Student Full Name: **Vinod Mankad Vasudevan**

Student STU Number:

Course Title:

Dissertation Title: **Performance Benchmarking through Machine Learning in Virtual Infrastructure**

Registration Number:

Enrolment date: **July 2013**

Word Count:

MSc Dissertation

Performance Benchmarking through Machine Learning in Virtual Infrastructure

Author:

Supervisor:

Student STU Number**:**

Date:

Declaration and Statements

This dissertation is a product of my own work and is the result of nothing done in collaboration. I consent to RDI's free use including online reproduction, including electronically, and including adaptation for teaching and education activities of any whole or part item of this dissertation.

(Student signature)

(Typed student name)

Acknowledgements

Journey as a Post graduate student is a rewarding yet an arduous one. Even with the utmost sincerity, perseverance, and dedication from my part, this long endeavour would not have been successful without the blessings of my lord and cooperations from my family members, advisor, professors, co-workers, friends, and the sponsors. Graduate research is a symbiotic ecosystem where individual accomplishment is only a part of a collective glory. The dissertation contributions belong as much to me as to these people surrounding me.

Without any iota of exaggeration, this work is only possible because of the active and constructive guidance from my academic advisor, Steve Presland. From my inception into the academic program, his supportive, compassionate, prudent, and sometimes admonishing mentorship has been a constant catalyst in stewarding my career. There have been brief moments of glitch between his expectations and my levels of commitment; but, overall, I feel these interactions have transformed me into a better researcher, and a better academic candidate.

Also, I would like to thank my co-workers and friends in bankmuscat, especially Gopinathan Nair and Sarbudeen. I am so privileged to have you as my friends and lab mates. I would like to specifically mention their expert instructions on setting up lab machines and troubleshooting.

Pradeep Kumar is gratefully acknowledged for having helped proof-reading and fact-checking earlier drafts of this paper.

I would finally like to thank Craig, Lynsey & Joanna for their general support in the dissertation.

Abstract

The present research aims at developing a machine learning algorithm which could predict CPU usage and possible performance degradation in cloud based systems. Machine learning has become one of the crucial topics in computing and serves to be one of the cornerstones of the information technology sector. Machine learning aids to the prediction of future data with the purview of facilitating decision making under the circumstances of uncertainty. Hence, the present research utilised Linear regression machine learning algorithms to predict CPU usage of two CPU intensive applications- MS-Outlook and Oracle HRMS. The present study will be a fundamental model for system administrators to predict performance of cloud systems as with cloud computing, the resource needs are ever increasing and hence, performance monitoring should be established. The study further recommends implementation of the proposed model in cloud based environment and the need to monitor CPU intensive applications deployed in cloud using the model.

Table of Contents

List of Tables

List of Figures

List of Graphs

CHAPTER I: INTRODUCTION

1.1 General Introduction about Machine Learning Algorithms and Complex Systems

Machine learning has become one of the crucial topics in computing and serves to be one of the cornerstones of the information technology sector. However, the functionalities of machine learning are concealed from rest of the computerised world. Machine learning becomes a dire need when complexity and adaptivity are considered. The need for machine learning algorithms arises when certain systems are complex and require the usage of large data sets. The factor of adaptivity also emphasises machine learning wherein programs should adapt and act as per the changes in the input (Example: Speech recognition systems) (Shalev-Shwartz & Ben-David, 2014). Machine learning aids in the identification of predicting future data with the purview of facilitating decision making under the circumstances of uncertainty. Smola and Vishwanathan (2008) claim that the persistent expansion of data had compelled the need for computer aided smart data analysis systems that will further add up to the progress of technology. The context of machine learning in resource allocation and prediction of resource consumption in cloud and grids was studied earlier by Matsunaga and Fortes (2010). The research stated that the development of modern machine learning algorithms had idealised the abilities of computer systems to manage several parameters such as the size and speed of memory and storage, input data characteristics, CPU parameters and so on. In this purview, the impact of machine learning in evaluating the performance metrics of cloud based system is evident.

The advent of virtual computing and cloud based services has changed the potentials of the information technology (IT) and IT-based businesses. Scalability and flexibility are the two main factors that make cloud computing so popular. These factors in addition pave way for reliability in the services offered to the end users (Ibrahim *et al*., 2010). Technology has impacted the traditional ways of computing wherein every business firm may possess dedicated servers to store huge amount of data and the applications that are required to manipulate stored data. However, with the evolution of technology, the need for supersized servers became the primary emphasis and several factors such as increasing data storage and operational convenience are inevitable as the sector progresses every day. Cloud computing serves as the solution to manage the ever-increasing requirements of IT based businesses.

Virtualisation of resources should comply with compatibility and performance standards (Joshua & Ogwueleka, 2013). However, performance metrics of these systems suffer from serious issues such as security, failure to recover cloud data, the increasing number of users and so on. The current study is restricted towards the performance benchmarking of virtual systems wherein the scope of the study will be completely focussed upon machine learning and the implementation of linear regression model for performance prediction.

The question which must be considered: Is Machine learning technique a viable approach to calculate performance benchmarking in cloud based virtual environment?

1.2 Research Questions

Can machine learning algorithms be used as a technique to observe the performance complications of Enterprise cloud-based systems running on a virtual environment?

1.3 Research Aim

The main aim of the present study is to determine the feasibility of machine learning algorithms as a viable approach to observe the performance complications of cloud-based systems running on a virtual environment

1.4 Research Objectives

1. To find a model that can benchmark application performance by looking at esxtop values.
2. To propose a model to alert system administrators on a possible permanent degradation of performance.
3. To evaluate and conclude whether Machine Learning is a viable approach to the prediction of performance degradation in cloud

1.5 Scope and Significance of Research

The research shares significance in the implementation of a benchmarking technique to evaluate the performance of systems in cloud based virtual infrastructure. However, the scope of the current study is limited to the implementation of machine learning algorithms using linear regression model.

The next section outlines the organisation of the dissertation.

1.6 Chapterisation of the dissertation

This section provides a chapter by chapter breakdown of the dissertation:

Chapter 2: THE LITERATURE REVIEW

In this section, previous researches based on the purview of the current research study will be analysed.

Chapter 3: RESEARCH METHODOLOGY

This chapter will cover the research methodology which includes the type of methodology used in the research. In the present research, different machine learning algorithms are analysed and a suitable algorithm is selected for the study.

Chapter 4: DATA COLLECTION/ EXPERIMENTS

This chapter describes the process of data collection used in the research and the tools used to perform the investigation. The section explains in detail the type of system architecture used in the present research, and feature selection and modelling.

Chapter 5: RESULTS

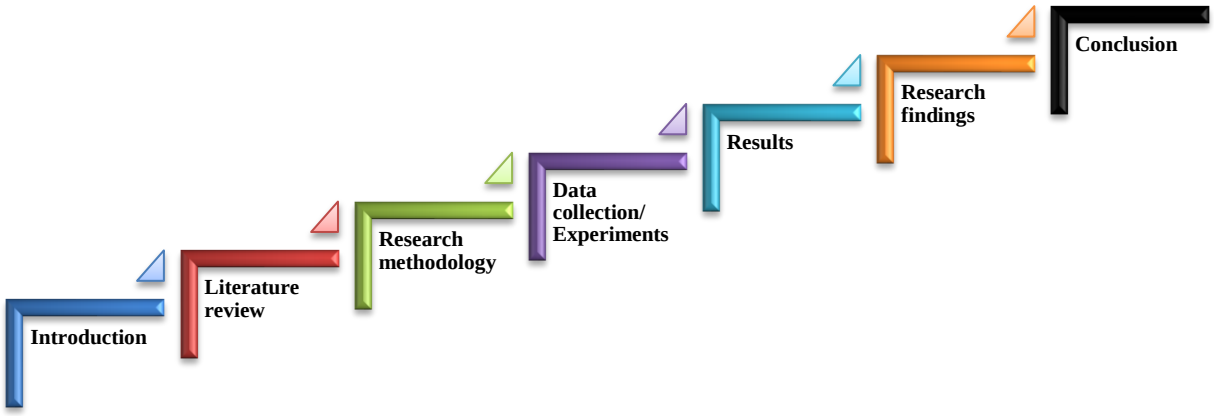
Chapter 5 elaborates the results acquired in the present research. The use of a specific machine learning algorithm to predict fundamental performance degradation in cloud deployed application is expatiated and the results of the model are presented.

Chapter 6: RESEARCH FINDINGS

This chapter summarises the findings of the investigation and discusses to what extent the original aims of the research were met. The implications of problems encountered are discussed and suggestions for future work are made.

Chapter 7: CONCLUSION

This chapter concludes the present research summarising the entire paper and the necessity for future researches in the same context.



CHAPTER II: LITERATURE REVIEW

2.1 Machine Learning Algorithms

In the past years, there have been substantial investigations and rapid progresses in machine learning in computers and information technology. Anon (2012) elucidated the basics of machine learning with a simple statement, 'predicting the future based on the past'. The general idea of machine learning is based on the concept of analysing patterns from the past and predicting the future. Since machine learning is based on predictions, the problems associated with the same pertains to predicting values, yes/no responses, classes and ranking. Different algorithms such as Naive Bayes, the mean classifier, nearest neighbours and the Perceptron are employed to solve problems associated with machine learning.

In light to machine learning and its applications, Harrington (2012) expatiated machine learning with real time examples. Machine learning is predominantly used in many sectors; however, the visibility of the same is hidden. It has profound applications in face recognition, handwriting digit recognition, spam filtering in email, and product recommendations in e-commerce applications. The implication of machine learning is stated as any field that requires data to be interpreted and acted upon can be benefited through machine learning. Machine learning uses statistical methods to predict values. Statistics and engineering principles are coupled in machine learning wherein engineering principles provide solution to problems and statistics determines the method of modelling the solution.

The quintessence of machine learning lie in the prediction of time series values which are considered valuable in many fields. Ahmed *et al*. (2010) emphasised the importance of machine learning algorithms as the means to time series forecasting. The study considered several machine learning algorithms such as Bayesian neural networks, multilayer perceptron, radial basis functions, neural networks (also called kernel regression), generalized regression, K-nearest neighbour regression, and CART regression trees, support vector regression and Gausian processes and compared the efficiency of these systems in terms of performance. The results of the study revealed Gaussian process regression and multilayer perceptron as the best performing algorithms. However, the efficiency of machine learning algorithms depends on data representation. Bengio *et al*. (2014) studied the concept of representation learning which could be used to extract the required information in systems which in turn aids in classification and prediction. The previous research bases on deep

learning that leads to the extraction of abstract information which however are powerful representations of data.

The essence of machine learning can be predominantly found in applications associated with web based manipulations. Awad (2012) studied the use of machine learning algorithms like SVM, KNN and GIS to resolve the web page classification problem. The exponential growth of web pages eased the experience of users who search for information. However, this increases the complexity of managing the web pages (optimising web pages and ranking) and target information will be hindered from deliverance to users. This is mitigated with the aid of two techniques: hierarchical browsing and keyword searching. The varying quality and the information content affect the target information, and the organisation of the pages does not facilitate simple search. In order to manage the issue, different machine learning algorithms are applied which include k-nearest neighbour (k-NN algorithm), Bayesian algorithm, support vector machine (SVM), neural networks, and decision trees.

Chen and Zhang (2014) demonstrated the challenges, techniques and applications of Big data in which machine learning shares a significant part. The study displays the relation between Big data problems and other fields of computing. Researchers claimed that the exploitation of Big data with the aid of machine learning techniques may prove useful to the field of commerce and business applications as immense data can be extracted from Big data. In addition, the study also states that Big data management in cloud computing will progress in the long run since Big data problems, when addressed provide potential developments to cloud systems.

Aleem *et al*. (2015) claimed that machine learning can be used as an efficient tool to detect software bugs. The previous research stated that machine learning algorithms have wide scope in extracting useful information and solve problems where less information is prevalent. This is in line with the context of issues related to the software domain wherein learning is characterised by constant changes in the system in accordance to circumstances. With this purview, machine learning could be implemented as a predictive model to analyse useful data. Machine learning helps in the extraction of information after classification which in turn aids in the analysis of collected data from different perspectives. In light to this context, the study utilised software modules and different machine learning algorithms to study the different datasets with the purview of predicting software defects. The results of the study conveyed that machine learning deemed to be useful over the software bug datasets.

2.2 Machine Learning in Cloud Computing Systems

Though the practical applications of machine learning are profound in many fields, its predominance can never be neglected in the field of cloud computing. In this context, Khanghahi and Ravanmehr (2013) studied the challenges associated with the performance evaluation in cloud computing. Cloud computing resources should be compatible with end user systems and must comply with power and performance factors. High performance is one major issue that should be addressed with top priority. This includes performance of services that have high influence on end users and service providers. Performance evaluation includes assessing the factors such as network capability, number of processes (I/O) that can be processed per second, average waiting time per unit time, throughput, CPU utilisation, and so on. Paliwal (2014) elucidated the challenges pertaining to performance issues in cloud computing wherein three cloud models: Software as a Service (SaaS), Platform as a service (PaaS) and Infrastructure as a Service (IaaS) are investigated and explored. The distinctive features of cloud computing systems are explored which include scalability, resilience, elasticity, multiple tenancy and workload management. Performance based hindrances in cloud deployments are related to the distinctive features of cloud systems wherein the predominant features are availability, performance, scalability and performance. Advanced data analytics, according to this study is considered to be the feasible way to cater better performance in cloud based systems.

Yet another study by Zia and Khan (2012) identified the key challenges with respect to performance in cloud computing and revealed that functional and technical challenges pose serious threat to the performance of cloud systems. Cloud systems should be uniform to provide better quality of service. This includes functionality of the software, power and bandwidth factors. However, utilisation of cloud resources may sometime result in functional and technical complexities such as resource access issues, dynamic discovery and so on which are all performance related issues. The study identified various challenged in cloud computing systems which include storage allocation and services, scaling, scheduling, location of data centers, SQL query processing and architectural constraints. The roles of cloud providers are discussed and the features which affect the performance of cloud systems were identified more generically. Performance of cloud based systems should be managed with methods that imply low cost and better quality of service.

The aforementioned researches all emphasise the need to study the performance issues related to cloud based systems. Ardagna *et al*. (2014) with the same purview studied the challenges associated with the management of quality of service of cloud based systems based on resource allocation with respect to performance, reliability and availability. However, the study concluded with the notion that the emergence of different technologies in cloud based systems makes the analysis of service quality difficult. Several previous studies had focussed on hybrid models which impart the use of queuing models and machine learning techniques. The significance of queuing models lie with the knowledge of system network and the infrastructure which in turn provides precise predictions on performance.

2.3 Machine Learning in Anomaly Detection Systems

In addition, several researchers had identified the potentials of machine learning in anomaly detection systems with the purview of implementing security over the cloud. Bhat *et al*. (2013) proposed an anomaly detection system with the aid of machine learning algorithms for virtual machines on the cloud. According to the report on Internet security (Computer Economics, 2007), the number of attacks over internet resources has increased over the years and has led to financial losses. Further, these attacks are targeted on cloud computing resources. The proposed model of the study detects any intrusion or anomaly in the system which occurs in parallel with the training of the system. The research incorporates the use of Naïve Bayes Tree Classifier and a hybrid approach of Naïve Bayes Tree and Random Forest (Bhat *et al*., 2013). In the same context, Gander *et al*. (2013) uses complex machine learning procedure and event processing rules to detect anomalous behaviour of the system that is being monitored by populating the metrics specific to the users. The researchers consider the application of existing anomaly detection tools to have two-fold problems. Firstly, existing solutions for anomaly detection over the cloud do not detect specific attacks. Secondly, the focus of these tools is specific to only a single layer of network abstraction. This emphasised the need for a monitoring service that renders security to all facets of network. Anomaly detection is coupled with the use of machine learning algorithms (García-Teodoro *et al*., 2009). The proposed model facilitates the monitoring of real time anomalies that occur in cloud based infrastructure, especially those that occur in any layer of the networked system (Gander *et al*., 2013). A survey by Haq *et al*. (2015) focussed on the deployment of machine learning algorithms in the detection of anomalies and intrusion in cloud based systems. With the purview of analysing the current trends in implementing machine learning for intrusion detection in cloud systems, the study considered the essence of 49 related literature studies

and the emphasis is directed towards the classifier design (single, hybrid and ensemble). Related conventions such as the comparison of classifier algorithm, the datasets used and the core experiment modules of previous literature (49 studies) were explored and analysed.

On the contrary, a survey by Ahirwar *et al*. (2014) stated that statistical anomaly detection techniques are more efficient than machine learning approaches. The study surveyed the different types of anomaly detection techniques and generalised them into three categories namely statistical, data mining based and machine learning based anomaly detection. The study discerned the fact that statistical anomaly detection technique is the most efficient anomaly detection technique than the other two techniques. This is extended by the fact that this technique does not require prior knowledge about intrusion or previous patterns of intrusion and recent anomalies can also be detected.

