Cloud costs forecasting

Towards predictable IT costs

25-10-2012

Centric IT Solutions
K.L. van Ingen

CLOUD COST FORECASTING

Master thesis # IKU- N/A

Student#: 3561593

Utrecht University

AUTHOR

Kevin Lennard van Ingen

Master Business Informatics

Utrecht University

kevin.van.ingen@gmail.com

SUPERVISORS

dr. S.R.L. Jansen	prof. dr. S. Brinkkemper	L. Blom
Utrecht University	Utrecht University	Centric IT Solutions
Department of Information and Computing Sciences	Department of Information and Computing Sciences	Research and development
S.Jansen@cs.uu.nl	S.Brinkkemper@uu.nl	Leen.Blom@centric.eu
1st supervisor	2nd supervisor	Company supervisor

SUPPORTING ORGANIZATION:

Centric IT Solutions, Gouda, The Netherlands

Abstract

Although scalability has been one of cloud computing' greatest merits, it leaves the consumer and the operator with uncertainty about future costs and corresponding capacity requirements. The IT industry is partially transitioning towards IT as a utility, which impacts both price and costs (Truong Huu et al. 2010). This uncertainty could prove to be too difficult to conquer for some organizations and holds back acceptance of cloud computing technology. Several cloud infrastructure providers have recently introduced cloud cost calculators of some sort to come up with an estimation of monthly costs (Corporation 2012; IBM corporation 2012). An important assumption made by these cost calculators is that customers are required to know how much they will consume, which. Until now there has not been a significant contribution in literature that addresses the forecasting of costs for cloud services.

This research adopts a design science research strategy (Hevner 2007). An IT artifact is created that is able to forecast cloud costs. The current state-of-the-art in the understanding of cloud costs in published literature is researched using a structured literature review (Kitchenham 2007). The majority of researchers include cost savings as a rationale for their research. Several quantitative forecasting techniques have been researched according to the classification of Armstrong (Armstrong and Green 2005). A decision matrix was created to aid the selection of the proper forecasting technique. A technical solution is created that forecasts infrastructure level usage on CPU, memory, network and storage which combined create a significant portion of the cloud costs. A controlled experiment is used to test the application of forecasting theory in the context of cloud costs. An expert review with five cross domain experts is performed to validate – and help interpret – research findings. After having conducted this research, it is concluded that extrapolation forecasting on cloud costs has great potential for cloud platform operators.

In this work I present a literature review on cloud costs, a comparison of the application of forecasting techniques in the context of cloud computing costs, a controlled experiment with a forecasting technique on a real-life case, a validation by means of an expert review, and research directions for future research.

Preface

With this thesis report I conclude my two year master program Business Informatics at Utrecht University. In two years, an idea evolved in my head about the context and topic of my master thesis project. I was determined to look for a contribution to both academia and practice, in a context in which I can properly apply my knowledge from both theory and practice. It was clear to me that I wanted the support and involvement of experts and experienced individuals from practice. This includes upper management, as well as the people that get their hands dirty. It is my personal opinion that science is the servant of practice. Both are intertwined, especially in the field of information science. However, it is important to acknowledge that both can have different purposes. My personal interest in information economics, business, as well as a profound interest in understanding information technology brought me on the subject of cloud technology.

This thesis is the result of seven months of research, which is performed at a large IT firm called Centric IT Solutions in The Netherlands. A company with substantial experience in a broad spectrum of IS/IT related products, covering infrastructure, software development and testing, and IT services, servicing both public and commercial clientele. Centric provided me a practical problem, in which I would be able to fuel my technical interest in cloud infrastructure and software development, combined with an understanding of business and economics. The diversity of this problem proved to be a challenge in both scope as well as scale. It involved a multi-disciplinary research project, where I found myself to be much like an equilibrist. This has proven to be an enormous intellectual challenge, which was both fulfilling and defiant at times.

