Corporate image and loyalty

Mohammadoghli *et al.* [20] described a correlation that was found between the good reputation of a company and the number of customers they attracted, and the motivation of the company’s employees. The perceived brand of a company was found to be important, as a correlation existed between the brand of the company and corporate reputation [20].

2.3.6 Summary of the driving factors

The findings of various studies undertaken provided ample evidence that poor service offerings would impact negatively on customer retention. Therefore, it may be inferred that improving on service availability and quality would be important to the mobile operator business environment. If mobile operators do not take these factors into consideration, their customer base might be affected adversely, which would lead to possible churn. As VOMS is a component within a PrePaid system, if this portion of the PrePaid system were to be unreliable, it would impact a customer's ability to recharge and purchase services. In the following section (2.4) attention will be paid to the attributes that an existing VOMS should have and additional areas of the architecture that might need to be investigated to improve the availability. The cost of the new VOMS architecture would need to be considered and the choice of high availability structures should be evaluated and considered accordingly.

2.4 SERVICE AVAILABILITY

Taking due cognisance of service availability was important in this study, as it was required to determine the components required for designing a higher availability in order to improve an existing VOMS system.

As described by Calzolari [21], the term availability refers to the likelihood that a system is functioning correctly at a specific point in time. The ability of a system to persist during a failure of one or more of its system components will result in it being called fault tolerant. A failure occurs when a service has failed and does not service the subscriber base, in other words, the service is no longer functioning the way it has been intended to according to specifications outlined. The purpose of a high availability system is to provide services to the client base that is connected to it, despite errors or failures. Redundancy has been introduced as a possible solution to this problem. Service availability stemmed from the concept of a high availability system. In the unlikely event of a failure, the services needed to be re-established within a very short period of time, or else, not fail at all [21].

Having stated that, the next step would be to examine the various techniques that are required to achieve high availability for systems in which high service availability is crucial. In examining these techniques, of them all would have to be made use of in the new VOMS system. To create a highly available system, a choice of good quality redundant hardware would form the foundation of the new VOMS system. Although it would be important to use good hardware, this alone would be no guarantee for service availability. The good architecture of the overall system could significantly improve the service availability and thus a careful understanding of high availability concepts would be required.

2.4.1 Hardware clustering

According to Pfister [22], cluster computer could be defined as connecting multiple computer systems together to form a single logical computer server. Cluster computing is not a new technique and has been around for many years. A cluster computer provides an added benefit of parallel processing. Parallel processing implies that more computational power will be available, and thus the added hardware may be used to process requests that much faster. It is worth noting the distinction between fault tolerant systems and clusters. Fault tolerant systems comprise independent, redundant subsystems and clustered computers are built upon independent, replicated subsystems. In fault-tolerant systems, should there be a failure of the primary system, the secondary would take over, although, under normal operating conditions the secondary system would not assist the primary server. The behaviour of the cluster computer is different in that it shares the work load all the time. This is the fundamental difference between a fault-tolerant or redundant system and the clustered computer architecture, also known as the replicated subsystem [22].

According to Bakery and Buyya [23], clusters can be built with low-cost work stations to emulate and compete with expensive high-end parallel proprietary servers. Clusters would require the following:

- Many high-performance systems
- An operating system that is layered

- Fast Network Interface Cards
- Fast Network Switches
- The cluster middleware layer.

The main benefits of using a cluster computer system are:

- Improved performance
- Easily scalable
- Significant throughput
- Achieving high availability.

According to Bakery and Buyya [23], clusters provide an alternative to mainframes and high performance end servers, which come at great cost. The operating systems that can be used to create these cluster nodes are: Linux, HP-UX and Solaris, to name a few. Bakery and Buyya [23] contend that the complexity of setting up a cluster was noted as fair and not overly complex [23]. Figure 2.3 illustrates the structure of cluster computing. All the computers were linked together to form the cluster.

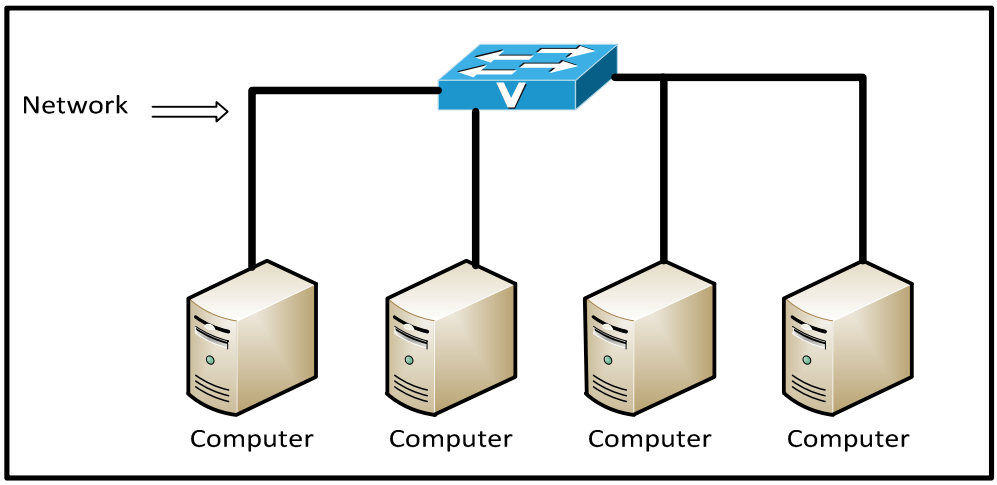


FIGURE 2.3: Cluster Computing [23]