Machine learning is closely related to pattern classification which in turn is related to Support vector machines. Hsu *et al*. (2003) elaborates the concept of support vector classification and support vector machine. Support vector machines (SVM) are used for the purpose of data classification. Its functionalities could be used easily rather than the traditional neural networks. The basic understanding of SVMs begins with the term called 'classification' which means separation of data into training and testing datasets. Every instance of the training set includes a target value and attributes. SVMs function with the basic idea of developing a model based on the training data which forecasts the values of the test data provided the attributes of test data are given. SVM operates in two steps: Data pre-processing and model selection. Data pre-processing implies representation of data in the form of categories and model selection implies the selection of an appropriate model applicable as per the requirement. A study by Luts *et al*. (2010) formulates the functionalities of SVM in pattern classification. The study incorporated the approaches that could be used to perceive data and predict future data. SVMs act as tool to balance risk minimisation and model complexity. SVM classifiers are widely used in the field of bioinformatics and other related disciplines due to the fact that these classifiers are of high accuracy, flexibility in modelling several data sources and the capability to operate with data such as gene expressions (Scholkopf et al., 2004). According to Shawe-Taylor and Cristianini (2004) and Scholkopf and Smola (2002), SVMs fall under the grouping of kernel methods which means an algorithm that operates only on dot-product data. SVM is utilised in intrusion detection systems with limited data wherein data dimensions do not affect the accuracy. A study by Li and Liu (2010) displayed an intelligent module that functions in support to intrusion detection

and prevention system with the aid of SNORT and firewall. In this study, SVM is used with the SNORT to improve the efficiency of intrusion detection system. In a study by Ajila and Bankole (2013) the significance of SVM as a prediction model is emphasised. The study envisions the need for predicting future demands of resource before proffering service level agreements to end users. With this purview, the study proposed cloud client prediction models for TPC-W benchmark web application with the aid of three machine learning techniques namely Support Vector Machine (SVM), Neural Networks (NN) and Linear Regression (LR). SLA metrics such as the response time and throughput are also included in the study with the purview of more robust scaling and decision. The results of the study revealed that SVM provided best prediction other than the other techniques. Yet, Fletcher (2012) also revealed that SVM provides high level of accuracy in terms of prediction which implies that SVM considers even microstructural effects.

One of the most predominant services in cloud systems is the Infrastructure-as-a-Service (IaaS) which enables end users to utilise resources through lease. The advent of IaaS provides application service providers to manage resource needs in a cost effective way which implies reduced complexity of maintaining a separate physical infrastructure. Multiple users can share cloud systems through virtualisation of cloud resources that helps in the isolation of services to end users (Barham *et al*., 2003). However, anomalies in cloud based systems may occur due to reasons such as issues in resource management, and software and hardware systems. These issues arise only during large scale implementation and execution since developers would have tested the systems offline or with small scale implementation. System administrators may find the task of identifying bugs and performances issues in virtual machines complex. These issues led Tan *et al*. (2012) to propose an anomaly detection system called PREdictive Performance Anomaly pREvention (PREPARE) system for virtualized cloud systems. The proposed system combines the functionalities of anomaly detection and prevention of virtualisation based complexities which on the whole prevents anomalies with respect to performance in cloud based systems. PREPARE uses machine learning algorithms to achieve early anomaly detection before any issue induces a performance anomaly. The use of statistical learning algorithms to identify anomalies in cloud systems stem from the study by Bodík *et al*. (2009) which states that the shortcomings on performance of complex real world systems could be addressed with the aid of machine learning algorithms and statistical models. The previous study demonstrated the capabilities of statistical machine learning algorithms to model the training space and the use of a closed-

loop framework to automatically control changes in Internet applications. The study provides valuable insights to researchers on the collaborative study of machine learning, control theory and cloud systems.

2.4 Performance Benchmarking

With the increase in the number of cloud users increases the need for benchmarking the performance of cloud based services. This includes performance assessment of cloud infrastructure which ultimately decides the retaining of cloud users. In addition, cloud services could also be improved through these benchmarks. According to Hanbury *et al*. (2012) , benchmarking is deemed to be a predominant technique in the analysis and retrieval of information With this purview, Folkerts *et al*. (2013) identified one of the approaches to benchmarking in Cloud based systems. The core principle behind cloud benchmarking lies with careful consideration of performance and cost of service. The steps involved defining actors and the System under Test (SUT) which includes a number of black boxes and unstable components. Use cases were developed and analysed wherein it is inferred that benchmarking plays a vital role in performance of cloud based services. The premise of the study provides new insights for scenario based benchmarking. In the same context, Iosup *et al*. (2013) studied the approaches and challenges in benchmarking cloud applications. The study utilises an approach for benchmarking IaaS based cloud services and discusses the challenges in implementation of the same. The study also emphasised the need for testing other than traditional black box and isolated-user testing. Four classes of challenges with respect to benchmarking are discussed: methodological, system property-related, workload related, and metric-related. Other factors that are emphasised with the previous study are benchmarking models that should be more domain-centric and should elasticity and variability metrics.

A similar study by Kang *et al*. (2012) further elucidated the detection of performance anomalies in cloud based systems. Service Level Agreements (SLA) plays a vital role in cloud service contracting in which cloud providers describe the information about type of services and performance metrics to the cloud users. This implies the cloud provider to serve precisely the resources that are stated in the contracted service level agreements with necessary capabilities such as power, bandwidth and so on. However, with the ever-increasing technologies and increase in the number of cloud users, cloud providers find the provision of adequate resources with the stated capabilities in the SLA to users to be

complex. This in turn causes violations in SLA. In order to mitigate such issues, the study utilises a new performance diagnostic framework, known as the DAPA which helps system administrators to analyse issues with respect to performance anomalies and the sources of SLA violations. The system proposed by the study utilises several statistical approaches to analyse the relationship between performance of the application and the performance metrics of the virtual machine. The study implemented a prototype DAPA on a cluster of virtual machines in order to diagnose SLA violations in an enterprise application. The results of the study revealed that DAPA framework could analyse the attributes that are found to be suspicious in performance interference of virtual machines and physical systems which in some way or another is related to violations of SLA.

An association between performance benchmarking and machine learning is explored by many researches and the study by Farias *et al*. (2013) further discerned the estimation of SQL query and response time in the cloud. A majority of cloud based services pertain to assessing data, hence database management is considered to be valuable in powering the applications that form the critical elements of cloud software heap. Since organisations require the quality of service guarantee in the service level agreements (SLA), it is deemed a necessity for cloud providers to offer SLA with adequate information about performance which is closely related to estimation of time and resources. Specific to DBMS systems, cloud decision making could be facilitated if the estimated time for the execution of incoming query is known or is predicted before. A model system is proposed to address the issue of predicting execution time of query before execution which is treated with the aid of machine learning algorithms. The results of the study discerned the quick prediction of query response time. However, errors arise which are negligible in the context of the research.

A study by Amannejad *et al*. (2015) explains the need for machine learning based interference detection techniques to address the issues of performance interference in cloud based systems. It becomes evident with the increasing number of cloud users web services have become highly dependent on cloud based systems. The problems are often associated with virtualisation of physical machine to users with the aid of virtual machines. It is deemed that the performance rate of cloud based systems based on the functionalities of the physical system can be inferred through the detection of clock cycles per instruction. However, this facility is restricted to the end cloud users who will therefore be unable to access such metrics. The access of end cloud users towards performance assessment is limited to metrics of virtual machine such as the transaction response times. In this study, the complexity in the

detection of interference to cloud performance is addressed with the aid of a machine learning interference detection approach. The concept behind this approach is to use a collaborative filtering technique that assesses whether a transaction processed by a web service is affected by/ suffering from interferences. The results can be used to facilitate remedial actions through reporting. The study claims that the proposed model detects 96 per cent of interference issues with very negligible false alarms.

Issues in relation to virtualisation had been studied by number of researchers in addition to the aforementioned research by Amannejad *et al*. (2015). In light to the same context, Nirmala and Sridara (2015) proposed a model that utilises the particle swarm optimisation technique based on k-means++ machine learning algorithm. Virtualisation is the concept of isolating virtual machines that literally run on a single physical machine. Such isolation normally poses serious issues with respect to performance wherein the isolation of performance does not occur as per the design of the virtual architecture. This implies performance degradation wherein the performance of one system may be degraded by the applications that run on other virtual machines. Further, performance degradation is coupled with execution time that is performance issues may affect the execution time. In turn, increased execution time leads to degraded throughput. Considering the implications of these issues, the study implemented particle swarm optimisation technique based on k-means++ algorithm. When compared with other approaches, the proposed model provides better throughput with reduced execution time and cost.

Vazquez *et al*. (2014) compared five different performance benchmarking tools to assess the performance of cloud systems. Cloud systems provide services such as computing, networking and storage access and the access to these resources is through virtual environments. The perspective of computer professionals over the years is to scale resources according to user requirements provided such scaling does not degrade the performance of the system. The need for benchmarking is emphasised by end cloud users to assess the cloud environment that is committed by the cloud service provider. With this purview, the study compares the functionalities of five different performance benchmarking tools which include CloudCmp, CloudStone, HiBench, YCSB, and CloudSuite. These tools assist in the assessment of workloads and cloud computing services. In the selected tools, HiBench is used as a benchmark aid that particularly aims in the elements of Hadoop framework. MapReduce and database component (HDFS) is exercised through the assignment of realistic workload. Benchmarking tasks of HiBench is categorised into microbenchmarks, machine

learning tasks, web search and HDFS benchmarks. The paper however concludes with general implications that selection of proper services for benchmarking is deemed a necessity for cloud based systems.

2.5 Summary

Previous studies were analysed with the purview of elucidating the role of performance benchmarking in virtual infrastructure using machine learning algorithms. According to a number of researches, it is deemed that performance benchmarking is inevitable in many situations of conventional computing. Computing evolved from traditional standalone physical datacenters specific to an organisation to cloud based computational systems. However, different issues arise in cloud based systems which are generally centred on performance. Khanghahi and Ravanmehr (2013) and Paliwal (2014) elucidated the complications in cloud based systems which provided further premise to increasing the performance of these systems. Since performance and performance benchmarking become the premise of interest for the present study, machine learning algorithms are utilised to automate the function of performance benchmarking in Cloud systems and virtual infrastructure. The profound uses of machine learning extend from computing to other fields such as science, finance and so on. A significant functionality of machine learning algorithms is the forecasting of time series. Ahmed *et al*. (2010) and Bengio *et al*. (2014) further studied the significance of machine learning algorithms in finance related predictions such as stock volatility and price prediction. The scope of machine learning is vast; however, in this context, machine learning will purely be studied in the context of cloud computing and virtual infrastructure. Anomaly detection in cloud based systems is another predominant area of operation of machine learning algorithms. Similar to anomaly detection, performance degradation could also be identified through machine learning algorithms. Further, the perceptions of machine learning are focussed on performance benchmarking with the aid of SVM. Several benchmarking tools were elucidated by Vazquez *et al*. (2014) which includes a machine learning tool for performance benchmarking. This further intrigues the researcher to use machine learning algorithms as a tool for performance benchmarking in cloud systems running in virtual environment.

2.6 Motivation of the study

Performance degradation and the mechanisms to identify the same have been conducted by a number of researchers. Ipek et al. (2006) in a study utilised ANN techniques

as a mechanism to predict memory and CPU performance through cross-validation. However, Lee and Brooks (2006) used regression models to forecast power and performance usage of applications in the SPEC2000 and SPECjbb benchmarks. Khan et al. (2007) claimed predictive modelling to mitigate the problems of prediction accuracy in CMP. Dubach et al. (2007) focussed on utilising linear regression and neural networks as a combined model to predict performance degradation in micro-architectural systems. All the previous studies motivated the present research to further narrow down and use any of the machine learning algorithms to predict performance in terms of CPU usage in cloud based system applications.

2.7 Research gap

The prospects of linear regression have been discussed by many previous researchers with special emphasis laid upon SVM and ANN as machine learning algorithms; however, no literature has utilised linear regression to reveal performance degradation in Cloud environment. Hence, the present study focuses on simple linear regression algorithm as the proposed model to identify performance degradation in cloud based environment.

CHAPTER III: RESEARCH METHODOLOGY

In cloud based systems, it is a dire necessity to determine the quality of service and the check periodically whether the aspects of service level agreements are met. Software applications running in the cloud which are memory intensive are required to be monitored so as to identify possible performance degradation which may occur with the increase in the workload. Performance prediction in cloud environments is a task of great importance since degradation due to excessive or overload could not be identified during the development phase where software testing and debugging are not potential enough to identify errors that occur due to unpredicted events. Software degradation issues occur after deploying applications in the cloud. Though new errors due to memory and CPU overload shows up in the future, it is deemed to affect the quality of service rendered by the service provide which over time degrades the services (van Hoorn et al., 2009). In the modern era of computing, grid level operations are replaced with cloud based architecture wherein software performance is a crucial issue. The definition for performance engineering by the ITIL is the set of skills, practices, tools and techniques used in all phases of software cycle to ensure the delivered solution to meet the requirements of the solution in terms of performance (Orand, 2011).

Barber (2006) claimed certain goals of performance management which include increase in the revenue which is triggered by timely software responses, software does not crash due to issues in performance, software refactoring should not occur due to performance degradation, and frequent update of new software with old one due to performance issues. However, in the client side, performance degradation prediction should be a mandatory process so as to identify issues that may occur due to overloaded usage of cloud deployed applications. Okanovic and Vidakovic (n.d.) claims three processes of performance management which are common for both service providers and the client users- service level management, capacity management and problem management. Performance management data should be collected so as to provide a realistic image of the cloud service. Hence, a system to monitor software performance is essential since this system provides necessary data. Furthermore, a system proposed to identify performance degradation should be very little affecting the performance of rest of the system. The present study hence discerns that with certain data collected as performance metrics, trend analysis could be performed which aids towards predicting the future performance of the software application.

In this regard, the present paper utilised linear regression model for performance prediction and trend analysis in cloud deployed applications.

3.1 Machine Learning Approaches

Performance analysis and troubleshooting have heavily used machine learning techniques. The CARVE project makes use of simple regression analysis for predicting the performance impact of memory allocation to VMs. Regression is used by Kundu *et al*. (2010) to map a resource usage profile received from a physical system to one that can be used on a virtualized system. But the accuracy of regression analysis is poor when it is used for modeling the performance of virtualized applications under different levels of resource contention. Tree-Augmented Bayesian Networks were introduced by Cohen *et al*. (2004) to identify system metrics attributable towards SLA violations. An administrator is enabled to predict if certain values for specific system parameters are indicators of application failures or performance violations. Bayesian networks are used to construct signatures of performance problems based on performance statistics and clustering similar signatures to support searching for previously recorded instances of observed performance problems (Cohen *et al*., 2005). The usefulness of Bayesian classifiers are challenged by Bodik *et al*. (2010) and they used logistic regression with L1 regularization to compute the metrics relevant to fingerprint computation. It was effective for automatic performance crisis diagnosis and it facilitated remedial actions. These techniques help in bottleneck identification and predicting whether certain resource usage and/ or application metrics would lead to SLA violations. How much SLA violation would be incurred or how resources should be allocated to prevent future violation are the issues which are unaddressed. Performance prediction can be addressed the following way – given a set of controllable/observable parameters, what would the application's performance be? It can later be used within an optimized resource allocation or VM sizing framework. The use of fuzzy logic is considered by Xu *et al*. (2008) to model and predict the resource demand of virtualized web applications. The VCONF project used reinforcement learning with ANN to tune the CPU and memory configurations of a VM to achieve good performance for its hosted application (Rao *et al*., 2009). CPU resource was the target of these solutions. Memory and I/O contention will also be addressed here. ANN models are applied to address "what-if" questions. The initial application of ANN was promising but it revealed several limitations of its applicability (Kundu *et al*., 2010).

The parameter to capture I/O contention in shared storage platform can make way for arbitrary inaccuracy in the model. Constructing a single model encompassing the entire parameter space in a multi-dimensional model is defective. Initially ANN and subsequently new modelling techniques were discussed. The new techniques should overcome earlier limitations and come up with a set of real-world virtualized server benchmarks. Few models can be used for accurate VM sizing. The power of both ANN and SVM approaches to machine learning is explored. SVMs are normally used as a strong classification technique, but SVM based regression (SVR) is popularly used in systems data modelling. SVR was used in order to build models to predict response time given a particular load for individual workloads co-hosted on shared storage system (Uttamchandani *et al*., 2005). It has also been used to model power consumption as a function of hardware and software performance counters (McCullough *et al*., 2011). SVR was not used for performance prediction of virtualized applications earlier. However, the present research hypothesises that Linear Regression could be used as a preliminary performance degradation prediction algorithm.

The present study adopted the use of linear regression algorithm as a model to predict responses in a quantitative manner. The application of linear regression has been in existence for years in many applications and has served as the premise for the development of modern statistical machine learning algorithms (James *et al*., 2013). The prospects of linear regression are not completely utilised in machine learning wherein other concepts such as SVM and ANN are widely used. These inferences motivated the researcher to focus in the present paper on linear regression to forecast CPU utilisation and performance degradation in cloud environment. Linear regression is identified to be more superior to other regression types such as Gaussian processes as disk I/O prediction becomes more accurate in linear regression (Ahmad & Bowman, 2011).

3.2 Linear Regression

In computer parlance, the most commonly used dynamic resource management approaches used are linear regression, support vector machine (SVM), and artificial neural network (ANN). In general, linear regression is the basic algorithm that acts as the baseline to SVM and ANN algorithms. However, in some cases, system specific features that are generated by the SVM and ANN are also captured by linear regression (Fiala, 2015).

In the field of machine learning, especially as in the case of supervised machine learning, the classification and regression are two divisions. While the output for classification based problems are based on discrete class labels, regression learning is based on the prediction of continuous quantities (Rasmussen & Williams, 2006).

Linear regression is the simplest yet most efficient statistical technique used to determine the scores of a dependent output variable (Y) through the score of the independent variable (X). Since independent variables could significantly predict the values of dependent variable, this regression approach could be used to create the equation of linear approach to predict the values of Y (output and dependent) from X (Heatherlench, 2010) . In regression, the aim is to map one real-valued space to another. Linear regression is identified to be the simplest type of regression: more effective and more efficient (Hertzmann & Fleet, 2010). The linear regression algorithm finds the linear hyperplane which is generalised by the equation of the straight line

'y=mx+c'.