I am sincerely grateful for everyone who assisted me during this thesis project and guided the process. I would like to thank my supervisor dr. Jansen who guided me throughout this project, especially on appreciating and respecting difference between science and practice. Your passion for the field is both inspiring and contagious. Additionally, I would like to express my gratitude to my second supervisor prof. Dr. S. Brinkkemper, for the feedback provided on my thesis and equally important for fueling my passion for science the past two years. Equally important is the support of Centic IT solutions for the resources which were made available to me. This involved numerous people, for whom unfortunately, this page is too short to thank separately. Your experience and critical minds have been invaluable to this research! You all received me with patience, respect, and open mindedness. I would like to express my gratitude to Leen Blom, who mentored me throughout this project. You always encouraged me to keep an open mind and look beyond the constraints of here and now. I admire your unprejudiced and patient character. Furthermore, I would like to express gratitude to Atte Visser, who supported me during this project by being an invigorating conversational sparring partner. I would also like to thank my current employer Rotterdam University and my colleagues, who sponsored me during this research in both time and by being flexible in mind on the busiest moments.

Finally, I would like to thank my family, friends, colleagues and especially my girlfriend Marianne, for their enduring encouragement, patience, enthusiasm and support.

Gouda, Oktober 2012

Kevin van Ingen

Table of contents

Chapter 1. Introduction

1.1 Problem statement

Currently, the shift towards cloud computing together with pay-per-use calculation, impacts the way organizations pay for IT. One of the major drivers behind cloud adoption is potential cost savings (M. D. D. Assunção, Costanzo, and Buyya 2009; Deelman et al. 2008; IDC 2009). Although, these terms make great marketing materials, in reality, organizations that have systems that are currently running on premises solutions, moving to the cloud comes with great uncertainty of costs. In a traditional – investment intensive – IT environment, organizations make investments in hardware, software customization, licenses, implementation costs and have operational maintenance costs. Although IS/IT costs have been a complex accounting concept in the past, most organizations have a good idea about IT costs on a five year term, as they are depreciating their investments. Cloud computing comprises a change from a high capital expenditures (*CAPEX*) situation towards a high operational expenditures (*OPEX*) situation, which causes great uncertainty for clients of cloud operators as costs are hard to predict, even for technologists.

At the moment, many organizations are transitioning, to an as-a-service type cost calculation, based on the presumption that it will decrease costs without insights what future costs might be. In the traditional high CAPEX situation systems were sized for peak demand, which in practices means systems are overprovisioned up to 90 % (Greenberg et al. 2008). Due to a more efficient use of available resources, which lead to a decrease in overprovisioning, organizations have to find ways to perform more efficient infrastructure *sizing* (Truong Huu et al. 2010). In this respect *sizing* is the fitting of the software layer on the platform layer and the platform layer on the infrastructure layer. Cloud technology offers the flexibility to size for current demand. However, whenever system load increases, somebody still has to pay the bill. It can either be the operator or the client. The business model behind the cloud product eventually determines who takes the 'risk', and thus the bill for the service. It can either be the platform operator or the client itself.

Whether or not cloud is cost effective is still up for debate, as technologies for metering still seem inadequate (Garfinkel 2007; Ibrahim, He, and Jin 2011; H. Wang et al. 2010). Moreover, these authors argue that metering is difficult on the system level, even for highly technical people when there is in-depth knowledge about the software that runs on these servers, let alone when the software stack that runs on them is compiled out of proprietary products. In addition, it would be nearly impossible at the moment for the less technically skilled professionals to develop a sensible notion about costs in an a priori situation. A recent journal publication stating a research agenda on cloud computing acknowledges the difficulty to related service requirements to low-level requirements – like CPU or memory capacity (Q. Zhang, Cheng, and Boutaba 2010). Authors have highlighted that a manner to explain costs in terms which are understandable and measureable for a customer is necessary for accountability purposes but also for *fair play* in multi-tenant environments (Ibrahim, He, and Jin 2011; Mihoob, Molina-Jimenez, and

Shrivastava 2010). From the perspective of a cloud operator, forecasting means having detailed knowledge in the why and how costs occur on a customer level. This knowledge is a valuable asset, which makes a potential business case for problem at hand.