While cluster computing is more reliable than traditional fault-tolerant systems, according to Manopulo [24], the evolution of cluster computing is moving to grid computing. The full potential of clustering over many sites is the key to grid computing. It needs to be noted that the full potential of grid computing can only be realised through high performance networks. If there are any issues and latency creeps in, it will cause serious performance issues. The main strength behind grid computing is the significant parallel processing that it offers [24].

The existing VOMS system has been built on fault-tolerant hardware. This principle of computer clustering is important as the new design will focus strongly on this technique to increase the reliability. Cost is an important factor for the new design, as it needs to be cost effective to implement the new VOMS. The principles of clustering work in favour of the new VOMS design, as the existing VOMS system uses fault-tolerant hardware (such as HP Non-Stops), and is more expensive than commodity hardware that is used to build the cluster.

2.4.2 Virtualisation

Various forms of virtualisation and implementations thereof exist. According to Rose [25], there are two distinct virtualisation techniques, namely full system virtualisation and Paravirtualisation. The history behind virtualisation began when corporations used large computer systems to process their application requirements. It soon became too expensive to allow such a large system to be used by a single application or user, which made way to time sharing. Time sharing allowed applications and users to use a larger computer system simultaneously. Computers were becoming ubiquitous and time sharing had its problems. The problem of using a large single computer system which hosted various concurrent applications and users was that when any issues occurred with the system, all applications and users would be impacted. Those companies that could afford to purchase multiple smaller computer systems managed to combat this issue. This setup of using smaller computers worked as there was both a performance increase and it created isolation. Such a method of multiple computers would mitigate the effects downtime had on the single large system; however, it became evident that it was not feasible to maintain so many individual systems. The progress made in terms of virtualisation led to the implementation of full virtualisation and Paravirtualisation. Full system virtualisation worked as follows: it was designed in such a way that an operating system and its

application could be executed on the virtual hardware similarly as if it were run on the main system hardware. While full system virtualisation was developed for main frame systems, Paravirtualisation was created for PC architectures. Although it was possible to add full virtualisation to PC architectures, it was noted that there was poor performance when doing so [25].

Figueiredo and Dinda [26] highlighted the importance of the Virtual Machine Manager (VMM) layer in virtualisation. The full implementation of the future development of virtualisation was dependent on the evolution of the VMM. This would result in better scalability and performance increases [26].

Menascé [27] noted that when applications were run on a virtual machine, there would be a phenomenon known as virtual machine slow down. This would occur if many virtual machines would be added to a single physical machine. As these many machines contended with the CPU resource on the physical machine, it was noted that the response times would deteriorate accordingly. This scenario was for a physical system with a single CPU. The following formula was used to calculate this:

$S_V = f_p \times N_e + \left(1 - f_p\right)$ [28].

Where f_p was the "fraction of privileged instructions executed by the VM" and N_e the "average number of instructions required by the VMM to emulate a privileged instruction", the result is shown in Figure 2.4 [27].