The time taken for prediction is O(1) once the linear model is defined. The application of the linear regression algorithm is deemed to be significant in machine learning approaches and is used in many algorithms such as Support vector machines (SVM) and Locally Weighted Projection Regression (LWPR) (Matsunaga & Fortes, 2010).

3.2.1 Linear regression and CPU utilisation prediction

Prediction of future resource constraints is deemed to be a necessity in the management of cloud based resources. The consumption of CPU resources which is denoted by the percentage of CPU utilisation is calculated in the present study wherein the input data (X) could be number of users (MS-Outlook application) or order lines per day (Oracle HRMS). The linear regression model proposed in the present study predicts CPU utilisation according to the history of past CPU utilisation over a month ago. The proposed algorithm uses linear regression method as a prediction function.

$$\boldsymbol{y_x = mx + c}$$

Wherein for each value substituted for 'x', the value for the variable 'y' which is CPU utilisation will be calculated. The function displays the relationship between the current CPU utilisation and future utilisation which is elaborated in the pseudo code given below:

Linear regression CPU prediction algorithm

***Input**: x, number of users/ order lines per day*

***Output**: CPUPredict*

1. *// approximately predict the CPU utilisation percentage based on past CPU utilisation*
2. For i=1 to *k* do
3. $m \leftarrow \sum(Y_i - Y)(X_i - X) / \sum(X_i - X)^2$
4. $c \leftarrow y - m * X$
5. *// where 'm' is the slope and c is the 'y-intercept'*
6. End For
7. *//use the linear regression function*
8. $CPUPredict \leftarrow m * x + c$
9. *Return CPUPredict*

In the algorithm presented above, the value of input is the number of users/ number of orderliness per day for which the CPU utilisation percentage is to be predicted. In such as scenario, the slope and y-intercept (m and c respectively) are calculated with the mean value of variables X and Y wherein historical data is collected for a period of k = 31 days. Calculating the values for 'm' and 'c' from past data, the CPU utilisation could be predicted. Merely substituting the value of x in equation 8 of the above pseudo code will reveal the predicted CPU utilisation value and the function will be used to forecast the same on the utilisation of all Virtual Machines in the host.

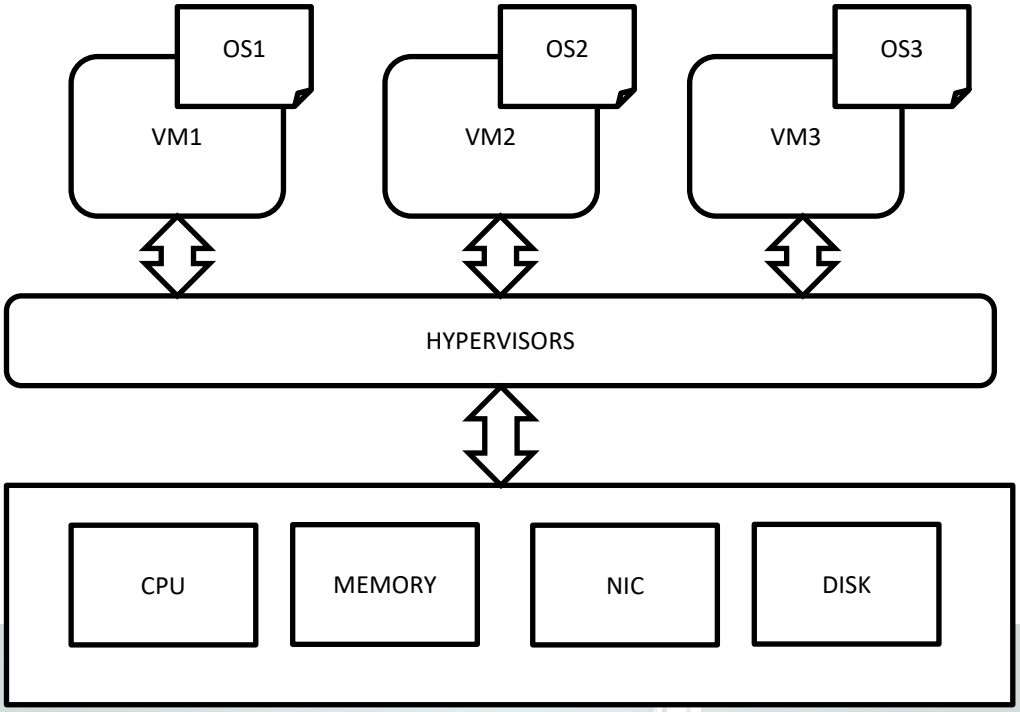
CHAPTER IV: DATA COLLECTION/ EXPERIMENTS

The main objective of the present study is to devise a performance prediction model which analyses the utilisation of a system metrics in the past and provides performance degradation alerts when the number of users exceeds the preferred limit. Such a process involves understanding the system metrics, the analysis of these metrics with the utilisation rate and then forecast the CPU utilisation rate for a number of users. In this context, several previous researches were reviewed and the analysis of those articles is described in the previous sections. However, the following section will discuss the overview of cloud architecture, the materials and methodologies, model training, the type of analysis technique incorporated, and the concepts used in the study. Lastly, the proposed linear regression algorithm is presented.

4.1 Overview of Cloud architecture

The evolution of modern computing strategies in line with the development of cloud based systems has provided adequate space for the emergence of virtual machines (VM). VMs in cloud architecture operate with the execution of an operating system that is isolated from a single physical machine. A single physical machine could hold a number of virtual machines running on it (applications, memory and OS isolated for different virtual machines). The advantages of using virtual machines include increased resource allocation, easy administration and lowered power consumption. Furthermore, such utilisation of physical resources enables end users to obtain on-demand computing resources. The implementation of a virtual system is facilitated with the aid of a hypervisor which allocates resources to VMs. The hypervisor is a software layer that lies between the physical system and the VMs. Resources such as memory and peripheral software are allocated to physical machines through the hypervisor. Furthermore, the hypervisor system is in charge of isolating a virtual system in such a manner that the functions of a VM are envisioned as if it is running on its own hardware. Virtualisation is the key in the isolation of virtual hardware devices such as CPU and memory which is facilitated by hypervisors. As mentioned earlier, hypervisor is a software layer with more than 100,000 lines of code (Perez-Botero *et al*., 2013).

Figure 1: Bare Metal Hypervisor Architecture



Source: Adopted from Pawar and Singh (2015)

4.2 System architecture

The present study imparts the use of machine learning for the prediction of performance degradation and CPU utilisation rate in Cloud environment. Hence the following concepts are described- VMware, Citrix XenApp, ESXi, and esxtop.

VMware- A complete virtualisation suite that provides services such as virtualisation, management of cloud infrastructure, optimisation of resources, availability of applications and the automation of cloud operations. The infrastructure setting of the suite enables the virtualisation and aggregation of physical resources across different systems and creates a virtual resource pool to the datacentre in a virtual environment. Additionally, distributed services such as policy driven resource allocation and consolidated backup of the datacentre are also integrated within the suite. The main aim of integrating all the services in a single suite is to establish cloud services in a cost effective manner (VMware, 2007).

Citric XenApp- Citrix is the leading industry which provides 'XenApp', a solution for application delivery which enables remote accessing of windows application from any

internet enabled device over any network. The datacentre is secured to protect applications and desktops which in turn enhances corporate security (Citrix, 2014).

VMware ESXi- It is a layer that run on physical servers to virtualise processor, storage, memory and resources into many virtual machines (VMware, 2011).

esxtop- It is a command based tool which is used to show the usage of CPU as world and group entities (Baca, 2013).

The present study used 4 HP Blade servers configuration for which the configuration is as follows:

Model: HP ProLiant BL460c Gen8
16CPUs x 2.6999Ghz
Intel(R) Xeon (R) CPU E5-2680@ 2.70 GHz
Processor Socket- 2
Cores per Socket- 8
Logical Processor 32
Hyper threading Active Memory
196GB Physical memory

VMware vSphere 5 Enterprise plus is used for the provisioning of virtual machines. The Enterprise plus edition holds a range of features which transform data centres into virtual infrastructures that are flexible for modern application. The operations management capabilities tend to reveal the issues in performance and optimisation of resources through a single unified console (Haddara & Zach, 2011).The suite is licensed to be installed in two physical systems. Citrix XenApp is installed in all the VMs and only the four specific cores (0-3) of the VMs are used leaving others idle. The virtual disks for the VMs are installed on VMware's clustered file system (VMFS) and ESXi is also installed in the same system (VMFS). The VMFS system is a high performance, scalable, and symmetric file system used to host VMs on a shared storage block. The system utilises clustered locking protocol using storage links (Vaghani, 2010). Storage constraints were met using HPE 3PAR StoreServ 7000 Storage which aids storage of information on a single system supporting advanced features such as tier automation and storage federation. A separate physical machine is allocated for the VMware vSphere client to manage the resources of individual VMs. The statistics of the system utilisation is acquired using the esxtop utility. The esxtop is a useful command to troubleshoot and understand the virtual environment. Similar to the Linux's top command, esxtop provides real time statistics of the virtual environment with respect to

resource planning and utilisation within the ESXi command-line interface (Preston, 2013). The collected data is analysed using a separate HP blade server machine with quad-core Intel 2.7 MHz dual core processor, 8 GBs of physical Memory, 7.2k rpm disk, with Ubuntu-Linux-10.10. The modelling tasks were performed using this machine. Data is collected from live production environment and the process of data collection is done for a period of one month.

4.3 Model Training

A hypervisor is also known as virtual machine monitor (VMM). This is responsible for the allocation of certain resources such as CPU, memory capacity and disk bandwidth to the virtual machines. High levels of virtualization lead to allocation of resources to VMs through measuring the performance metrics such as throughput and response time. In order to acquire the performance metrics of the virtual architecture, periodical recording of the system's parameters should be performed to acquire data sets. This procedure is also known as model training. After the training data sets are acquired, a model should forecast the performance which can be achieved through the certain values assessed from the recorded parameters (Kundu, 2013).

However, no model till date has been developed to benchmark performance metrics which could be used to provide service level alerts to system administrators when a potential degradation of performance is observed. There is no feasibility in acquiring various dimensions of inputs by the machines, process them and act intellectually during critical times. Many researches have been done in this area of which a good number of works emphasized on application performance prediction based on low level performance counters which are related to cache allocation, miss rates and usage(Stewart *et al*., 2008; Doyle *et al*., 2003). It is difficult to utilize such models in virtualized environments since hardware performance counters cannot support at the VM granularity is not available generously in production hypervisors. An opportunity is available to model the performance of an application with regard to size of the VM or underlying hardware resource allocation being provided by virtualized environments. It is possible to change the resources allocated to a VM in an online manner since they are fungible. Hence, the perceptions of the current review focus on the statistical data analysis of the system centric performance metrics from the past experience of the machine.

4.4 Feature selection and data collection

The selection of features for the present research utilised the esxtop command line tool to acquire CPU utilisation data. Two datasets are collected in the present study wherein two workload applications are considered (Oracle HRMS and MS-Outlook). Using esxtop, the CPU utilisation percentage is obtained for a period of one month (31 days). The variables for Oracle HRMS dataset are number of order lines per day and CPU utilisation and the variables for MS-Outlook are number of users and CPU utilisation respectively.

CHAPTER V: RESULTS

In simple terms, regression analysis is a method used to investigate the relationship existing between variables. In the present study where virtualization parlance is considered in cloud environment, the relationships that exist between variables are as follows:

I. Number of sessions versus memory utilization
II. Physical input/ output operations versus the utilization of subsystems

There are different types of regression relations which on a broad context are classified as linear and non-linear regression, and simple and multiple regression (Payne, 2012). In the present study, the simple linear regression technique is used. In linear regression, the aim is to identify the relationship between two variables wherein the general formula is similar to that of the straight line formula which is as follows:

$$Y = mx + c$$

Where C is the y-intercept of the line and m represents the slope. Additionally, 'y' is termed as the dependent variable, target variable or response, 'x' is defined as the independent variable and 'c' is the unknown error or noise. The values of 'm' and 'c' are unknown whereas the values of the dependent variable (y) and independent variable (x) are known quantities (Meinshausen & Kunsch, 2016).

For the analysis of performance, the present study uses linear regression model wherein the research hypotheses the presence of relationship existing between two system metrics such as "user calls" and "utilization of CPU/ CPU usage". In testing, the value of CPU / memory is to be predicted using "user calls" and hence 'user calls' is taken as the independent variable or predictor whereas the value to be predicted is taken as the dependent variable or response variable which is "CPU utilization". Hence, the response variable is represented in the 'Y-axis' and the predictor value is represented in the 'X-Axis'. Applying the study concept to the general formula for linear regression, the following equation is obtained:

CPU utilization (or) CPU usage = user call * m + c

Where m and c are regression coefficients and the values for the same are to be determined. Once the values for the regression coefficient parameters are found, the

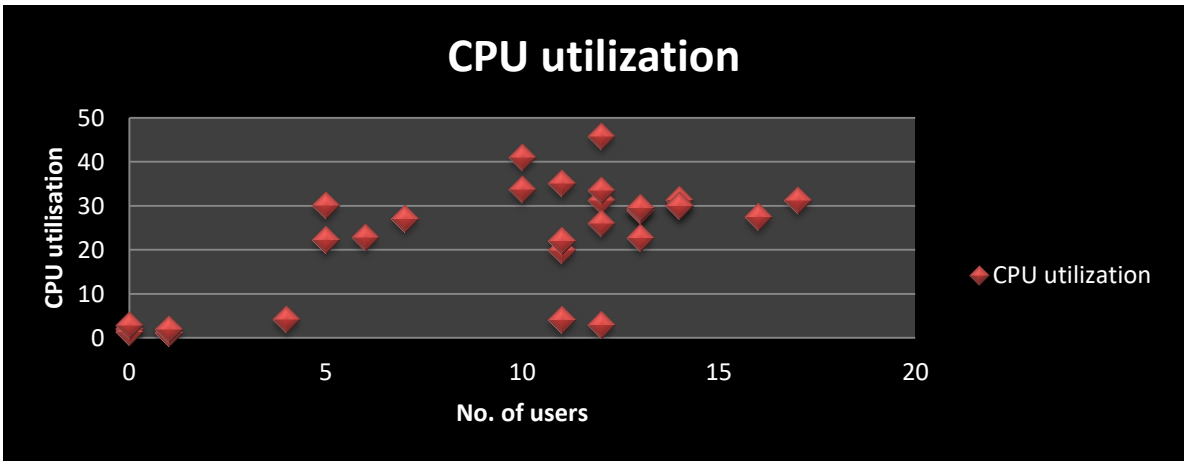
utilisation of system CPU could be predicted by simply substituting the values for user calls. However, the prediction of response variable holds intricate issues and conditions which should be met.

1. If there exists a linear relationship between the dependent and independent variable, then the values of X and Y are plotted in a scatter diagram should reveal a straight line.
2. A strong correlation should exists between response and prediction variable.

Both the conditions imply the existence of strong linear relationship between response and prediction variables.

In the present paper, the performance of organisational support systems will be analysed. The major workload applications used are the oracle HRMS and Microsoft outlook application. The key business metrics for oracle HRMS and outlook application are number of order lines and number of users respectively. The aim of the present study is hence to predict the value of CPU utilization based on the number of order lines or the number of user entered into system. In this regard, the values of average CPU utilisation for MS-Outlook and the Oracle HRMS are recorded for a period of one month (31 days). Following is the graph based on the utilisation of CPU with the number of users utilising the Outlook application for a month (31 days).

Graph 1: CPU utilisation for MS-Outlook



A scatter plot is used to display the variables in the (x,y) format as a 2-dimensional graph. In Graph 1, the response variable (Y) is the CPU utilization/ CPU usage rate and

prediction variable (X) is number of users. Following is the table containing CPU utilization samples and number of users, collected throughout a month (31 days).

Table 1: Sample collection for MS-Outlook

Sample	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Number of users	16	14	13	11	12	0	0	12	6	14	12	13	4	1	5	10
Average CPU Utilization	27.51	31.43	22.74	20.01	2.97	2.1	2.01	45.87	22.83	30.11	31.27	29.04	4.21	1.21	30.11	41.09
Sample	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
Number of users	14	13	12	1	0	11	5	17	11	7	11	0	10	12	13	
Average CPU Utilization	30.01	29.01	33.49	1.93	1.43	35.07	22.31	31.32	22.08	27.11	4.11	2.91	33.73	26	29.59	

In Graph 1, it is revealed that there exists a relationship between the two variables as a straight line; however some points scattered all over the plot which reveals that the CPU utilisation is not following the trend. The scattered away points in the graph are known as outliers. Similar sample collection is carried out for Oracle HRMS application for which Table 2 and Graph 2 are drawn where response variable (Y) is CPU utilization and prediction variable (X) is number of order lines.