This master thesis results in knowledge relevant to the scientific field as practice. Disciplinary knowledge will be communicated through a master thesis report and a derivative scholarly paper. Although the field of cloud computing has received much attention the last decade with an emphasis on the advantages of scalability, the downside, being uncertainty – especially for migrating customers – has hitherto been unaddressed. The research addresses a problem that will break some uncertainty barriers that exists in the field of cloud computing today, moving the area of cloud computing forwards. The thesis report provides both theoretical knowledge and practical knowledge about how forecasting techniques can be employed to predict costs for cloud solutions. I define the following problem statement:

The uncertainty of future costs for pay-per-use cloud services hinders adoption of cloud services in organizations.

1.2 Research question

To address the problem statement, insights in the relationship between the pre-cloud and post-cloud situation is necessary. This thesis project is aimed at answering the question and related sub questions below.

How can cost forecasting techniques be employed for the prediction of variable cloud costs?

Three sub questions are derived from the main research question:

a. *What variables determine cloud costs and how do they relate to each other?*
b. *Which cost forecasting techniques exists, and how can they be used for the prediction of cloud costs?*
c. *How can cloud cost variables and a forecasting technique be integrated in a technical solutions that predicts cloud costs?*

1.3 Research method

In order to answer the research question, a method called *design science research* (DSR) is adopted, which has become a common practice over the recent years in the field of information systems research (Hevner et al. 2004; S.T. March and Storey 2008). The primary focus of a DSR project is the creation of an IT artifact (Gregory and Wayne 2010). This focus fits the research project and question nicely. The process of DSR involves the search for a relevant problem, the design of an IT artifact – a technical forecasting solutions – and its ex ante and ex post evaluation.

Critics of design science argue about the necessity of a formal articulation of a process and mental model of the research under consideration (Peffers et al. 2006). The research model is a compilation of the efforts of several publications that have contributed to a communicable research model (Nunamaker, M. Chen, and Purdin 1991;

Rossi and Sein 2003; Takeda and Veerkamp 1990). The research is articulated through the application of a model as is shown in Figure 1.