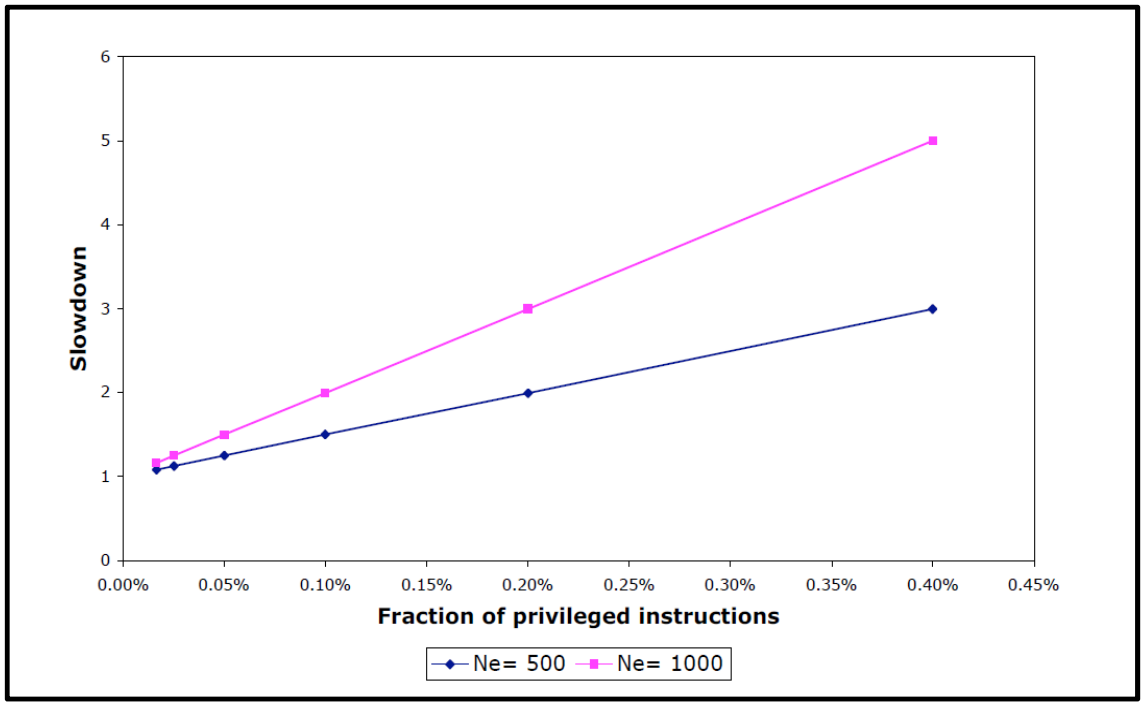


FIGURE 2.4: Execution Slowdown, taken from [27]

With modern computers, the systems have many CPUs with multiple cores. The problem described by Menascé [27] would be mitigated by the sheer number of CPUs on modern day systems.

The new VOMS system will require virtualisation for the benefit of performance independence. If the applications would be dedicated to separate VMs, the benefit would be two-fold, first, each application would have its own resources, and second, in the unlikely event that an application caused any problems, it would only impact on itself and would have no impact on any neighbouring VMs and their applications.

The requirement for virtualisation would be to enhance the overall service availability of the existing VOMS, thus contributing to a much higher availability. Rose [25] proclaimed that there was a significant cost saving by the use of virtualisation; this would make the new VOMS more economically feasible to develop and implement.

It is worth noting that, according to Scarfone [28], there were two implementations of Full Virtualisation, namely Bare Metal and Hosted. Figure 2.5 depicted a Bare Metal, Full

Virtualisation system. The main difference was that the Bare Metal virtualisation method had no base OS. The Virtual Machine existed directly on the hardware. This was a suggested approach for servers. In Figure 2.6 the Hosted Virtualisation is depicted. This latter virtualisation technique better suited desktop systems. Furthermore, a concern for security would need to be taken into consideration. Bare Metal Virtualisation was inherently better suited and was more secure than a Hosted Virtualisation system. There were drawbacks to deciding on Bare Metal Virtualisation in that it could only be installed on a limited amount of hardware. Hardware components such as Ethernet controllers and graphic cards were of specific concern for Bare Metal Virtualisation. On the contrary, there were very few limitations with regard to specific hardware for the Hosted Virtualisation system [28].

Based on this discussion of the literature, the new VOMS would make use of full virtualisation and implement a Bare Metal solution.

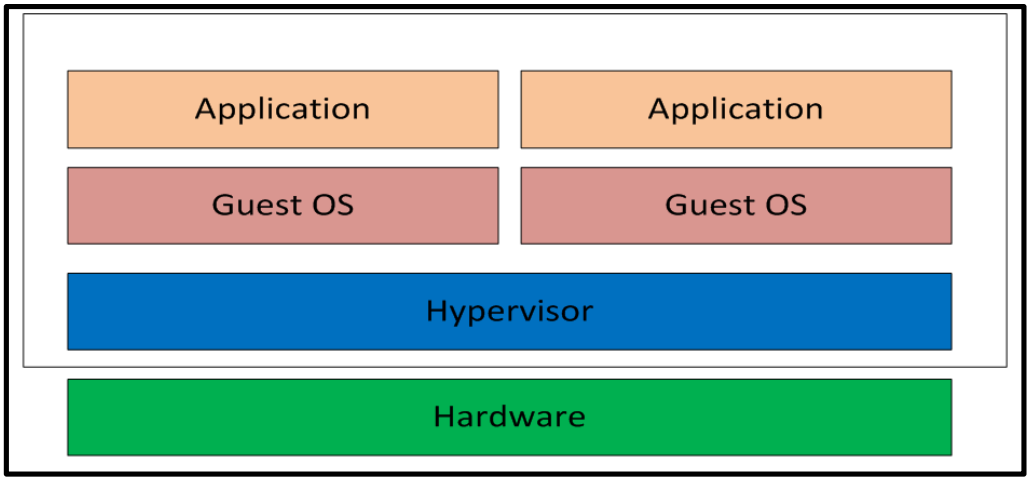


FIGURE 2.5: Full Virtualisation, Bare Metal [28]