Graph 2: CPU utilisation for Oracle HRMS

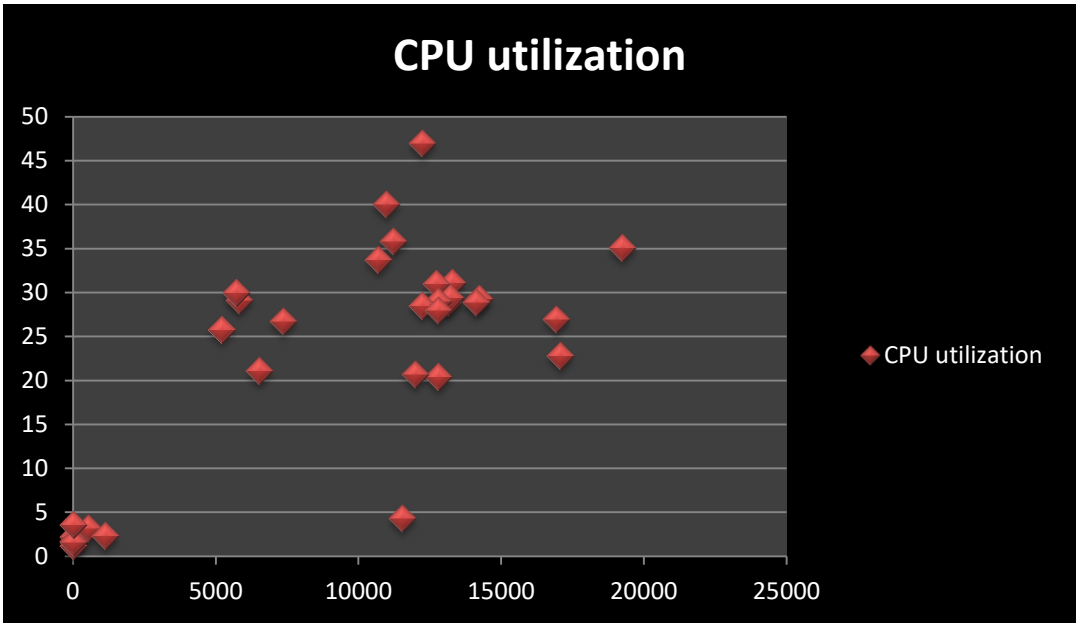


Table 2: Sample collection for Oracle HRMS

Sample	1 day	2 day	3 day	4 day	5 day	6 day	7 day	8 day	9 day	10 day	11 day	12 day	13 day	14 day	15 day	16 day
Order Lines /day	16934	13294	12789	11987	0	0	12229	6511	14234	12722	13140	534	1	5821	5721	10966
Average CPU Utilization	26.98	31.11	20.45	20.73	1.22	2.22	47	21.08	29.27	30.99	28.93	3.21	1.71	29.11	29.99	40.07
Sample	17 day	18 day	19 day	20 day	21 day	22 day	23 day	24 day	25 day	26 day	27 day	28 day	29 day	30 day	31 day	
Order Lines /day	14121	13215	19234	1123	0	11234	5214	17073	7350	11550	0	10699	12802	12237	12801	
Average CPU Utilization	28.90	29.44	35.11	2.33	1.56	35.87	25.78	22.79	26.73	4.32	3.6	33.72	29.01	28.52	28.01	

Both Graph 1 and Graph 2 reveal the relationship between the variables which appear to be in a straight point whereas some points are found scattered from the line. These scattered points are known as outliers.

5.1 Analysing the relations with Correlation Coefficient

The measure of strength and the direction of linear relationship between the prediction variable and the response variable are calculated using the correlation coefficient. Basically, the value of correlation coefficient should exist between the range -1 and +1; however the higher the value of r (positive values), the stronger is the relationship between the two variables. However, with negative values, it is discerned that the relationship between the predictor and the response variable is negative which means with increase in X value, Y value decreases and vice versa.

Correlation coefficient is calculated using the formula as follows:

$$\text{Coefficient correlation}, r = \sum(y_i - y)(x_i - x)/\text{SQRT}(\sum(x_i - x)^2\,(x_i - y)^2)$$

The traditional calculation of coefficient correlation is replaced with the calculation of 'r' using the function CORREL () in MS-Excel (The Nuffield Foundation, 2008). The square of coefficient correlation (r^2) is calculated which represents the prediction variable. From the collected data samples, calculating the r value for the relationship between the number of users and CPU utilisation in outlook sessions, the r2 value is found to be 0.508 wherein r= 0.713265. This discerns the fact that 50.8 per cent of CPU utilization is used during outlook user sessions. However, the remaining 49.2 per cent utilisation could not be explained. When the value lies within the ± 1 range, then there exists perfect correlation between response and prediction variable and if the values is zero, then there is absence of correlation. Any other value of correlation coefficient within the range -1 and +1 indicates limited correlation.

Correlation Coefficient (r)	Practical Meaning
0.0 to 0.2	Very Weak
0.2 to 0.4	Weak
0.4 to 0.7	Moderate
0.7 to 0.9	Strong
0.9 to 1.0	Very Strong

Source: Adopted from Rowntree (2003)

5.2 Estimation of regression coefficient for outlook data

Since it is established that there exist a strong relationship between the response and predictor variable, the regression coefficient is to be estimated which is similar to the

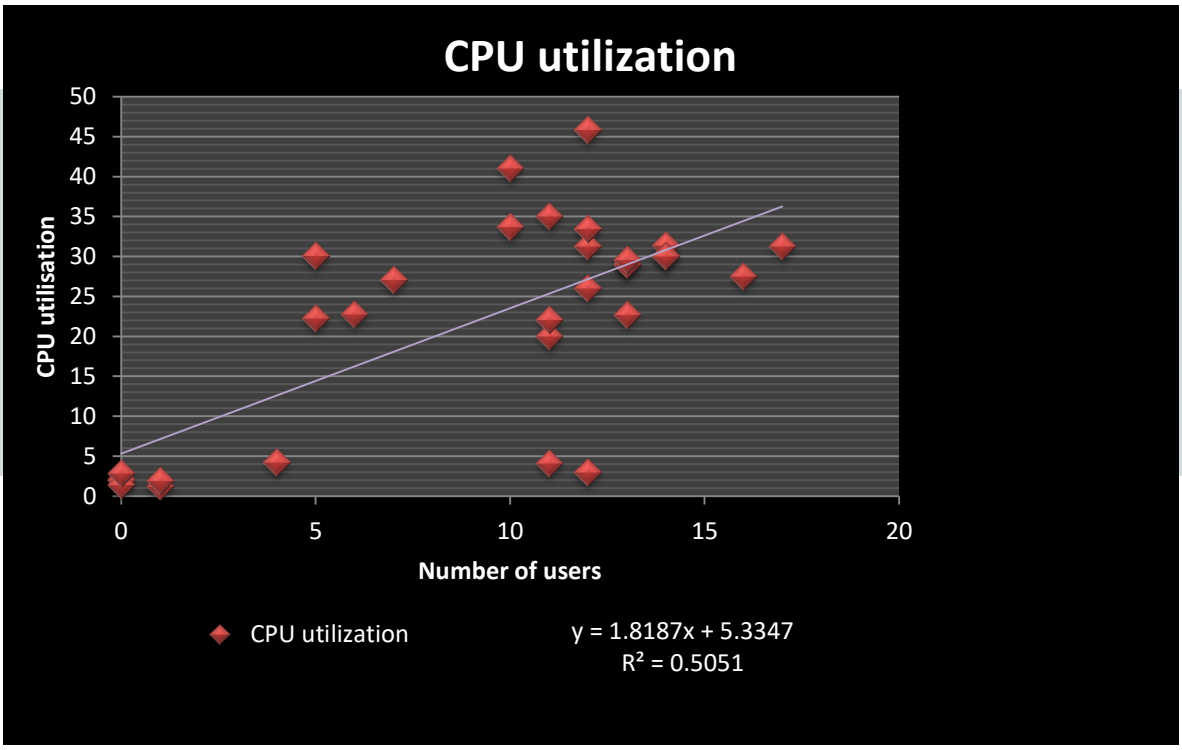
measurement in straight line equation wherein the best fit point with the line will be considered. Based on the least square method, the calculation of the values of slope 'm' and the y-intercept 'c' is given as follows:

$$\boldsymbol{Slope}, m = \sum (Y_i - Y)(X_i - X) / \sum (X_i - X)^2$$

$$\boldsymbol{y - intercept}, c = y - m * X$$

Where, 'X' denotes the mean value of all the x-values (number of users), and 'Y' denotes the mean value of all the y-values (CPU utilisation). In this manner, the values of y-intercept, slope and r2 are calculated (Graph 3).

Graph 3: Average CPU Utilisation for Outlook users



From the graph, it is discerned that Slope m = 1.8187 and y = intercept c = 5.3347.

5.3 Estimation of regression coefficient for Oracle HRMS data

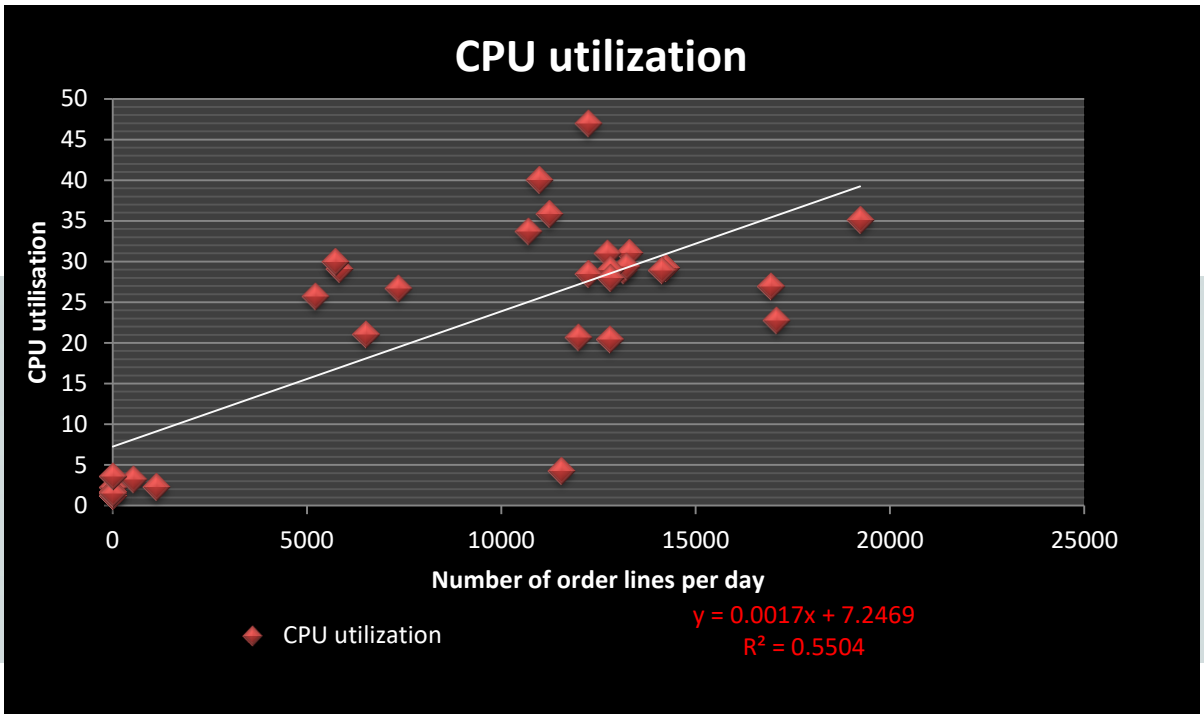
Similar to the calculation of regression coefficient in the case of Outlook user sessions, the same is calculated for Oracle HRMS data. Based on the least square method, the calculation of the values of slope 'm' and the y-intercept 'c' is given as follows:

$$\text{Slope}, m = \sum (Y_i - Y)(X_i - X) / \sum (X_i - X)^2$$

$$y - \text{intercept}, c = y - m * X$$

,where 'X' denotes the mean value of all the x-values (order lines per day), and 'Y' denotes the mean value of all the y-values (CPU utilisation). In this manner, the values of y-intercept, slope and r2 are calculated (Graph 4).

Graph 4: Average CPU Utilisation for Oracle HRMS users



From the graph, it is discerned that Slope m = 0.0017 and y = intercept c = 7.2469.

5.4 Forecasting through Linear regression

Examining both the workload applications by analysing the collected samples revealed the values for Slope 'm' and the y-intercept 'c'. All the values could be used to forecast the performance of cloud based application which is explained using the following illustration:

Case 1: Predicting CPU utilisation from the value of number of users (MS- Outlook)

Suppose the number of outlook user sessions in a day be 14. For outlook workload, the calculated values for Slope m = 1.8187 and y = intercept c = 5.3347 and hence the value for y (CPU utilisation) could be predicted as,

Y-estimate (CPU utilisation) = 1.8187 * 14 + 5.3347

= 30.7965%

From the collected data set for number of users utilising Outlook for a month, the average CPU utilisation of 14 users in a month is 30.11 per cent and the predicted value for the same number of users is 30.7965 per cent which holds very meagre difference which is known as 'residual'. In general, a positive residual denotes an underestimated value of 'y' at that point and negative value of y implies overestimated 'y' value.

Case 2: Predicting CPU utilisation from the value of number of order lines/ day (Oracle HRMS)

Suppose the number of order lines per day be 7350. For Oracle HRMS workload, the calculated values for Slope m = 0.0017 and y = intercept c = 7.2469 and hence the value for y (CPU utilisation) could be predicted as,

Y-estimate (CPU utilisation) = 0.0017 *7350 + 7.2469

= 19.74 %

From the collected data set for number of order lines per day for a month utilising Oracle HRMS, the average CPU utilisation with 7350 order lines per day is 26.73 per cent and the predicted value for the same number of order lines per day is 19.74 per cent which holds a considerable difference (6.99 per cent difference) which is known as 'residual'. In general, a positive residual denotes an underestimated value of 'y' at that point and negative value of y implies overestimated 'y' value.

5.4.1 Residual removal

In a scatter plot plotted with values of variables to find the straight line, there occurs data points which are not in trend hence are away from the main data trend. These points are known as 'outliers' (Williams, 2016). The causes of outliers are varied- backups, report generation or misinterpretation problems that occur during data collection (Ishikawa *et al.*,

2010). It is deemed that processes which are not a part of the general routine during production will generate outliers. Removal of outliers is an important area of forecasting using regression analysis wherein it is necessary to analyse whether such outliers are part of the production workload. Residuals are standardised by dividing the residuals by the standard error estimates. The standardised residuals have mean 0 and standard deviation as 1 (Gelman & Hill, 2007). Two common modes of calculating standardised residuals is as follows- Residual mean square error from model fitted to the complete dataset, Residual mean square error from model fitted to the complete dataset except the i-th observation (Blatná, 2006). Plotting the standardised residuals against the values of response variables (estimated) will display the standardised residual plot.

In our case, it was Friday when sample number 22 data was collected. There were scheduled batch processes running in the background which made the CPU high utilization which is the reason for positive residual (underestimated CPU utilization). Since these batch processes are not run regularly and does not fall under business requirement, such data could be removed from out data set with proper notification and documentation. So the bottom line is never to forecast in nonlinear areas. For CPU subsystems, this limit is around 65 % for outlook application and for oracle HRSM it is 70 % of CPU is the optimal benchmark.

CHAPTER VI: RESEARCH FINDINGS

6.1 Need for performance prediction in cloud environments

6.1.1 SLA violation issues

Performance management is deemed to be important in hosting enterprise services in the cloud environment. For a cloud administrator and the user of a cloud, SLAs guarantee the objectives of the service delivered provided certain criteria such as response time are to be met. However, there is also a necessity for cloud administrators to efficiently utilise server resources to save costs of computing. These inferences put forth certain questions such as- a) what will be the performance delivered to the end user and how much resources shall be allocated to necessarily achieve performance? b) In what ways shall system performance problems be avoided with changing customer workloads? End users always look for performance even when services are migrated within platforms of when he/she is to venture newly into cloud environment.

There are strictly no exact answers to the aforementioned questions since these answering these questions is even a complicated task in non-cloud based distributed computing systems (Menascé *et al*., 1994). Since resources are shared in cloud environments, the task becomes even more complex. Moreover, since changes in the provisioning of could services may affect the operations of users/ enterprises, capacity planning should be performed during operation and performance prediction to be performed on a timely basis. This calls upon the proactive management of performance metrics in order to avoid SLA violation penalties which further require the prediction of application performance under different workloads (Huber *et al*., 2011).

6.1.2 Scaling resources in the cloud

An underlying issue with cloud computing is the scaling of resources in the cloud. Before the advent of cloud computing, traditional model implies the job scheduling and queing wherein the scheduler takes the job of executing the processes in the queue. To effectively schedule processes, user may request for resources which are required to complete a process. In such a scenario, all resources are requested and a trade off is made to completed the task quickly. However, this is not applicable in a cloud environment wherein users allocate the required resources. Scheduling of tasks is restricted/ abstracted in the cloud

which is to facilitate cloud use. However, certain tasks which are meant to fit an application may not actually serve the same. Only a small proportion of applications fit specific domain in the cloud and hence Infrastructure-as-a-Service (IaaS) is commonly adopted by users rather that Platform-as-a-Service (PaaS). In IaaS, users are provided the access to utilise virtual hardware which are allocated to their use. Scaling issues are so common in PaaS.

In cloud computing, there are two major concepts- shared infrastructure and abstraction. When a cloud service is opted by a user, the user is given an isolated environment; however, the infrastructure that is provided is shared between multiple users. Furthermore, a layer of abstraction decouples the infrastructure from the applications that access it. This type of decoupling is the process of deploying applications wherein the security, reliability and mapping of applications onto physical machines are all maintained by the cloud provider. It is deemed that sharing of administration and infrastructure aids users with economic benefits.

6.1.3 Capacity and resource planning

Virtualisation is the concept in which part of the disk, CPU and memory related resources on physical machines is dedicated to the applications running on it. The operations in cloud environment are two fold- the physical infrastructure and virtual machine placement are managed by the cloud provider wherein the operating systems and the type of application are managed by the users of the cloud service. In this case, cloud administrators isolate resources based on 'hard isolation' mode wherein there is little or literally no interaction between the applications on a physical machine. Additionally, no resources are shared in this scenario. Such isolation of resources and cloud services suffers from various performance issues, which in the context of managing database systems prove more hazardous. Following are the reasons-

1. Capacity planning is performed most often by calculating the virtual hardware resource allocation to resources. This ultimately results in imprudent measurement of performance metrics.
2. Multiplexing and sharing is prevented in hard isolation, hence there is a considerable loss of performance.