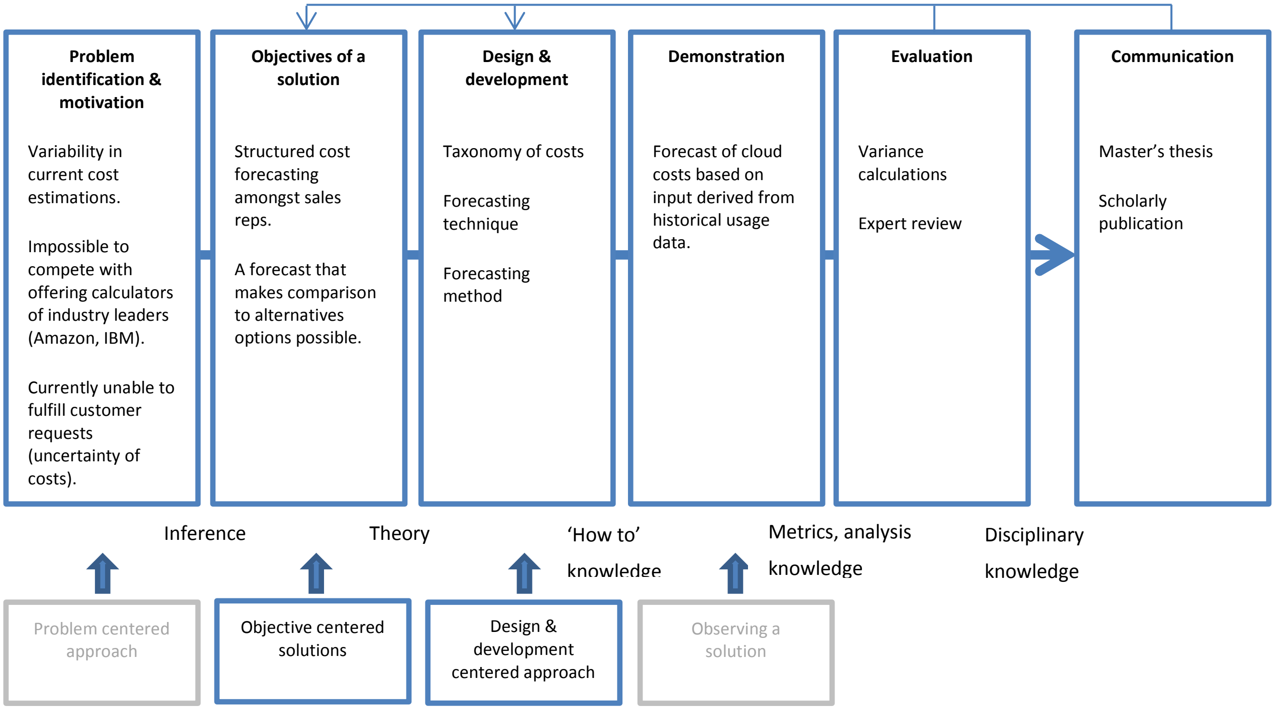

Problem identification & motivation
Variability in current cost estimations.
Impossible to compete with offering calculators of industry leaders (Amazon, IBM).
Currently unable to fulfill customer requests (uncertainty of costs).
Objectives of a solution
Structured cost forecasting amongst sales reps.
A forecast that makes comparison to alternatives options possible.
Design & development
Taxonomy of costs
Forecasting technique
Forecasting method
Demonstration
Forecast of cloud costs based on input derived from historical usage data.
Evaluation
Variance calculations
Expert review
Communication
Master's thesis
Scholarly publication
Inference
Theory
'How to' knowledge
Metrics, analysis knowledge
Disciplinary knowledge
Problem centered approach
Objective centered solutions
Design & development centered approach
Observing a solution

FIGURE 1: DESIGN RESEARCH PROCESS

In order to adhere to the *design science* research strategy, the seven criteria of *design science* are applied and reflected on in Table 1.

Criteria		Application
1	Design as an artifact	The research results in the construction of a formally described method that depicts the abstract process of cloud cost forecasting. Additionally, the evaluation of forecasting techniques results in a proof of concept tool.
2	Problem relevance	The research project originated from a practical problem of a large IT organization in The Netherlands. The research will result in in-depth understanding of this problem and potentially a solution that will generate business in the near future.
3	Design evaluation	• The forecasting method will be evaluated through interviews and an expert review. • The forecasting spreadsheet artifact will be evaluated through a comparison between the existing situation without the artifact and through the application of the artifact.
4	Research contributions	• For the supporting company the research results in an improved understanding of cloud costs and a forecasting method. • Scientific contribution is made through the construction of a publishable article which describes the research results on cloud costs forecasting.

5	Research rigor	• Theory and practice are connected using the three *design science* research cycles (Hevner 2007). • A knowledge base will be constructed. Through the literature study an overview of relevant literature will be compiled. Literature review will be performed through the principles of Kitchenham (Kitchenham 2007). Only books, journals, workshops and conference papers will be used. • Evaluation of the forecasting artifact will be done using the variance which is measured comparing estimated and real costs of a recent project. Furthermore an expert review is performed in order to acquire validation from professionals with field experience.
6	Design as a search process	Part of the research uses a systematic literature review (Kitchenham 2007). Additionally unstructured interviews are held in the inception phase of this research to explore the problem domain.
7	Communication of the research	This research will result in both a thesis and a research paper aimed at publication at a conference or workshop.