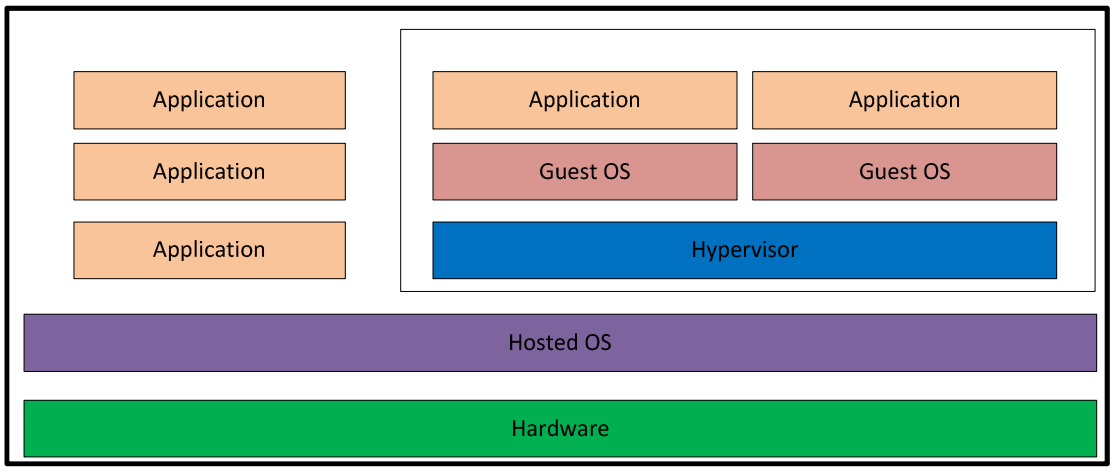


FIGURE 2.6: Full Virtualisation, Hosted [28]

2.4.3 Networking

Networking was a crucial part of the overall design. A network would provide connectivity, and would connect users to their application servers. Additionally, if there was a requirement for high availability, the architecture of these networking components would be crucial. The scope and focus for this research project with regard to networking would be centred on the construction of a network topology, which would support the high availability systems.

According to Ranjbar and Hutton [29], there were various factors that would need to be considered to design a high availability network. The most prominent of these considerations were the construction of a network in either a series or a parallel. Figure 2.7 shows the basic way in which a serial network is connected, and Figure 2.8 the parallel network. It was noted that availability of the serial networking structure was less than that of the parallel network. As the path between the user and the server for a serial network could be disrupted if any of the links or routers were to fail. For the parallel network, there were redundant paths and routers in the event of a failure. Although the parallel network would offer a better availability, it was not only the design of the network that would contribute to the higher availability, but, more importantly, the protocols that would manage those routers. These protocols were Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Enhanced Interior Gateway Routing Protocol (EIGRP). The routing protocols would determine the course of action taken in the event of a loss of a link or the loss of a router. These protocols were

specifically compiled to keep track of the available routes. The most basic of these routing protocols was RIP, while the most extensively used protocol was OSPF. EIGRP was a CISCO proprietary protocol [29].

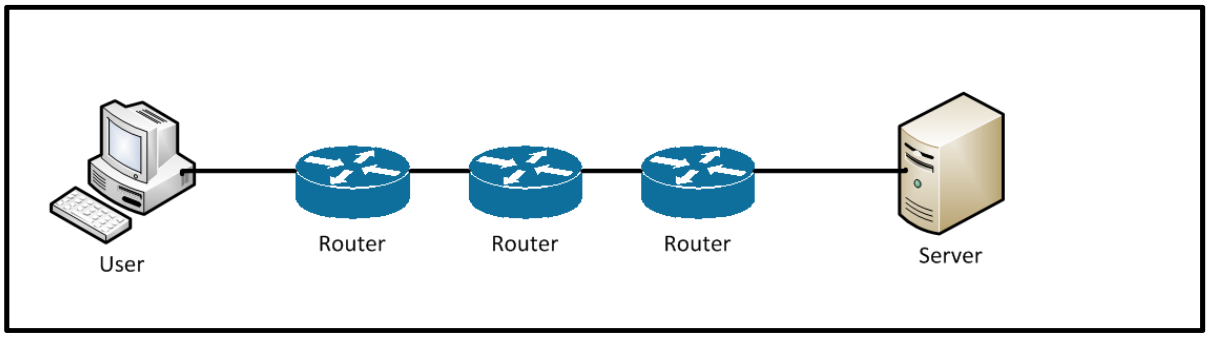


FIGURE 2.7: Serial Networking [29]