3. Decoupling of cloud services prove fatal for tuning or provisioning of DBMS in cloud (Mozafari *et al*., 2013).

All these reasons tend to limit the use of VM-based database-as-a-service (DBaaS) by many business enterprises. This motivated the present study to define what is required in the case of VM-resource sharing- Performance prediction. The solutions to all the aforementioned problems relate the need for performance prediction wherein a model is required to predict CPU utilisation for a given input. In the present study, two applications, MS-Outlook and Oracle HRMS are considered that are CPU-intensive. In the present study, the aforementioned problems could be addressed as with a model to predict CPU utilisation, performance could be predicted by merely determining the value.

6.2 Findings of the proposed model

All the aforementioned issues in cloud environment are intricacies that lurk within any cloud based service and hence there is a need to address the issues. In this context, the present research considers performance prediction to be the most suitable strategy to identify the optimal benchmark for applications that are executed and run by end users in cloud environments. There are several important factors that affect the performance of cloud services which are described as follows:

i. Security- Users claim the risks of DDoS attacks in the cloud which may affect network performance
ii. Data recovery- Time taken to recover data on system failure is subjected to affect performance
iii. Storage capacity, buffer capacity, disk capacity, fault tolerance
iv. Increase in the number of users greater than the specific capacity (Khanghahi & Ravanmehr, 2013).

In the present study, the fourth factor 'increase in the number of users greater than specific capacity' is considered for which a performance prediction model is proposed. The present study utilised linear regression algorithm to forecast CPU utilisation for two specific applications running in cloud environment. We envision the degradation in the performance of cloud systems with increase in the number of users utilising a particular application or the number of order lines per day. The esxtop facility is used to collect information about the number of users and the number of order lines per day for a period of 31 days. The collected

data with the percentage of CPU utilisation is acquired for which regression analysis is performed.

The proposed linear regression algorithm takes the collected historic data which is the number of users/ number of order lines per day (X) and the CPU utilisation percentage (Y). With these data and the collected data for a period of 31 days, the slope 'm' and the y-intercept 'c' are calculated. The linear regression equation is hence set with,

i) Slope m= 1.8187 and y-intercept c= 5.3347 for MS-Outlook

ii) Slope m = 0.0017 and y = intercept c = 7.2469 for Oracle HRMS

The calculations for regression analysis to find the regression coefficient, slope and y-intercept were performed using Microsoft Excel and the graph is plotted for both MS-Outlook and Oracle HRMS application. We discern the presence of relationship between user call and CPU utilisation in our cloud environment. The regression analysis results revealed the value for R2 to be 0.5051 for outlook usage and 0.5504 for Oracle HRMS. This implies the dependency of CPU utilisation with the number of Outlook users as 50 per cent and 55 per cent for Oracle HRMS. However, there may be other factors affecting CPU utilisation in cloud based environment. Pu *et al*. (2010) in this context states that network I/O applications are the dominant workloads in cloud computing systems. However, in the current context, the pattern of CPU utilisation may differ based on various factors. Time of the day plays an important role in the usage of CPU. There is an increase in the number of users during the day and less users transacting with the system during the night. This implies the maintenance of sufficient resources all the time in the cloud environment and the service opted by the user to be expansive in order to provide uninterrupted services even with increased number of users. However, such over allocation of resources may lead to under-utilisation of the same if the number of users may not consistently increase or may not show increased patterns often (Armbrust *et al*., 2009).

The proposed model utilises esxtop to acquire CPU performance metrics. With the CPU data, the present research uses linear regression algorithm to predict performance degradation in cloud based applications wherein these applications are memory intensive. This implies the first objective of the present study to have been met wherein simple esxtop values are used to benchmark application performance. However, the proposed idea is not a stable concept. As in the case of the present research, two applications- MS- Outlook and

Oracle HRMS are considered to be deployed in the cloud and the values of number of users and CPU usage are calculated using the data collected for a specific month. It is however deemed that the values of CPU usage may vary over time and hence it is doubtful to utilise the present model as an accurate prediction tool. In such a case, there should be periodic collection of data from the cloud using esxtop so as to accurately predict CPU performance. Simply inserting values of number of users/ number of order lines in the present model will reveal CPU usage percentage which is susceptible to certain degree of noise, which is attributed to unpredicted events in the system. In several cases, CPU usage may vary due to number of background processes running in the system. These inferences necessitate acquiring data of CPU with high and low number of users for a specific period without noisy data which occur due to background processes running in the system. With these requirements considered in the proposed model, system administrators can predict possible permanent degradation of performance in cloud based applications thereby satisfying the second objective of the research.

However, the implementation of the proposed model in cloud based architecture could be performed with the aid of cloud programming framework called 'Hadoop'. According to Sheshasaayee and Lakshmi (2014), a number of machine learning algorithms could be implemented as MapReduce programmes in hadoop. This programming architecture even enables the end user to manage large data sets if many applications are to be monitored for performance degradation in cloud systems. The application of machine learning in cloud architecture is evidenced in the aforementioned statement and hence, there are possibilities of implementing the proposed linear regression model to alert performance degradation in cloud based applications.

A relationship definitely exists between the user call and the CPU utilization for both sets of data considered in the study. The output from this model is a good indicator of CPU usage as user call increases. Being a simple linear regression, the inverse of this relationship exists and can be easily computed. Given the data, more advanced and/or non-linear techniques might be a better fit. However, this will obtained at a cost of interpretability and proven existence of an inverse. In the context of an IT administrator required to make a quick decision, this model is a great starting point quantifying future performance.

CHAPTER VII: CONCLUSION

A simple statement given by Anon (2012) for machine learning which is 'predicting the future based on the past' became the main premise for the present research. The present paper utilised linear regression machine learning algorithm as the performance benchmarking algorithm to predict performance metrics in cloud based virtual environment. When users choose cloud environment, the purpose and objectives of a cloud service is to deliver performance which denotes the quality of service. Cloud administrators and end users/ business enterprises make an agreement known as SLA as a policy document guaranteeing the quality of selected cloud service. However, cloud users suffer from performance issues which are mostly concerned with the provisioning of cloud services from the service provider. However, some applications in the cloud, which are intended to be used based on limited capacity are often overloaded and hence, the performance of the cloud is ultimately affected. This became the premise of the present paper and hence a performance prediction model was devised. The collection of CPU usage metrics is performed using esxtop values and the collected data is analysed.

The proposed model uses linear regression algorithm to predict performance degradation in cloud environment. The present study acquired historical data of CPU utilisation for a month (31 days) using esxtop facility for two CPU intensive applications- MS-Outlook and Oracle HRMS. All virtual machines were installed with Citrix XenApp 7.6. Collected data were analysed in a separate machine and Microsoft-excel is used to perform regression analysis. The collected data are analysed for which the regression equation was identified. The present study revealed 50-55 per cent of dependency existing between CPU utilisation and the number of user utilising the application in the cloud. This further reveals the existence of other factors affecting CPU utilisation in selected applications deployed in the cloud. However, the simple model could be used to alert system administrators on a possible permanent degradation of performance if implemented as an application running in the cloud, monitoring CPU usage and performance degradation.

The present study is a fundamental model for system administrators to predict performance of cloud systems as the needs of computing in business enterprises are ever increasing. Suppose the number of users utilising a specific application in cloud is increasing every month in an enterprise. Business firms require continuity in the management of projects and every single minute of operation is valuable. However, when cloud services are

established, they are intended to provide efficient services for specified rated capacity. Administrators in the cloud user end should monitor performance of the cloud service in a periodic basis to identify potential performance degradation that may occur in the future. The proposed model is an excellent tool to quantify system performance. Machine learning could hence be used as a viable approach to the prediction of performance degradation in cloud. However, future studies are required to implement more accurate performance prediction models thereby facilitating system administrators to accurately choose and scale cloud services.

7.1 Limitations and Recommendations

The study is limited to the development of a regression model which will be used to predict the performance of cloud systems when the value of independent variable is merely substituted to the equation. The study discerns the fact that from the collected historical data, the optimal benchmark limit for Outlook application is 65 per cent and for the Oracle HRMS, it is 70 per cent. These values are subjected to change with different historical data. Hence the study defines certain limitations which may also be taken as recommendations for future research.

1) The present study analyses the business metrics of two CPU intensive applications namely MS-Outlook and Oracle HRMS. Hence, the proposed model could be extended to other applications deployed in Cloud which are CPU intensive.
2) The study recommends the need for collecting historic data of CPU intensive applications deployed in cloud based environments and the analysis of such data should be performed to obtain the required performance prediction equation. The values of CPU utilisation may vary in every set of data with changes in the number of users/ number of order lines.
3) The study considers the issues of performance degradation only on the basis of increasing number of users or increased number of order lines which are application-specific. There are other business metrics which are also important to predict CPU utilisation. In the present paper, we identify 50-55 per cent of dependency existing between number of users and CPU utilisation which reveals the existence of other factors. Hence, the study recommends future studies to be conducted to reveal important factors affecting CPU utilisation in cloud environments.

4) The proposed model utilised linear regression as a machine learning algorithm to identify performance degradation by simply entering numerical values and predicting CPU utilisation. The present study recommends the use of SVM and ANN to predict CPU utilisation as future studies.

REFERENCES

Ahirwar, M.K., Ahirwar, M.K. & Chourasia, U. (2014). Anomaly Detection In The Services Provided By Multi Cloud Architectures: A Survey. *IJRET: International Journal of Research in Engineering and Technology*. 3 (4). pp. 196–200.

Ahmad, M. & Bowman, I.T. (2011). Predicting system performance for multi-tenant database workloads. In: *Proceedings of the Fourth International Workshop on Testing Database Systems - DBTest '11*. [Online]. 2011, New York, New York, USA: ACM Press, pp. 1–6. Available from: http://portal.acm.org/citation.cfm?doid=1988842.1988848.

Ahmed, N.K., Atiya, A.F., Gayar, N. El & El-Shishiny, H. (2010). An Empirical Comparison of Machine Learning Models for Time Series Forecasting. *Econometric Reviews*. [Online]. 29 (5-6). pp. 594–621. Available from: http://www.tandfonline.com/doi/abs/10.1080/07474938.2010.481556.

Ajila, S.A. & Bankole, A.A. (2013). Cloud Client Prediction Models Using Machine Learning Techniques. In: *Computer Software and Applications Conference (COMPSAC), 2013 IEEE 37th Annual*. 2013, Kyoto: IEEE, pp. 134 – 142.

Aleem, S., Capretz, L.F. & Ahmed, F. (2015). Benchmarking Machine learning Techniques for Software Defect Detection. *International Journal of Software Engineering & Applications*. 6 (3). pp. 11–23.

Amannejad, Y., Krishnamurthy, D. & Far, B. (2015). Detecting performance interference in cloud-based web services. In: *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. 2015, Texas, USA: IEEE, pp. 423 – 431.

Anon (2012). *A Course in Machine Learning*. [Online]. Available from: http://ciml.info/dl/v0_8/ciml-v0_8-all.pdf.

Ardagna, D., Casale, G., Ciavotta, M., Pérez, J.F. & Wang, W. (2014). Quality-of-service in cloud computing: modeling techniques and their applications. *Journal of Internet Services and Applications*. [Online]. 5 (1). pp. 11. Available from: http://jisajournal.springeropen.com/articles/10.1186/s13174-014-0011-3.

Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G., Patterson, D.A., Rabkin, A., Stoica, I. & Zaharia, M. (2009). *Above the Clouds: A Berkeley View of Cloud Computing*. [Online]. University of California at Berkeley. Available from: https://www2.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf.

Awad, W.. (2012). Machine Learning Algorithms in Web Page Classification. *International Journal of Computer Science and Information Technology*. [Online]. 4 (5). pp. 93–101. Available from: http://www.airccse.org/journal/jcsit/4512ijcsit08.pdf.

Baca, S. (2013). *Performance and Troubleshooting with esxtop*. Expert Reference Series of White Papers. [Online]. Available from: http://www.globalknowledge.nl/content/files/documents/White-Papers/664368/WP_VM_Performance_and_troubleshootinng_esxtoppdf.

Barber, R.S. (2006). *BPT Part 2: A Performance Engineering Strategy*. [Online]. PerfTestPlus, Inc. Available from: http://www.perftestplus.com/resources/BPT2.pdf.

Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I. & Warfield, A. (2003). Xen and the art of virtualization. *ACM SIGOPS Operating Systems Review*. [Online]. 37 (5). pp. 164. Available from: http://portal.acm.org/citation.cfm?doid=1165389.945462.

Bengio, Y., Courville, A. & Vincent, P. (2014). Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. [Online]. 35 (8). pp. 1798–1828. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6472238.

Bhat, A.H., Patra, S. & Jena, D. (2013). Machine Learning Approach for Intrusion Detection on Cloud Virtual Machines. *International Journal of Application or Innovation in Engineering & Management*. 2 (6). pp. 57–66.

Blatná, D. (2006). *Outliers in Regression*. [Online]. Available from: http://www.laser.uni-erlangen.de/papers/upload/251.pdf.

Bodik, P., Goldszmidt, M., Fox, A., Woodard, D.B. & Andersen, H. (2010). Fingerprinting the datacenter: automated classification of performance crises. In: *Proceedings of the 5th European conference on Computer systems - EuroSys '10*. [Online]. 2010, pp. 111–124. Available from: http://dl.acm.org/citation.cfm?id=1755913.1755926.

Bodík, P., Griffith, R., Sutton, C., Armando, F., Jordan, M. & Patterson, D. (2009). *Statistical Machine Learning Makes Automatic Control Practical for Internet Datacenters*. [Online]. Available from: http://homepages.inf.ed.ac.uk/csutton/publications/HotCloud09.pdf.

Chen, C.L.P. & Zhang, C.-Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences*. [Online]. 275. pp. 314–347. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0020025514000346.

Citrix (2014). *XenApp*. [Online]. Available from: https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/xenapp-datasheet.pdf.

Cohen, I., Goldszmidt, M., Kelly, T., Symons, J. & Chase, J.S. (2004). Correlating instrumentation data to system states: a building block for automated diagnosis and control. In: *Proceeding OSDI'04 Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation - Volume 6*. 2004, Berkeley, CA, USA: USENIX Association, pp. 16–16.

Cohen, I., Zhang, S., Goldszmidt, M., Symons, J., Kelly, T. & Fox, A. (2005). Capturing, indexing, clustering, and retrieving system history. *ACM SIGOPS Operating Systems Review*. 39 (5). pp. 105–118.

Computer Economics (2007). *Annual Worldwide Economic Damages from Malware Exceed $13 Billion*. [Online]. 2007. Available from: http://www.computereconomics.com/article.cfm?id=1225. [Accessed: 31 May 2016].

Doyle, R.P., Chase, J.S., Asad, O.M., Jin, W. & Vahdat, A.M. (2003). Model-based resource provisioning in a web service utility. In: *Proceeding USITS'03 Proceedings of the 4th conference on USENIX Symposium on Internet Technologies and Systems-Vol:5*. 2003, Berkeley, CA: USENIX Association, pp. 5–5.

Dubach, C., Jones, T. & O'Boyle, M. (2007). Microarchitectural Design Space Exploration Using an Architecture-Centric Approach. In: *MICRO 40 Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture*. 2007, Washington, DC, USA: IEEE Computer Society, pp. 262–271.

Farias, V.A.E., Maia, J.G.R., Sousa, F.R.C., Moreira, L.O., Campos, G.A.C. & Machado, J.C. (2013). *A Machine Learning Approach for SQL Queries Response Time Estimation in the Cloud*. [Online]. Brazil. Available from: http://www.lbd.dcc.ufmg.br/colecoes/sbbd/2013/0023.pdf.

Fiala, J. (2015). *A Survey of Machine Learning Applications to Cloud Computing*. [Online]. 2015. Available from: http://www.cse.wustl.edu/~jain/cse570-15/ftp/cld_ml.pdf. [Accessed: 23 June 2016].

Fletcher, T. (2012). *Machine Learning for Financial Market Prediction*. [Online]. University College London. Available from: http://discovery.ucl.ac.uk/1338146/.

Folkerts, E., Alexandrov, A., Sachs, K., Iosup, A., Markl, V. & Tosun, C. (2013). Benchmarking in the Cloud: What It Should, Can, and Cannot Be. In: *Performance Evaluation and Benchmarking*. [Online]. pp. 173–188. Available from: http://link.springer.com/10.1007/978-3-642-36727-4_12.

Gander, M., Felderer, M., Katt, B., Tolbaru, A., Breu, R. & Moschitti, A. (2013). Anomaly Detection in the Cloud: Detecting Security Incidents via Machine Learning. In: *Trustworthy Eternal Systems via Evolving Software, Data and Knowledge*. [Online]. pp. 103–116. Available from: http://link.springer.com/10.1007/978-3-642-45260-4_8.

García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G. & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*. [Online]. 28 (1-2). pp. 18–28. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0167404808000692.

Gelman, A. & Hill, J. (2007). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. New York: Cambridge University Press.

Haddara, M. & Zach, O. (2011). ERP Systems in SMEs: A Literature Review. In: *Proceedings of the 44th Hawaii International Conference on System Sciences*. [Online]. 2011, IEEE, pp. 1–10. Available from: http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/vmware-vsphere_pricing-white-paper.pdf.