TABLE 1: DESIGN SCIENCE RESEARCH CRITERIA

In time, several authors have argued that – although DSR projects should be equally focused on the creation of an IT-artifact as well as new scholarly insights – in practice DSR is criticized for overemphasizing the creation of IT-artifacts at the cost of little communicable scholarly insights (Hevner et al. 2004). Essentially, the analysis and interpretation of DSR results are generically described and leave room for interpretation (Hevner 2007; Kuechler and Vaishnavi 2008; S.T. March and Storey 2008). For this reason, others have combined the DSR approach with the *grounded theory method* (GTM) (Baskerville and Pries-Heje 2009; Goldkuhl 2004; Jarvinen 2007). The focus of GTM is the discovery of *grounded theory* (categories and relationships between them) (Glaser and Strauss 1967). GTM stems from the field of sociology and is interpretive in nature. GTM has been used in the field of information systems, especially in areas where existing theory does not apply or little is known (Bryant 2002; Shannak 2009). The process of GTM involves theoretical sampling and constant comparisons. GTM is used as a research technique for qualitative data analysis of existing cost models. Concluding, the GTM is used to extract more knowledge from the interpretation of the IT artifacts.

To differentiate in the research process between the research questions and sub questions, a schematic overview of research approach for each sub question is provided in Table 2. The overview includes a separation in research approach – and techniques employed – and the validation of the outcome.

Sub question		Research approach	Validation
A	What variables determine cloud costs and how do they relate to each other?	Structured literature review Grounded theory method	Comparison of accuracy of applicable techniques Expert review
B	Which cost forecasting techniques are there and how can they be used to support the prediction of cloud costs?	Literature study	Comparison of accuracy of applicable techniques
C	How can cloud cost variables and a forecasting technique be integrated in a technical solution that predicts cloud costs?	Construction of a technical solution Controlled experiment	Application on case Quantitative analysis Expert review

TABLE 2: ADDRESSING SUB QUESTIONS

RESEARCH PROCESS

Method engineering – resulting in a process-deliverable diagram (PDD) – is used as a way of graphical depiction of the method (Weerd and Brinkkemper 2008). The PDD consists of a meta-process model and an integrated corresponding meta-deliverable model. The process model depicts the individual activities and the possible flow through them. The meta-deliverable model depicts deliverables and the relationship between them. The process and deliverables that are covered in this thesis project are depicted in Figure 2. The different activities on the left-hand side of the figure are elaborated on in Table 4, the concepts or deliverables on the right-hand side are elaborated on in Table 3.