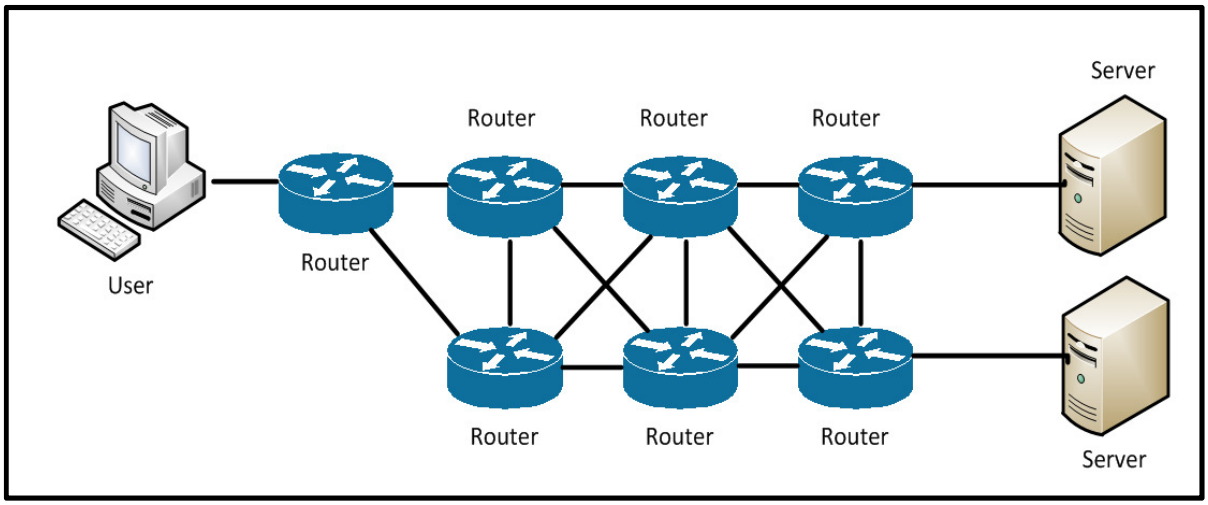


FIGURE 2.8: Parallel Networking [29]

While high availability routing was important as it assured that traffic would arrive at the required destination, another important protocol was Hot Standby Router Protocol (HSRP). It was another high availability protocol that Cisco had created. HSRP would link two routers together to form one virtual router. In this configuration, only one router would take traffic and become the primary router, thus leaving the second router as the backup. The secondary router's ports would be linked with the primary router's ports, thus creating a form of port expansion. This would then be exposed to the network as only one router. The physical connectivity can be seen in Figure 2.9. The downstream servers were connected to both routers with crisscross links. The strength of the design

was the high availability that was provided by HSRP. The HSRP protocol made use of hello packets to allow the two routers to keep in contact with each other. In the event of any failure on the primary router, the secondary router would know that it would need to take over the routing. Thus a user connecting from the network, through to the server they needed to access would always have access to servers. How long the secondary router would need to wait before it took over was configurable [30].

The HSRP protocol and the setup in Figure 2.9 will be used to construct the new VOMS system. This is another networking technique to ensure further resilience for the new design.

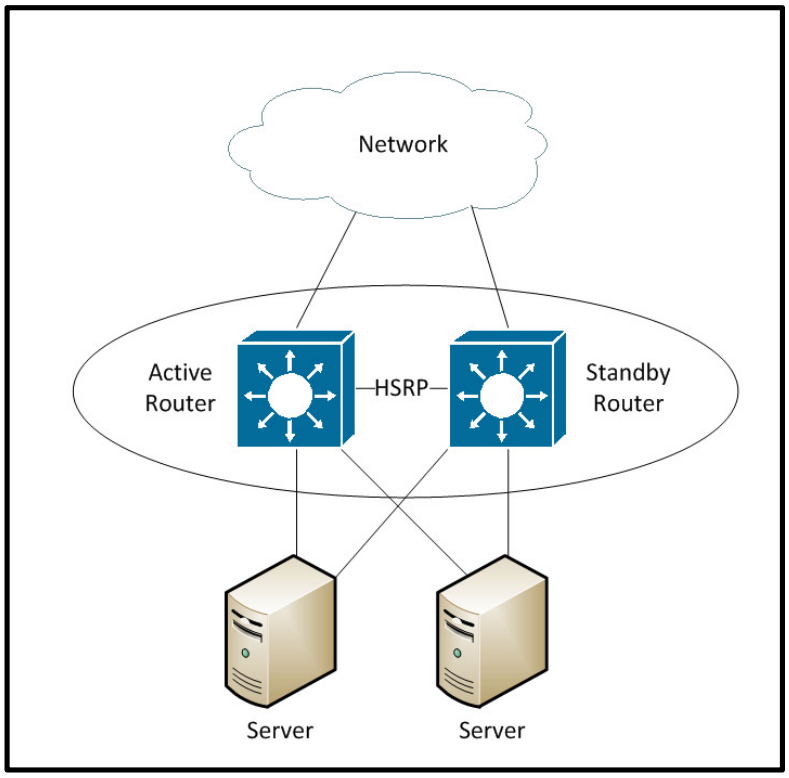


FIGURE 2.9: A network using Hot Standby Routing Protocol [31]

2.4.4 Load distribution

According to Kopparapu [31] there are two main types of load balancing, network load balancing and server load balancing. Network load balancing can disperse the load of network traffic over various paths, while server load balancing will divide the load over a multitude of front-end application servers. Load balancers have evolved over time and

became more complex with regard to their functionality. These functions included health checks that could be performed from a router to a server to determine if the server was available to process requests. This was done mainly to increase the reliability of the overall architecture. Additional benefits of load balancers were that they provided security by protecting the back-end servers as the load balancers were deployed on the front-end server farms. Access lists could be added to the router or multi-layer switch and thus would become a firewall adding additional front-end security to the application servers. Load balancers had the ability to provide intelligent routing, as they were able to inspect Transmission Control Protocol (TCP) packets and direct network traffic accordingly [31].