Hanbury, A., M¨uller, H., Langs, G., Weber, M.A., Menze, B.H. & Fernandez, T.S. (2012). Bringing the algorithms to the data: cloud–based benchmarking for medical image analysis. In: *Third International Conference of the CLEF Initiative, CLEF 2012, Rome, Italy, September 17-20, 2012. Proceedings*. 2012, Springer Berlin Heidelberg, pp. 24–29.

Haq, N.F., Rafni, M., Onik, A.R., Shah, F.M., Hridoy, A.K. & Farid, D.M. (2015). Application of Machine Learning Approaches in Intrusion Detection System: A Survey. *(IJARAI) International Journal of Advanced Research in Artificial Intelligence*. 4 (3). pp. 9–18.

Harrington, P. (2012). *Machine Learning in Action*. Shelter Island, NY: Manning Publications Co.

Heatherlench (2010). *Chapter 10: Linear Regression*. [Online]. 2010. Available from: http://heatherlench.com/wp-content/uploads/2010/01/10-regression.pdf. [Accessed: 23 June 2016].

Hertzmann, A. & Fleet, D. (2010). *Machine Learning and Data Mining Lecture Notes*. [Online]. University of Toronto. Available from: https://www.dgp.toronto.edu/~hertzman/411notes.pdf.

Van Hoorn, A., Hasselbring, W., Waller, J., Ehlers, J., Frey, S. & Kieselhorst, D. (2009). *Continuous Monitoring of Software Services: Design and Application of the Kieker Framework*. [Online]. Germany. Available from: http://www.informatik.uni-oldenburg.de/~voorn/bib/avanhoorn/091203-KiekerKiel-postfinal.pdf.

Hsu, C., Chan, C. & Lin, C.-J. (2003). *A Practical Guide to Support Vector Classification*. [Online]. Available from: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

Huber, N., Quast, M. von, Hauck, M. & Kounev, S. (2011). Evaluating and Modeling Virtualization Performance Overhead for Cloud Environments. In: *CLOSER 2011 - Proceedings of the 1st International Conference on Cloud Computing and Services Science*. 2011, Netherlands.

Ibrahim, A.S., Hamlyn-Harris, J. & Grundy, J. (2010). Emerging Security Challenges of Cloud Virtual Infrastructure. In: *In Proceedings of APSEC 2010 Cloud Workshop*. [Online]. 2010, Sydney, Australia, pp. 1–6. Available from: https://www.cs.auckland.ac.nz/~john-g/papers/cloud2010_2.pdf.

Iosup, A., Prodan, R. & Epema, D. (2013). *IaaS Cloud Benchmarking: Approaches, Challenges, and Experience*. [Online]. NY, USA. Available from: http://www.ds.ewi.tudelft.nl/~iosup/cloud-benchmarking12mtags.pdf.

Ipek, E., McKee, S.A., Caruana, R., de Supinski, B.R. & Schulz, M. (2006). Efficiently exploring architectural design spaces via predictive modeling. *ASPLOS*. [Online]. 41 (11). pp. 195–206. Available from: http://dl.acm.org/citation.cfm?id=1168918.1168882.

Ishikawa, A., Endo, S. & Shiratori, T. (2010). *Treatment of Outliers in Business Surveys: The Case of Short-term Economic Survey of Enterprises in Japan (Tankan)*. Bank of Japan Working Paper Series. [Online]. Tokyo. Available from: https://www.boj.or.jp/en/research/wps_rev/wps_2010/data/wp10e08.pdf.

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. London: Springer.

Joshua, A. & Ogwueleka, F. (2013). Cloud Computing with Related Enabling Technologies. *International Journal of Cloud Computing and Services Science (IJ-CLOSER)*. 2 (1). pp. 40–49.

Kang, H., Zhu, X. & Wong, J.L. (2012). *DAPA: Diagnosing Application Performance Anomalies for Virtualized Infrastructures*. [Online]. 2012. Available from: https://www.usenix.org/conference/hot-ice12/workshop-program/presentation/kang. [Accessed: 7 September 2016].

Khan, S., Xekalakis, P., Cavazos, J. & Cintra, M. (2007). Using PredictiveModeling for Cross-Program Design Space Exploration in Multicore Systems. In: *PACT '07 Proceedings of the 16th International Conference on Parallel Architecture and Compilation Techniques*. 2007, Washington, DC, USA: IEEE Computer Society, pp. 327–338.

Khanghahi, N. & Ravanmehr, R. (2013). Cloud Computing Performance Evaluation: Issues and Challenges. *International Journal on Cloud Computing: Services and Architecture*. [Online]. 3 (5). pp. 29–41. Available from: http://www.airccse.org/journal/ijccsa/papers/3513ijccsa03.pdf.

Kundu, S. (2013). *Improving Resource Management in Virtualized Data Centers using Application Performance Models*. [Online]. Florida International University. Available from: http://digitalcommons.fiu.edu/cgi/viewcontent.cgi?article=1990&context=etd.

Kundu, S., Rangaswami, R., Dutta, K. & Zhao, M. (2010). Application performance modeling in a virtualized environment. In: *HPCA 16 2010 The Sixteenth International Symposium on HighPerformance Computer Architecture*. [Online]. 2010, Bangalore, India, pp. 1–10. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5463058.

Lee, B.C.B.C. & Brooks, D.M.D.M. (2006). Accurate and efficient regression modeling for microarchitectural performance and power prediction. *ACM SIGARCH Computer Architecture News*. [Online]. 34 (5). pp. 185. Available from: http://portal.acm.org/citation.cfm?id=1168881.

Li, H. & Liu, D. (2010). Research on intelligent intrusion prevention system based on Snort. In: *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering (Volume:1)*. 2010, Changchun: IEEE, pp. 251 – 253.

Luts, J., Ojeda, F., Van de Plas, R., De Moor, B., Van Huffel, S. & Suykens, J.A.K. (2010). A tutorial on support vector machine-based methods for classification problems in chemometrics. *Analytica Chimica Acta*. [Online]. 665 (2). pp. 129–145. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0003267010003132.

Matsunaga, A. & Fortes, J. a. B. (2010). On the Use of Machine Learning to Predict the Time and Resources Consumed by Applications. In: *10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid)*. [Online]. 2010, pp. 495–504. Available from: http://homepages.cs.ncl.ac.uk/paolo.missier/doc/Matsunaga10.pdf.

McCullough, J.C., Agarwal, Y., Chandrashekar, J., Kuppuswamy, S., Snoeren, A.C. & Gupta, R.K. (2011). Evaluating the effectiveness of model-based power characterization. In: *Proceeding USENIXATC'11 Proceedings of the 2011 USENIX conference on USENIX annual technical conference*. 2011, Berkeley, CA, USA: USENIX Association, pp. 12–12.

Meinshausen, N. & Kunsch, H. (2016). *Regression*. [Online]. Available from: https://stat.ethz.ch/education/semesters/ss2016/regression/Regression.pdf.

Menascé, D.A., Almeida, V.A.F. & Dowdy, L. (1994). *Capacity Planning and Performance Modeling: From Mainframes to Client-server Systems*. Upper Saddle River, New Jersey: PTR Prentice Hall.

Mozafari, B., Curino, C. & Madden, S. (2013). *Resource and Performance Prediction for Building a Next Generation Database Cloud*. [Online]. Available from: http://db.csail.mit.edu/pubs/cidr-rel-cloud.pdf.

Nirmala, A.. & Sridara, R. (2015). Towards Performance Improvement of Cloud Applications under Virtualized Environment using PSO based K-means++ Algorithm. *International Journal of Engineering and Technolog*. 7 (5). pp. 1557–1563.

Okanovic, D. & Vidakovic, M. (n.d.). *Software Performance Prediction Using Linear Regression*. [Online]. Available from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.460.3994&rep=rep1&type=pdf. [Accessed: 7 September 2016].

Orand, B. (2011). *Foundations of IT Service Management with ITIL 2011 : ITIL Foundations Course in a Book*. 2nd Ed. J. Villarreal (ed.). [Online]. Black & White illustrations. Available from: https://www.bookdepository.com/Foundations-IT-Service-Management-with-ITIL-2011-Brady-Orand/9781466231320.

Paliwal, S. (2014). *Performance Challenges in Cloud Computing*. [Online]. Available from: https://www.cmg.org/wp-content/uploads/2014/03/1-Paliwal-Performance-Challenges-in-Cloud-Computing.pdf.

Pawar, S. & Singh, S. (2015). Performance Comparison of VMware and Xen Hypervisor on Guest OS. *International Journal of Innovative Computer Science & Engineering*. 2 (3). pp. 56–60.

Payne, R. (2012). *A Guide to Regression, Nonlinear and Generalized Linear Models in GenStat*. [Online]. United Kingdom: VSN International. Available from: http://www.vsni.co.uk/downloads/genstat/release15/doc/RegressGuide.pdf.

Perez-Botero, D., Szefer, J. & Lee, R.B. (2013). Characterizing Hypervisor Vulnerabilities in Cloud Computing Servers. In: *CloudComputing'13*. [Online]. 2013, Hangzhou, China: ACM Press, pp. 1–8. Available from: http://caslab.eng.yale.edu/people/jakub/papers/scc2013.pdf.

Preston, M. (2013). *Troubleshooting vSphere Storage*. [Online]. Mumbai. Available from: http://ftp.factor.lg.ua/books/PP.Troubleshooting.vSphere.Storage.Nov.2013.pdf.

Pu, X., Liu, L., Mei, Y., Sivathanu, S., Koh, Y. & Pu, C. (2010). Understanding Performance Interference of I/O Workload in Virtualized Cloud Environments. In: *2010 IEEE 3rd International Conference on Cloud Computing*. [Online]. July 2010, IEEE, pp. 51–58. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5558012.

Rao, J., Bu, X., Xu, C.-Z., Wang, L. & Yin, G. (2009). VCONF. In: *Proceedings of the 6th international conference on Autonomic computing - ICAC '09*. [Online]. 2009, pp. 137–146. Available from: http://portal.acm.org/citation.cfm?doid=1555228.1555263.

Rasmussen, C.E. & Williams, C.K.I. (2006). *Gaussian Processes for Machine Learning*. [Online]. Cambridge, Massachusetts, United States: MIT Press. Available from: http://www.gaussianprocess.org/gpml/chapters/RW2.pdf.

Rowntree, D. (2003). *Statistics Without Tears: A Primer for Non-Mathematicians*. Pearson.

Scholkopf, B. & Smola, A.J. (2002). *Learning with Kernels*. Cambridge, MA: MIT Press.

Scholkopf, B., Tsuda, K. & Vert, J.-P. (2004). *Kernel Methods in Computational Biology*. London, U.K: MIT Press.

Shalev-Shwartz, S. & Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. [Online]. United Kingdom: Cambridge University Press. Available from: http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding-machine-learning-theory-algorithms.pdf.

Shawe-Taylor, J. & Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge, UK: Cambridge UP.

Sheshasaayee, A. & Lakshmi, J.V.N. (2014). An Analysis of Linear Regression Algorithm implemented on Hadoop using Machine Learning Techniques. *International Journal of Innovative Research and Studies*. 3 (7). pp. 109–118.

Smola, A. & Vishwanathan, S.V.N. (2008). *Introduction to Machine Learning*. United Kingdom: Cambridge Univeristy Press.

Stewart, C., Kelly, T., Zhang, A. & Shen, K. (2008). A dollar from 15 cents: cross-platform management for internet services. In: *Proceeding ATC'08 USENIX 2008 Annual Technical Conference*. 2008, Berkeley, CA, USA: USENIX Association, pp. 199–212.

Tan, Y., Nguyen, H., Shen, Z., Gu, X., Venkatramani, C. & Rajan, D. (2012). PREPARE: Predictive Performance Anomaly Prevention for Virtualized Cloud Systems. In: *2012 IEEE 32nd International Conference on Distributed Computing Systems*. [Online]. June 2012, IEEE, pp. 285–294. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6258001.

The Nuffield Foundation (2008). *Using Excel for inferential statistics*. [Online]. England. Available from: http://www.nuffieldfoundation.org/sites/default/files/excel_inferential_stats.pdf.

Uttamchandani, S., Yin, L., Alvarez, G.A., Palmer, J. & Agha, G. (2005). Chameleon: a self-evolving, fully-adaptive resource arbitrator for storage systems. In: *Proc. of USENIX*

Annual Technical Conference. [Online]. 2005, pp. 75–88. Available from: https://www.usenix.org/legacy/event/usenix05/tech/general/full_papers/uttamchandani/uttamchandani.pdf.

Vaghani, S.B. (2010). Virtual Machine File System. *ACM SIGOPS Operating Systems Review*. 44 (4). pp. 57–70.

Vazquez, C., Krishnan, R. & John, E. (2014). Cloud Computing Benchmarking: A Survey. In: *Proceesings of International Conference on Grid & Cloud Computing and Applications*. 2014, Las Vegas, NV.

VMware (2007). *Introduction to VMware Infrastructure*. [Online]. Palo Alto, CA: VMware, Inc. Available from: https://www.vmware.com/pdf/vi3_35/esx_3/r35/vi3_35_25_intro_vi.pdf.

VMware (2011). *VMware vSphere Basics*. [Online]. Palo Alto, CA. Available from: https://pubs.vmware.com/vsphere-50/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-50-basics-guide.pdf.

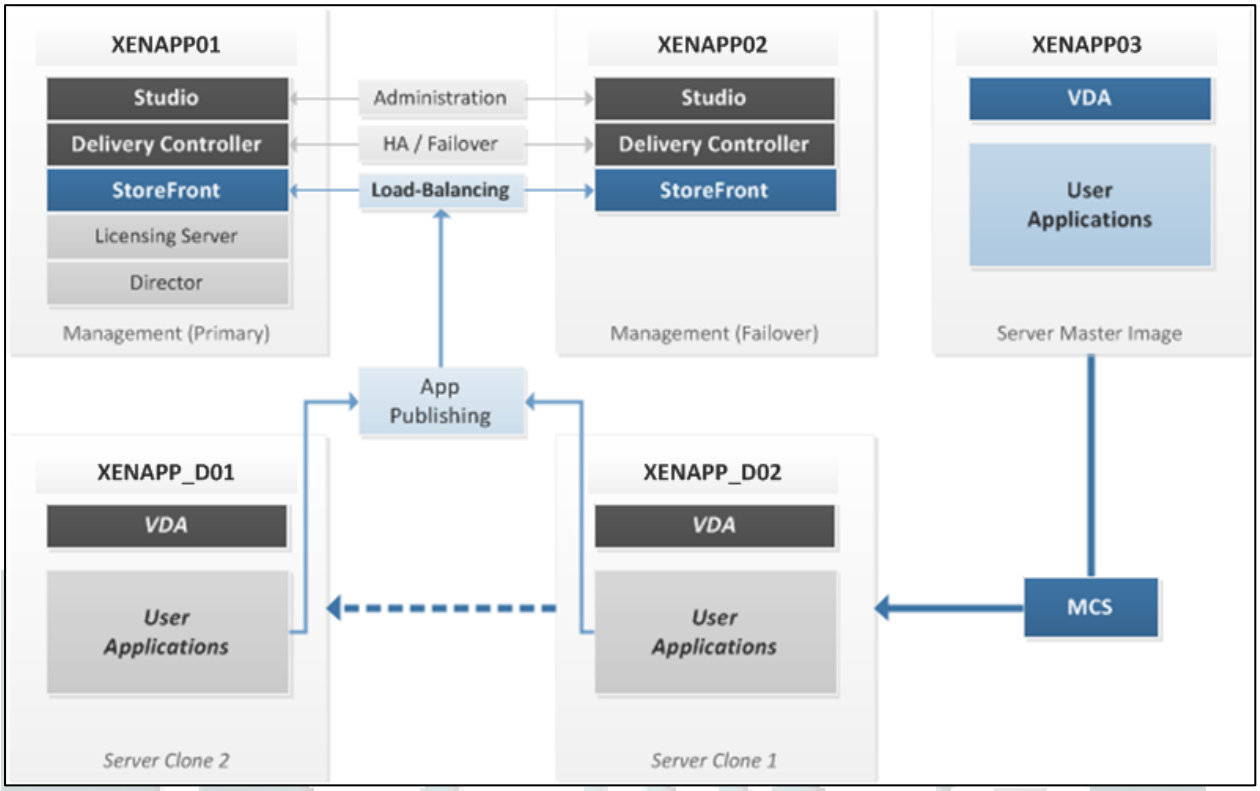
Williams, R. (2016). *Outliers*. [Online]. Available from: https://www3.nd.edu/~rwilliam/stats2/l24.pdf.

Xu, J., Zhao, M., Fortes, J., Carpenter, R. & Yousif, M. (2008). Autonomic resource management in virtualized data centers using fuzzy logic-based approaches. *Cluster Computing*. 11 (3). pp. 213–227.