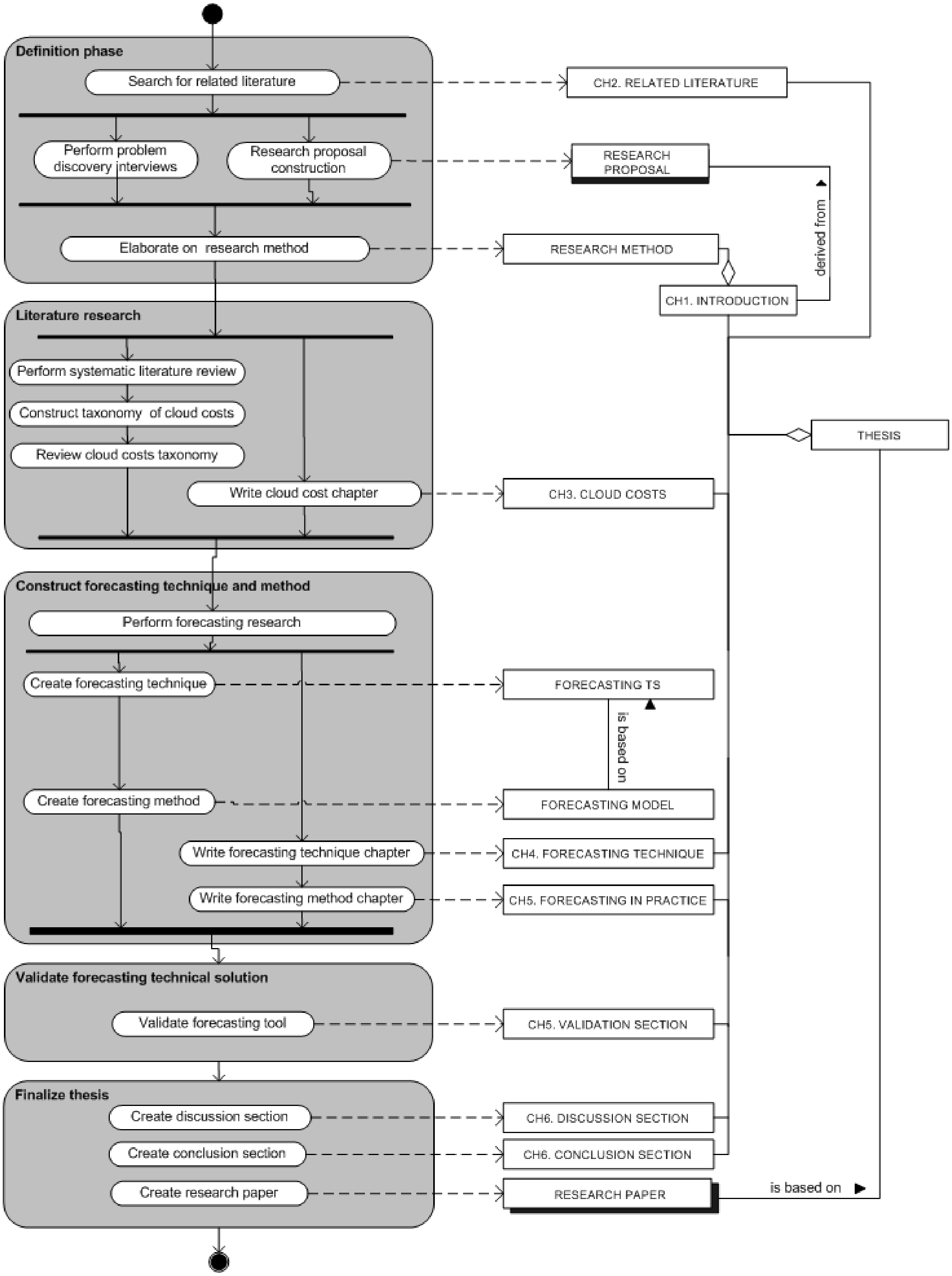

Definition phase
Search for related literature
Perform problem discovery interviews
Research proposal construction
Elaborate on research method
Literature research
Perform systematic literature review
Construct taxonomy of cloud costs
Review cloud costs taxonomy
Write cloud cost chapter
Construct forecasting technique and method
Perform forecasting research
Create forecasting technique
Create forecasting method
Write forecasting technique chapter
Write forecasting method chapter
Validate forecasting technical solution
Validate forecasting tool
Finalize thesis
Create discussion section
Create conclusion section
Create research paper
CH2. RELATED LITERATURE
RESEARCH PROPOSAL
derived from
RESEARCH METHOD
CH1. INTRODUCTION
THESIS
CH3. CLOUD COSTS
FORECASTING TS
is based on
FORECASTING MODEL
CH4. FORECASTING TECHNIQUE
CH5. FORECASTING IN PRACTICE
CH5. VALIDATION SECTION
CH6. DISCUSSION SECTION
CH6. CONCLUSION SECTION
RESEARCH PAPER
is based on

FIGURE 2: RESEARCH METHOD PROCESS DELIVERY DIAGRAM

DELIVERABLES

The thesis project is divided into four phases. The deliverables described in the process delivery diagram depicted in Figure 2 are outlined and elaborated on in Table 3.