In Figure 2.10, the concept of load balancing is represented. There were various methods to implement load balancing. A multi-layer switch could be configured with many rules for directing traffic to front-end servers.

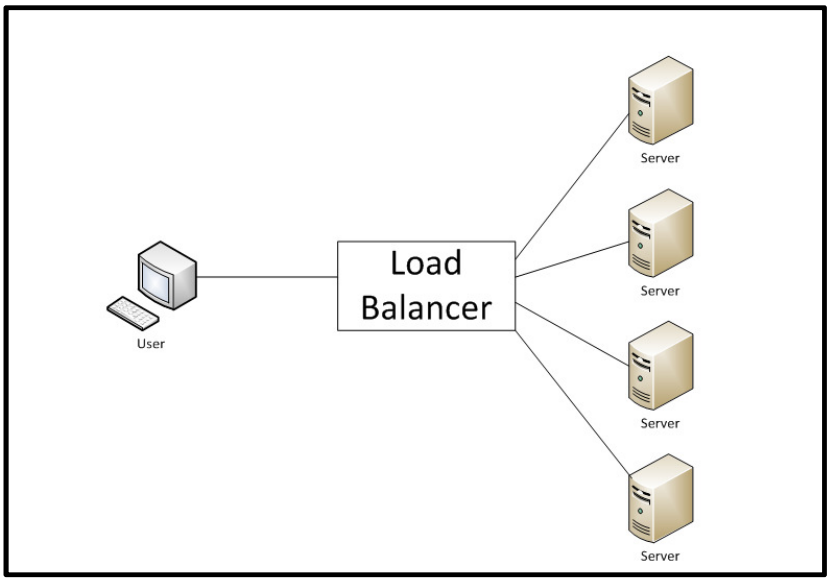


FIGURE 2.10: An example of a load balancer configuration [32]

Bourke [32] described load balancing as the distribution of traffic to several servers within a network and explained that the process was transparent to the client who originated the traffic. The main role of load balancing originated from the internet. It was a method to provide better performance to busy web sites. As single front-end servers would become overwhelmed with traffic, they needed to start load sharing traffic amongst many front-end application servers. Load balancers offered various benefits, two of which

were redundancy and scalability. Other driving factors for the use of load balancers were that application down time would lead to financial losses, especially for e-commerce web sites [32].

Cisco had a specific device which was named an Application Control Engine (ACE) module. This module could be configured to provide load balancing, with specific routing rules. In addition to the rules for routing, it had the ability to act as a firewall and provide additional security for the front-end and back-end servers. This was truly an example of an intelligent switch.

According to Cisco Systems [33], their ACE modules were designed to add value to networks by creating high availability networks, securing data centres and consolidating data centres. They achieved this through their intelligent OSI layer four and layer seven technologies. A significant advantage was the host health checks that could be done on the application servers. The ACE module could interrogate various application ports to determine if the applications were available for each server that it had a connection to. If a specific server's application were to become unavailable the ACE had the ability to remove that server from the service pool and redirect application requests to other available servers. This form of intelligent switching provided significant value, as this failover or auto switching would take place whenever there was a problem [33]. Operators or systems administrators could rely on the equipment to make such decisions when the problems presented.

The new VOMS design will need to make use of a multi-layered switch, which would act as the front end. Configuration would need to be created to allow the connectivity to the front-end application servers. Based on work done by Kopparapu [31] and Bourke [32], the load balancing technique would be employed to increase the availability of the new design. The recommendation of adding multiple front-end servers to mitigate the loss of an application server would provide a higher service uptime. While the new design would require more application front-end servers, these servers would be created as Virtual Machines. This would cut the cost of implementation and thus make the new VOMS more economically feasible to build.

2.4.5 Databases and database clustering

The choice of database would be an important decision. There are various databases on the market today. Some are very costly to use, especially when they are to be run commercially. The choice of database for the new VOMS system would be determined based on cost and reliability. Examples of databases that were available are: Oracle, Oracle's MySQL, MariaDB, MS SQL, PostgreSQL, Firebird SQL and Apache Derby, to name a few. Many other databases were available on the market, though only a few were suitable for carrier grade applications. Based on these criteria, along with complexity of installation and maintenance, Oracle's MySQL would be chosen as the best suited database tool.

Oracle's MySQL could be enhanced by creating what is known as a database cluster. Though similar in concept to the application cluster, the clustered database would provide higher availability than the standard installation. Figure 2.11 shows the architecture of Oracle's MySQL cluster.