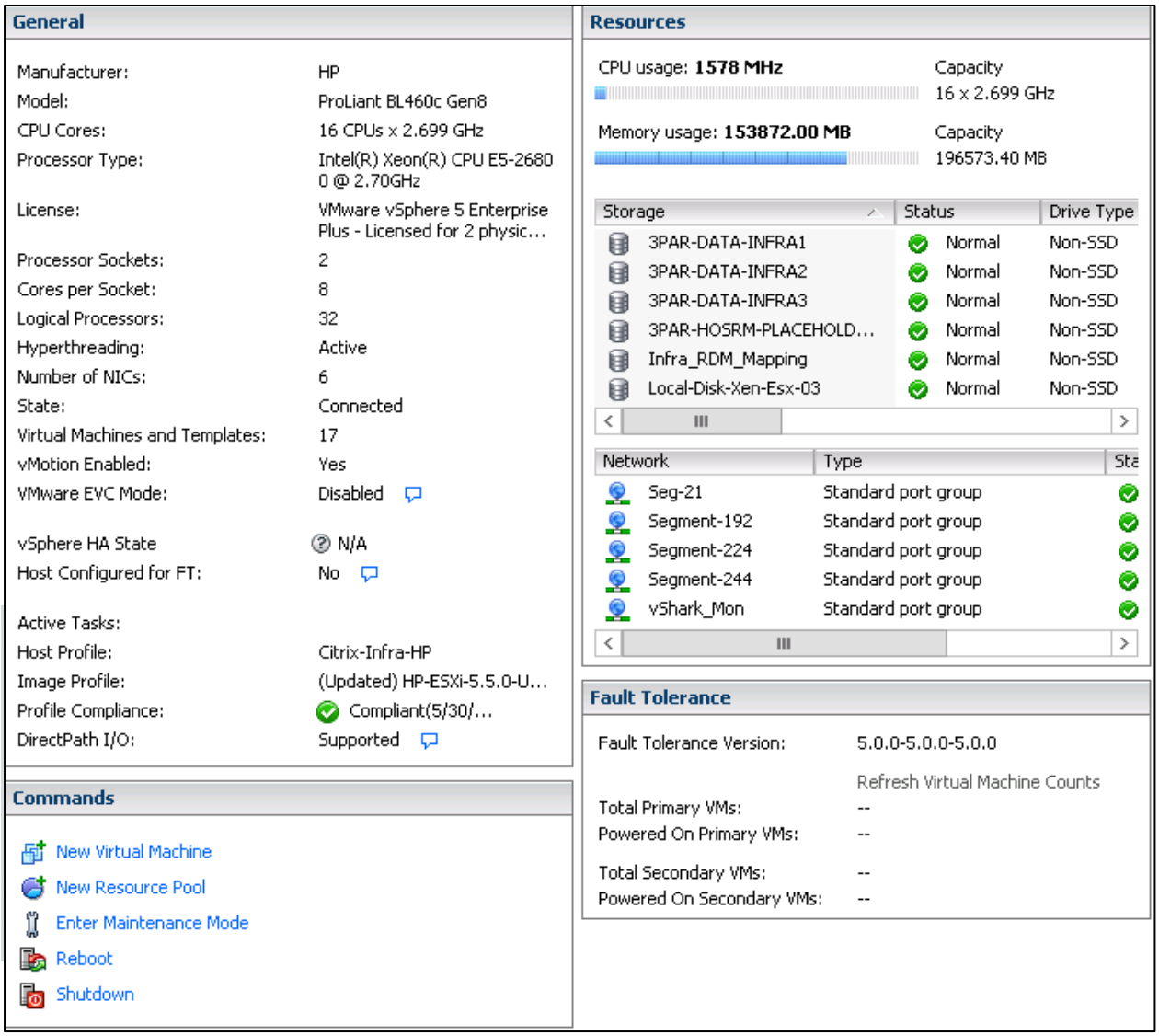
Zia, A. & Khan, M.N.A. (2012). Identifying Key Challenges in Performance Issues in Cloud Computing. *International Journal of Modern Education and Computer Science*. [Online]. 4 (10). pp. 59–68. Available from: http://www.mecs-press.org/ijmecs/ijmecs-v4-n10/v4n10-8.html.

APPENDICES

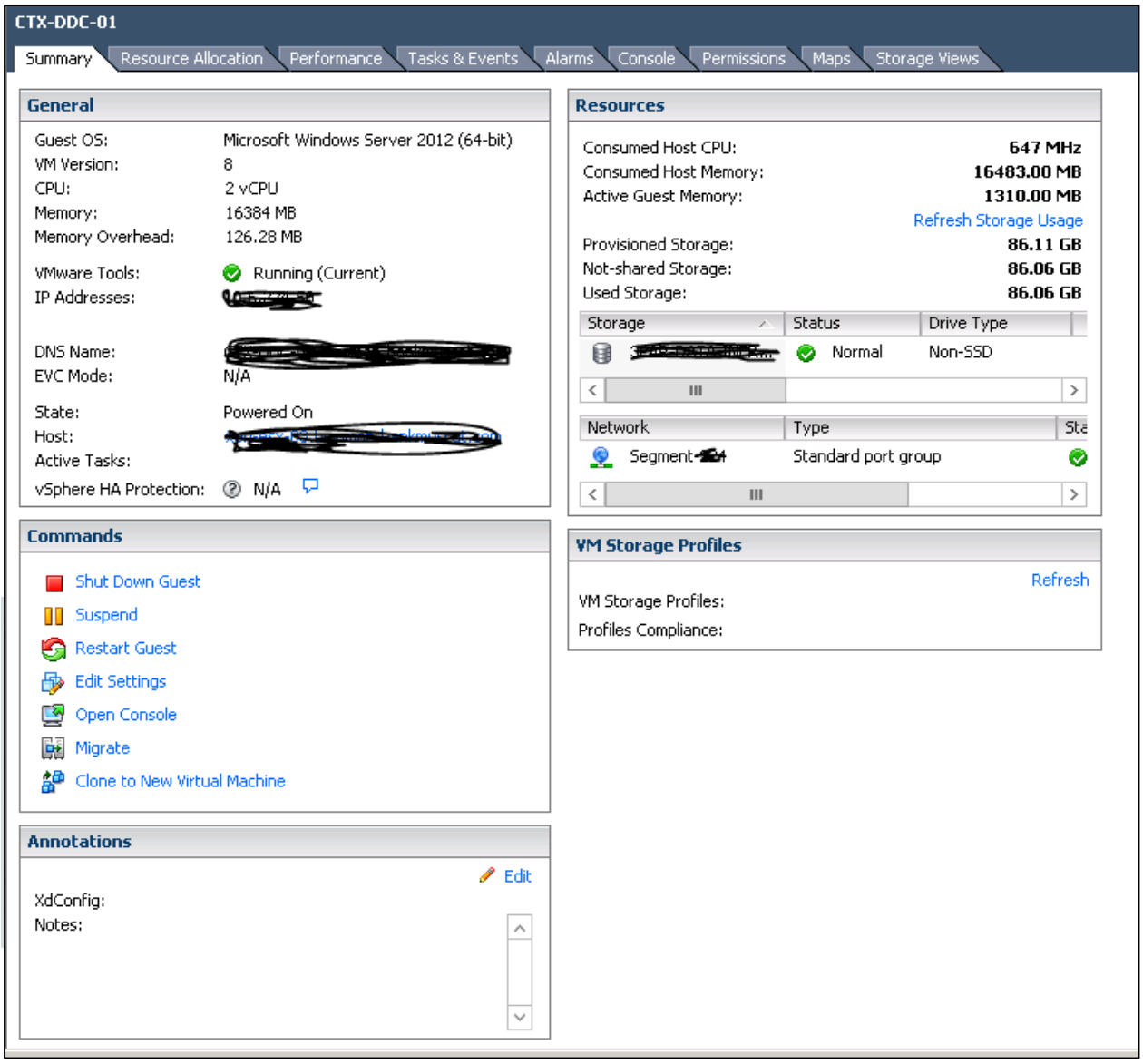
Appendix 1: High Level Solution Diagram



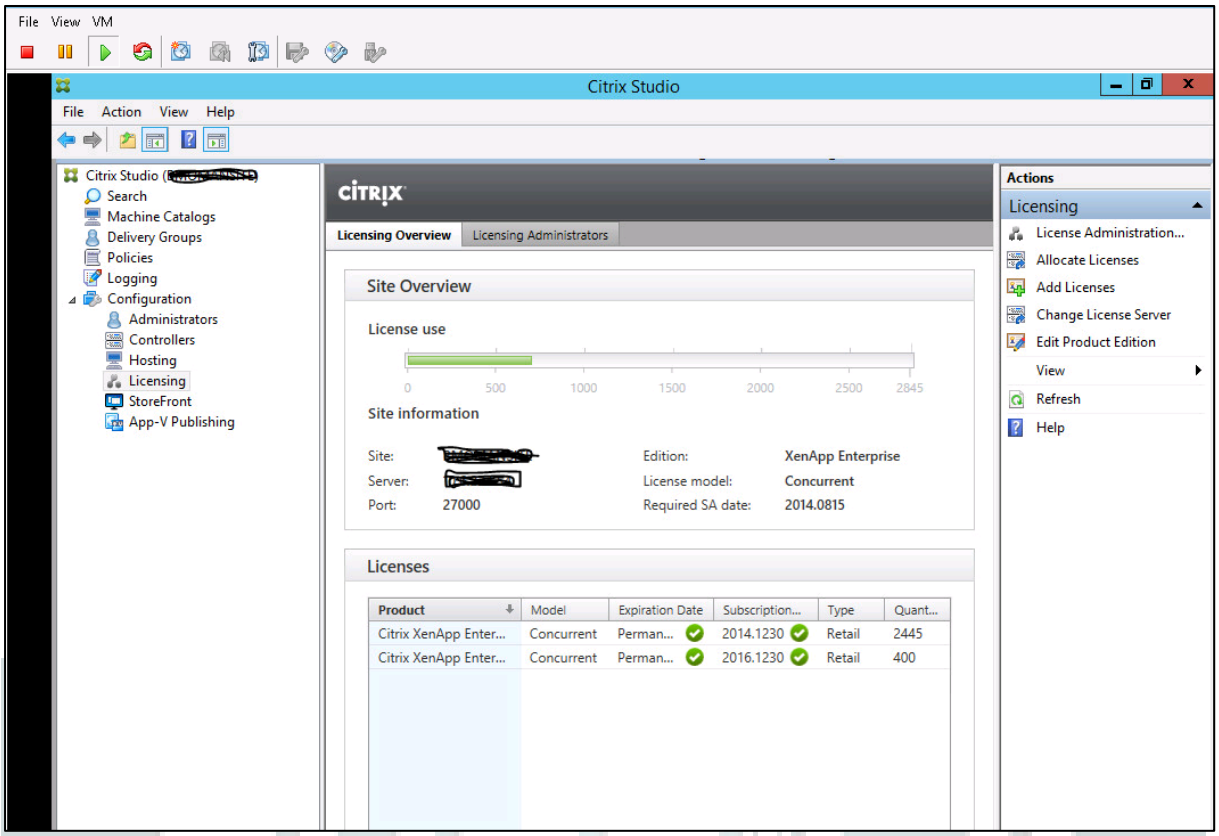
Appendix 2: Hardware Configuration



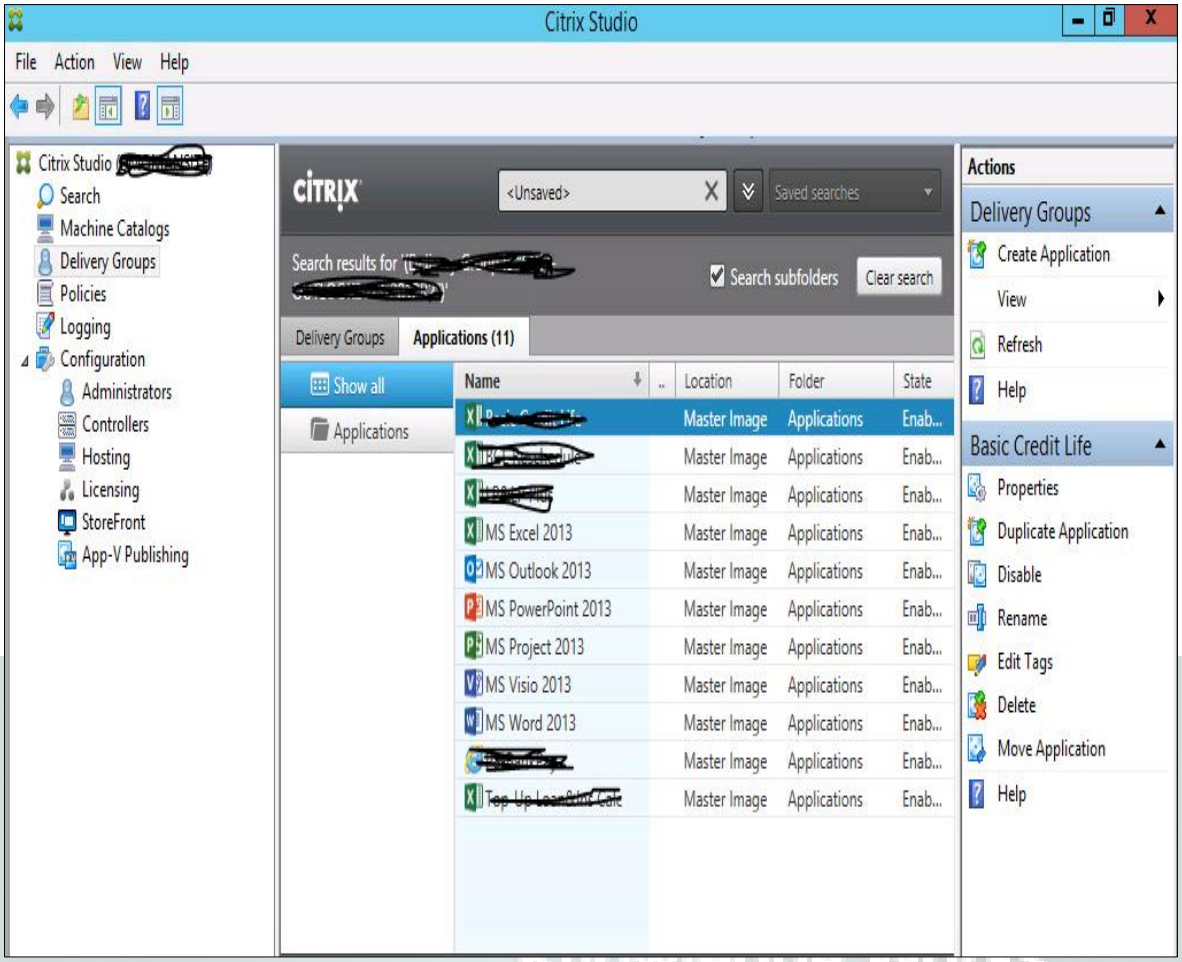
Appendix 3: XenApp Server Configuration



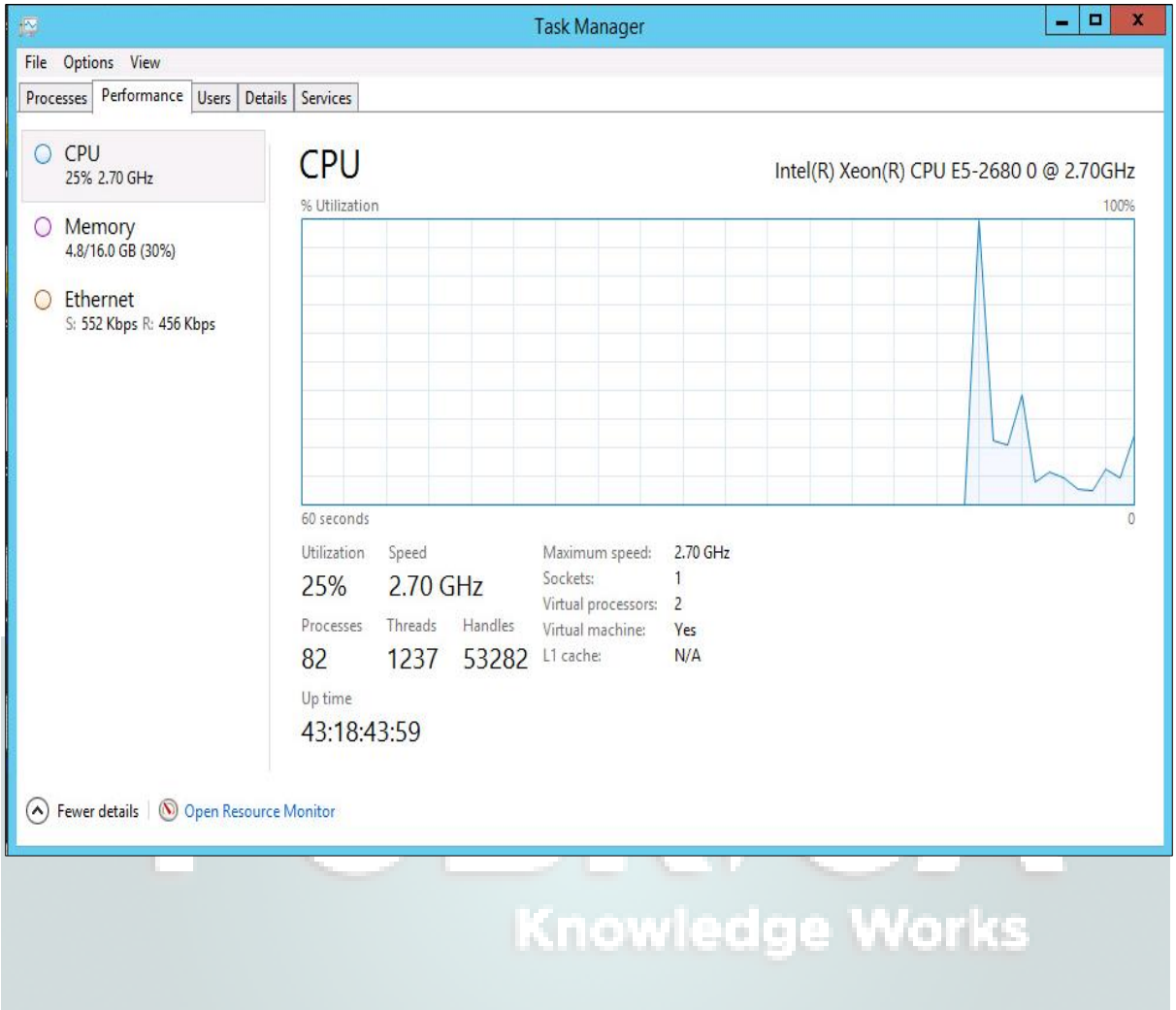
Appendix 4: XenApp Management Console (Apps Published)