Concept	Description
RESEARCH PROPOSAL	The report that specifies the initiation of the research project.
THESIS	The report that describes the result of a research project into cloud cost forecasting as well as how this results has come to be.
CH1. INTRODUCTION	This is an introductory chapter of the THESIS report. This incorporates an improvement on the RESEARCH PROPOSAL.
RESEARCH METHOD	This is a section of the CH1. INTRODUCTION chapter which covers an elaboration on the research method.
CH2. RELATED LITERATURE	This is a chapter of the THESIS report. It covers relevant literature about the research.
CH3. CLOUD COST	This is a chapter of the THESIS report. It covers a systematic literature review about the concept of cloud costs.
CH4. FORECATING TECHNIQUES	This is a chapter of the THESIS report. It covers an elaboration on several existing forecasting techniques, and discussed the relationship and potential for cloud cost forecasting.
CH5. FORECASTING IN PRACTICE	This is a chapter of the THESIS report. It suggests a process and deliverables of a cost forecasting cycle and its implementation.
CH5. EVALUATION SECTION	This is a section of the CH5. FORECASTING MODEL chapter. It covers the validation of the FORECAST TOOL and FORECAST METHOD artifact.
CH6. DISCUSSION SECTION	This is a section of the final chapter. It covers an academic discussion about the THESIS.
CH6. CONCLUSION SECTION	This is a section of the final chapter. It covers the conclusions of the findings of the research and future research directions.
FORECASTING TS (technical solution)	An artifact that consists of a software tool that takes usage metrics derived from cloud services and calculates platform and infrastructure costs.
FORECASTING MODEL	An artifact that consists of a method, comprising a set of activities and deliverables that produce a cloud cost forecast.
RESEARCH PAPER	A potentially publishable scientific paper based on the THESIS.

TABLE 3: PROCESS DELIVERABLE DIAGRAM CONCEPTS

RESEARCH PHASES

The research project consists of five phases. The thesis document is the result of all the phases and is worked on throughout the project.

Activity	Sub activity	Description
Definition phase	Search for related literature	Compose a section of related literature about the problem statement and the research question.
	Perform problem discovery interviews	Formulate a problem statement and validate this in practice.
	Research proposal construction	Compose an initial approach for a possible thesis project based on the problem discovery interviews.
	Elaborate on research method	When the project is granted an elaboration on the research method is provided using a step-by-step approach.
Literature research	Perform systematic literature review	Iteratively review scientific literature, based on a key word, according to inclusion and exclusion criteria.

	Construct taxonomy of cloud costs	Construct a taxonomy of cost variables based on scientific literature.
	Review cloud costs taxonomy	Review scientific literature about cloud costs.
	Write cloud cost chapter	Describe the composed artifact and the describe findings derived from the process of constructing an artifact.
Construct forecasting technique and method	Perform forecasting research	Research techniques of cost findings in literature and their applicability for cloud cost forecasting.
	Create forecasting technique	Cost forecasting can be performed on different manners. Through a literature study different techniques are evaluated based the applicability of the technique for cloud cost forecasting. The resulting set will be evaluated on applicability for Centric. When applicable, forecasting techniques are integrated in the cost forecasting model, future optimization that are not yet applicable will be elaborated on. The eventual cost forecasting model will be validated through a practical case.
	Write forecasting technique chapter	Describe possibilities of forecasting techniques in relation to cloud costs based on the literature review.
	Create forecasting method	An approach is created to forecast cloud costs.
	Write forecasting method chapter	Describe how forecasting techniques could be used in practice.
Validate forecasting technical solution	Validate forecasting tool	Perform a controlled experiment and document findings for an application of cost forecasting techniques. Perform an expert review to validate findings.
Finalize thesis	Create discussion section	Write a discussion about research, possible problems and provide future research directions.
	Create conclusion section	Write section which answers research questions and summarizes most important findings. Provide future research directions.
	Create research paper	Distill one or more important findings from the thesis and describe these in a scientific research paper.