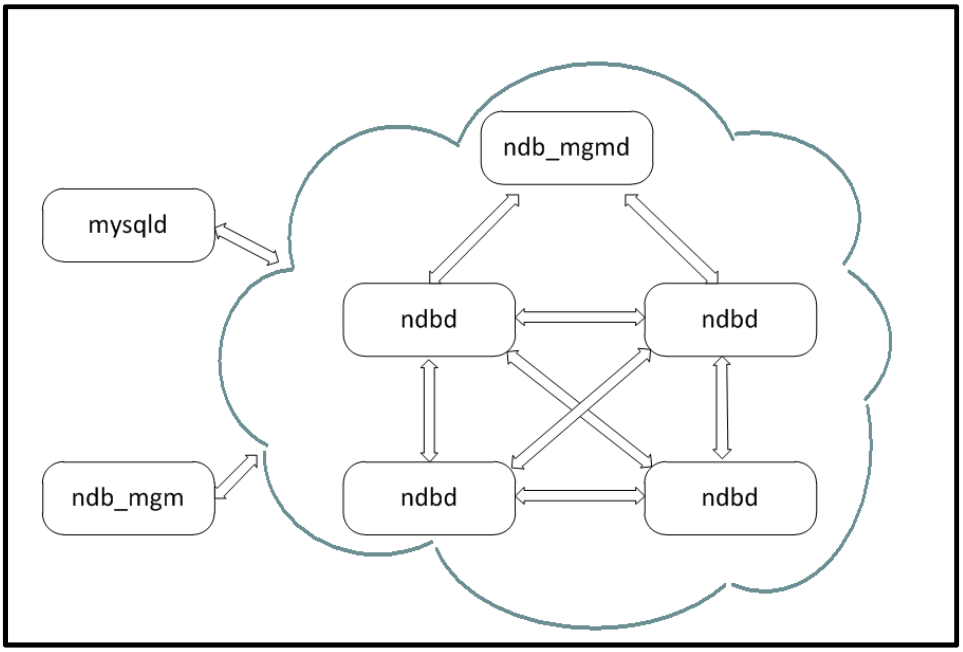


FIGURE 2.11: MySQL Cluster components [34]

According Bell *et al.* [34], MySQL clustering not only provided a high availability, but additionally scalable writing operations and low latency. The strength behind the cluster was the active/active system and that the database could be implemented as a multi-

master system which led to low replication lag. The components of the MySQL cluster, as can be seen in Figure 2.11, were defined as follows: *mysqld* was the MySQL server node, the *ndb_mgm* was the cluster's management client, the *ndbd* was the MySQL data node and the *ndb_mgmd* was the MySQL cluster management server. Applications that were designed for MySQL could be moved to the MySQL cluster without making any changes. The high-availability nature of the MySQL cluster entailed that data were replicated amongst the data nodes, so that in the event of the failure of a node, the data still would be available. The ndb_mgmd was responsible to manage the data nodes. The MySQL cluster was noted as a shared-nothing storage solution [34]. It meant that all the components performed independently, thus making the MySQL Cluster a very good high-availability choice.

Oracle's MySQL Cluster will be used as the database framework for the new VOMS design. Oracle's MySQL provided excellent data integrity and fast replication between nodes. To further enhance the database reliability, an ACE load balancer would be used in the design to further increase reliability. The Cisco ACE module would additionally interrogate the cluster nodes with health probes to determine their availability.

2.4.6 Replication

According to Moiz *et al.* [35], replication was an important function to protect data in the event of failures. Replication was important to create duplicate sets of data on alternate systems, to provide applications an alternate system to access during a failure event. By providing more database access options for applications, architectures with multiple database systems would increase the performance of the applications and provide a higher availability. There were two forms of replication, synchronous and asynchronous. Synchronous replication was used for systems that required high availability, whereas systems where data replication was not that important could use asynchronous replication. Synchronous replication was better known as a Master – Slave system while asynchronous was known as store-and-forward replication. Furthermore, there were three replication strategies that could be made use of, which are described as follows [35].

Snapshot: Which was considered the basic form or replication, where the entire database would be copied to another server's database.

Transactional: For this form or replication, only those transactions that took place would be replicated.

Merge: For systems that were not always connected to each other, they were able to transact independently and would be synchronised at certain periods. Only the data that changed on either system would be replicated across. Where conflicts arose, a predefined strategy was in place.

2.4.7 Reliability

According to Rausand and Høyland [36] the definition for reliability is: "*The ability of an item to perform a required function, under given environmental and operational conditions and for a stated period of time*" [36].