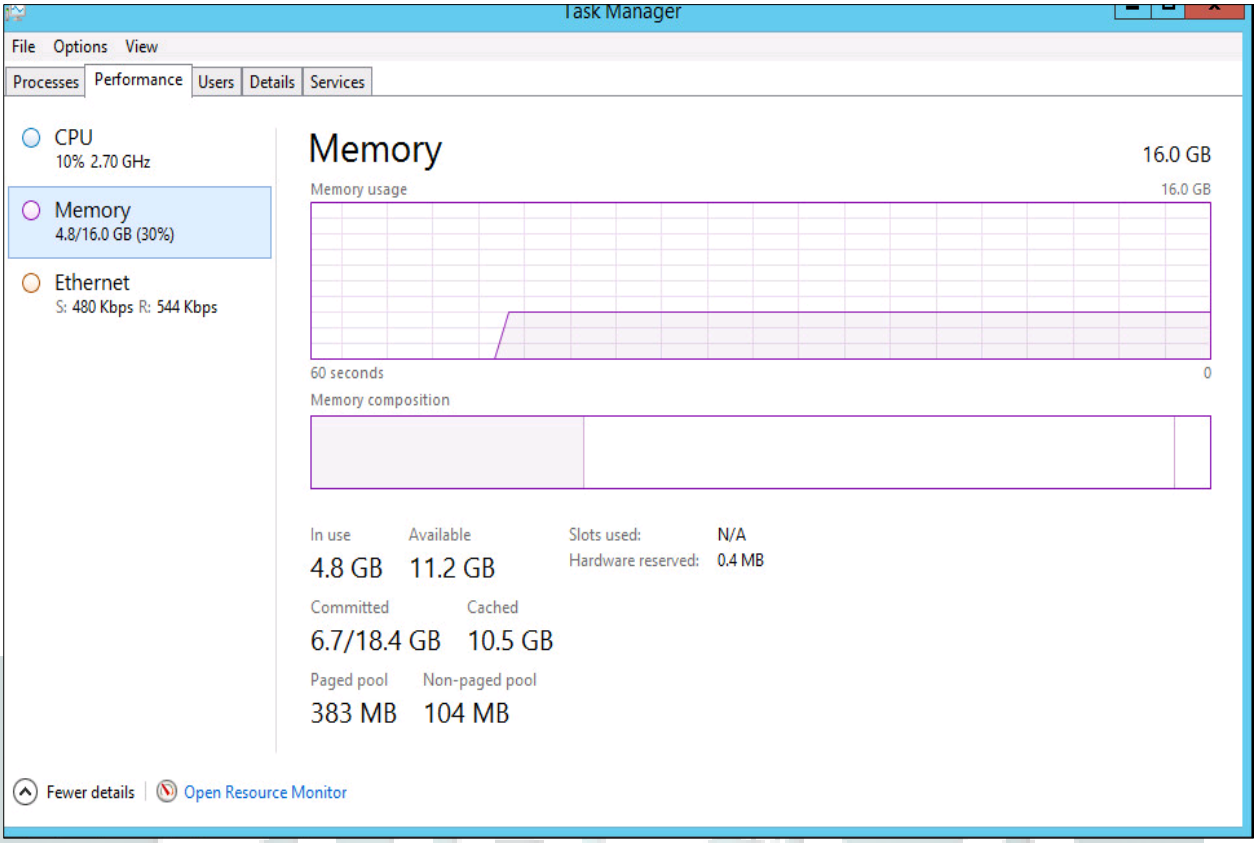
Appendix 5: Citrix Application Console



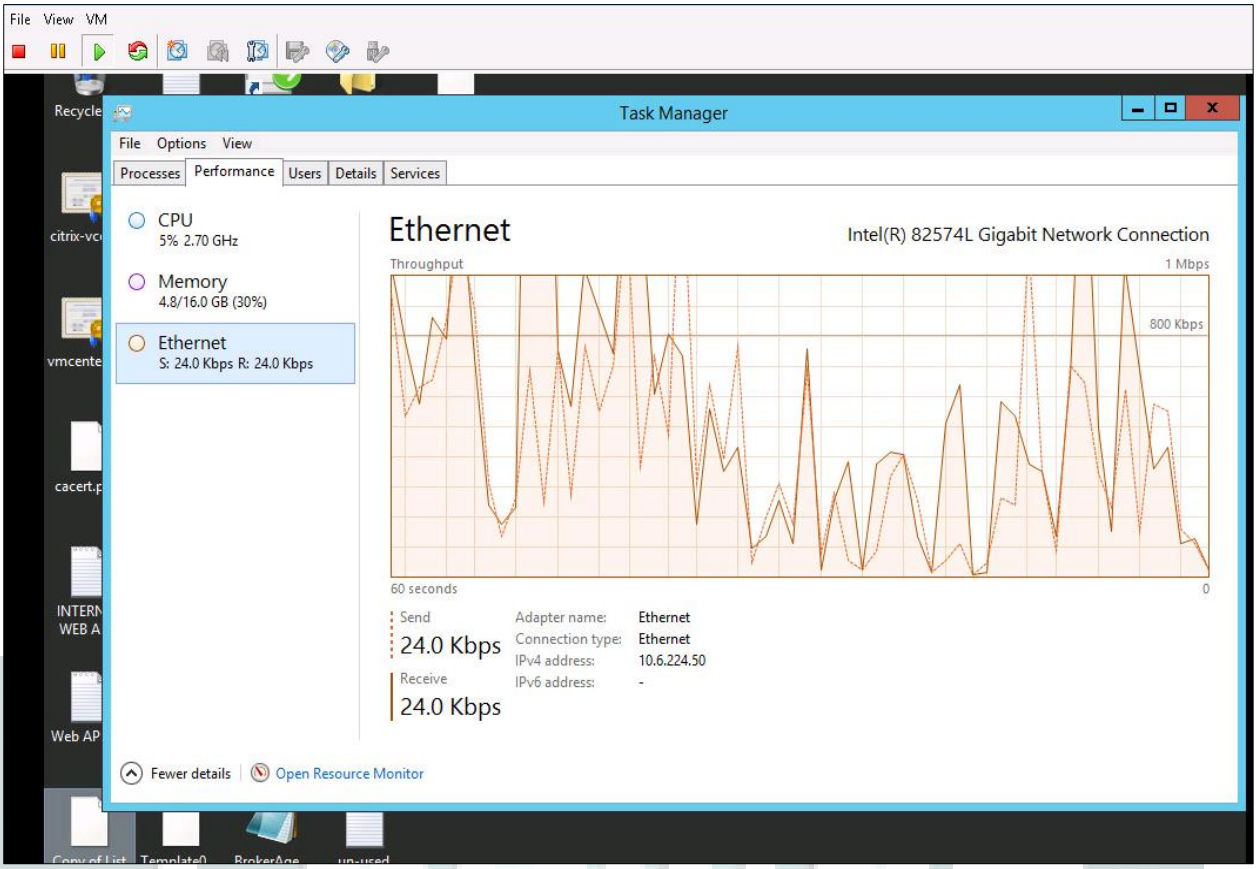
Appendix 6: CPU Utilization Graph



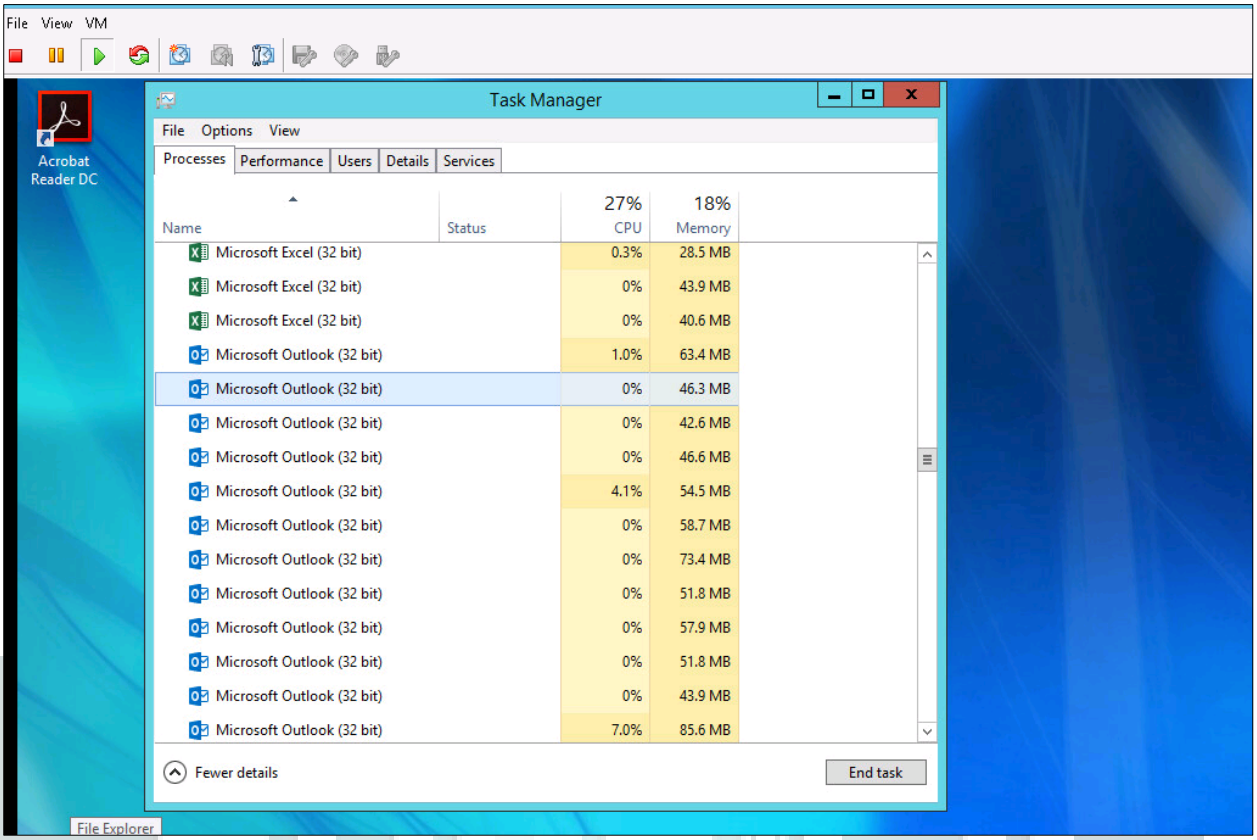
Appendix 7: Memory Utilization Graph



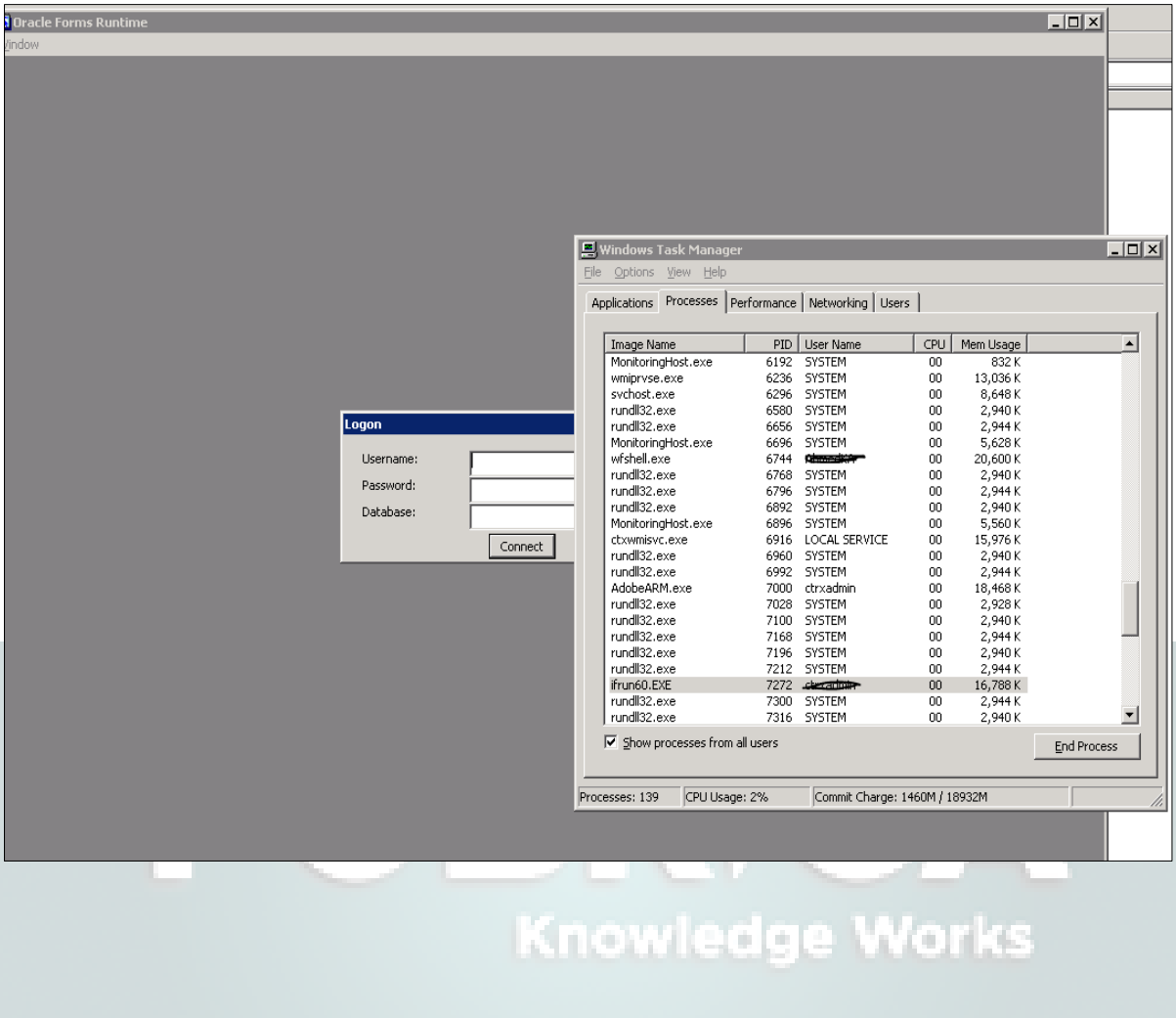
Appendix 8: Ethernet Utilization Graph



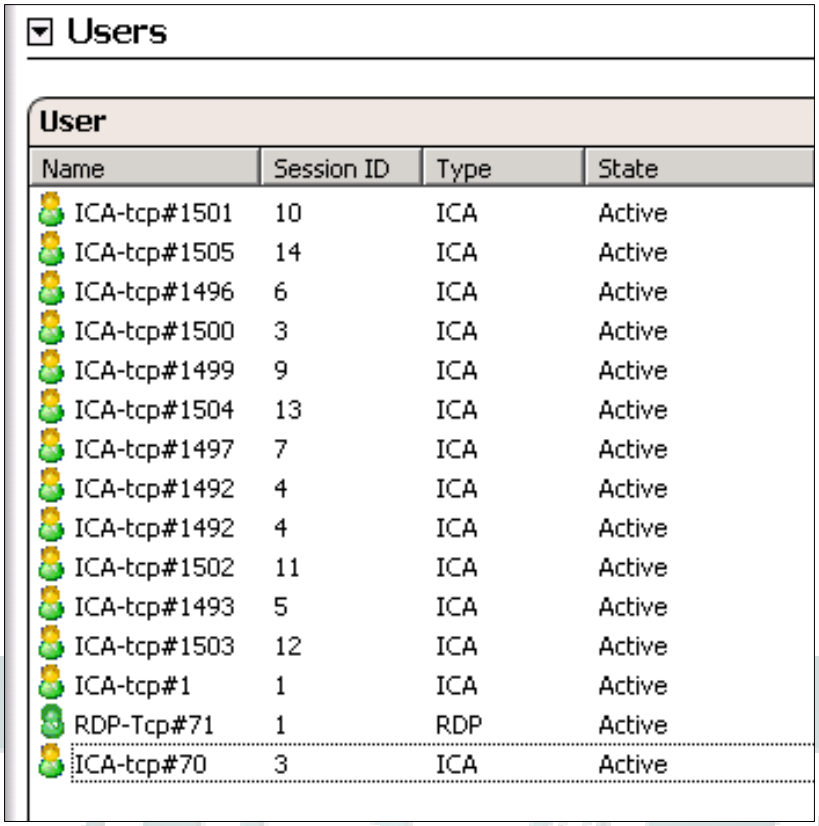
Appendix 9: Outlook Application Utilization CPU & Memory



Appendix 10: Oracle HRMS CPU 7 Memory Utilization



Appendix 11: User Session Screenshot



Users

User

Name	Session ID	Type	State
ICA-tcp#1501	10	ICA	Active
ICA-tcp#1505	14	ICA	Active
ICA-tcp#1496	6	ICA	Active
ICA-tcp#1500	3	ICA	Active
ICA-tcp#1499	9	ICA	Active
ICA-tcp#1504	13	ICA	Active
ICA-tcp#1497	7	ICA	Active
ICA-tcp#1492	4	ICA	Active
ICA-tcp#1492	4	ICA	Active
ICA-tcp#1502	11	ICA	Active
ICA-tcp#1493	5	ICA	Active
ICA-tcp#1503	12	ICA	Active
ICA-tcp#1	1	ICA	Active
RDP-Tcp#71	1	RDP	Active
ICA-tcp#70	3	ICA	Active