According to Nakagawa [37], the reliability theory had grown from defects during the Second World War from which valuable experiences were gained. With recent technology, innovation had greatly increased and the importance of reliability was required for creating good products which were reliable. The reliability theory was initially applied to the industrial, electrical and mechanical sectors and other sectors too had benefited - those other engineering sectors included computer and communication systems. It was possible to analyse complex real systems, both statistically and stochastically, which would add to improve the reliability of that system. While redundancy would be an important component in any real system, maintenance of the real system would surpass redundancy. There was a link between system reliability and the reliability of an individual component within the system. The reliability of the overall system could be improved by constant maintenance. Therefore, to achieve and maintain high reliability, the following three points need to be taken into account [37]:

- To prevent failures to occur, the units need to be properly serviced at regular intervals
- Any unit or system that has failed, needs to be replaced immediately
- Redundant systems or units need to be provisioned.

2.4.8 Availability

More than a decade ago, Marcus and Stern [38] claimed that it was not possible to achieve a system availability of six 9s with the technology of their time and proclaimed it was an unrealistic goal. They made their statement in the year 2000. Then Holenstein *et al.* (2007) [39] stated that not only was it possible to achieve six 9s but to press on to eight 9s and beyond, just a few years later. These high availability values were not specifically for system availability but for application service availability. It was possible to achieve extreme service availability by the specific design of the overall system. The system had to be of an active/active nature, meaning that the services had to run actively on more than one system simultaneously. To achieve the extreme availability times, in the light of components that could fail, the key was to fix the faults as quickly as possible [39]. Table 2.1 indicates the possible down time per year for the corresponding 9s.

Table 2.1: Down time per year for the corresponding 9s [39]

Number of 9s	Down Time per Year
99.999% (five 9s)	5.26 minutes
99.9999% (six 9s)	31.5 seconds
99.99999% (seven 9s)	3.15 seconds
99.999999% (eight 9s)	315.569 milliseconds
99.9999999% (nine 9s)	31.557 milliseconds
99.99999999% (ten 9s)	3.156 milliseconds

The Mean Time between Failures (MTBF) and Mean Time to Repair (MTR) were greatly responsible for defining a system's reliability. MTR could be defined as the average or the mean time to be repaired before the service could be brought back into service. The MTBF could be defined as the average or the mean time before the failure of the system occurred [37, 39].

Holenstein *et al.* [39] noted the following equations as being of importance:

Availability (A) = $\frac{MTBF}{MTBF+MTR}$ [39]

This could be simplified to:

$A \approx 1 - \frac{MTR}{MTBF}$ [39]

And the failure of the system noted by F, which is the opposite of A.

$F = 1 - A \approx \frac{MTR}{MTBF}$ [39]

If we made MTBF, the subject of the formula, then:

$MTBF \approx \frac{MTR}{1-A}$ [39]

Furthermore, Holenstein *et al.* [39] described the configuration required to achieve eight 9's as a system that required dual redundancy by using commodity hardware. By making use of HP's Non Stop servers, with dual redundancy, it was possible to achieve ten 9's. These feats were achieved with the assumption of an MTR of four hours, by making use of an active–active system and having 150% capacity. They concluded, therefore, that it was possible to use individual systems with availabilities of four 9's, and, with the correct system architecture, produce extremely high service application availability.

The existing VOMS make use of hardware which guarantees five 9's. This has no bearing on the application service availability which was discussed in Chapter 1. The new VOMS system would be an active-active system, making use of clustered hardware with virtualised application servers and database servers residing on the clustered hardware. According to Holenstein *et al.* [39], it could be possible to achieve an application service availability in excess of eight 9's and above with such an architecture. Eight nines equates to a service down time of less than a second per year. The goal would be to design and construct such architecture and determine if this would be possible. Theoretically all the

building blocks are available to achieve this. The technology at present lends itself to achieve this extreme application service availability.

2.4.9 Summary of service availability

A variety of aspects that contribute to service availability were discussed in detail in this section. It is important to understand the various technologies that are required to establish a highly available solution for mobile operators.

Based on these different technologies and concepts, the design considerations of the architecture of the VOMS need to be evaluated. While the literature survey discussed in this chapter, elucidated various themes, for example, clustering and virtualisation, the idea for the new VOMS architectural design would be to encompass both of these primary technologies to create a highly available system.

The idea followed that the physical servers would be clustered together to produce significant computational power and then to add virtual machine software onto these physical machines. Virtual machines would be created for the application and database servers.

The new VOMS applications would then exist virtually on all the physical machines. In the event of a physical server experiencing downtime, the virtual machine should not be impacted.

2.5 CONCLUSION

This chapter provided a background to the rationale and driving factors for the new VOMS architecture. The chapter also established further context for VOMS and the elements that are required for better service availability.

Through the use of various techniques, as mentioned in 2.4, VOMS architecture would need to be revisited to achieve a high nine availability. This, in turn, would feed back to the service offering that was given to the subscriber base and provide a recharge service that would possibly never be disrupted in order to allow subscribers always to be able to recharge their accounts.

The viability of the new VOMS system is important, and through this literature study, the various techniques that were researched all promoted a lower overall cost. This would support the building of a lower cost new VOMS system.

In the next chapter, Chapter 3, **Research design and methodology**, the research design and the methods applied are discussed.