TABLE 4: ELABORATION ON RESEARCH ACTIVITIES

DEFINITION PHASE

The definition phase incorporates an initial proposal, five unstructured interviews with professionals that have in-depth knowledge of the current situation at Centric as well as a good understanding of the cloud industry. The interview candidates are selected to provide an understanding of the problem statement from three different perspectives being: profession or function within the company, industry and affiliation with cloud services, these are depicted in Table 5. It is important to note that these interviews are only used to validate the research question, not a as a part of the research method itself.

Interview	Profession	Technical affiliation	Industry
1	Manager research & development	IaaS, PaaS, SaaS	General
2	Manager sales	Mainly SaaS	Supply chain & retail
3	Director IT services	IaaS, PaaS, SaaS	IT services
4	Consultant	Mainly IaaS	General
5	Consultant	Mainly IaaS	General

TABLE 5: RESEARCH DEFINITION INTERVIEWS

Literature review – cloud costs

The literature study is performed on the topic of cloud costs. Theory on the topic is gathered through published materials in journals, papers, conferences and workshops. A database is built of relevant publications is constructed through the use of a practice called systematic literature review (Kitchenham 2007).

According to the principles of an systematic literature review (SLR) a formal review protocol must be articulated (Kitchenham 2007). The research questions that need to be answered in the review are: what are sources of costs; can variables be determined that drive costs; what is the relationship between cost drivers and costs.

This review uses electronic literature databases well known to the field of information systems. The review will start by retrieving an initial set of literature through the use of a literature meta-search engine called Google Scholar. Google Scholar provides a gateway to several literature databases. Using a trial search approach with Google Scholar, keywords were selected that are used later on in the literature review. The keywords selected is: 'cloud AND costs'. Search results will be used to compile a literature database. The literature database is checked for completeness of the information and duplicates are removed in the preparation phase. Results are processed in three rounds using inclusion/exclusion criteria. In the first round inclusion is based on: study must be peer reviewed (conference proceeding or a journal publication); study must be in English; study must be available as a full text through the UU library. In the second round the title, keywords and/or abstract must indicate that the IT related concept of cloud cost is discussed; the article must indicate that costs are discussed (and not just mentioned). The third round consists of a full text review, for which inclusion is based on the criteria: study should actively discuss the sources and or attribution of costs to a cloud service; literature should contribute to the answer of the research question.

Data extraction

For data extraction, a structured form was used that is used to process each literature entry from the final set. The form relates to the three questions central in the SLR and includes traceability to the source literature, the structure is proposed in Table 6.

Paper	Cost variable and measure	Cost driver and relation	Remark
#ID	That what is paid for	Unit of that what drives cost and the relation	

TABLE 6: DATA EXTRACTION FORM FOR CLOUD COSTS

Synthesis of the data to taxonomy will be performed after the complete literature database is reviewed. The taxonomy of costs is constructed using aggregation and segregation of cost variables in an iterative manner.

For each paper, core concepts are being recorded for descriptive purpose. Core concepts are keywords and in case these are missing, core concepts are derived from the paper title and abstract (when available). The structure of the recording form can be found in Table 7. Also, a recording is being made of the opposition of the paper towards cloud products. A positive opposition is noted when a paper explicitly prefers cloud IT over traditional IT, neutral when there is no explicit preference and negative when the paper explicitly prefers traditional IT over cloud IT.

Paper	Core concepts	Opposition
#ID	Keywords, and important concepts	Positive/negative: explanation

TABLE 7: DESCRIPTIVE LITERATURE INFORMATION

Data quality is addressed by including conference, workshop and journal publications. The publications are retrieved by using the publishers search engines instead of a meta-search engine for which the quality is difficult to assess. Quality of analysis is addressed by using a predefined format for data-extraction which should improve possible replication of the findings. Also, for each literature item it is possible to trace back to where the literature item was discarded in the process and when not easily derived from the review protocol the reason for removal. Furthermore, search results are saved as an offline webpage to ensure that the query results are not changed during the research.

A CONTROLLED EXPERIMENT

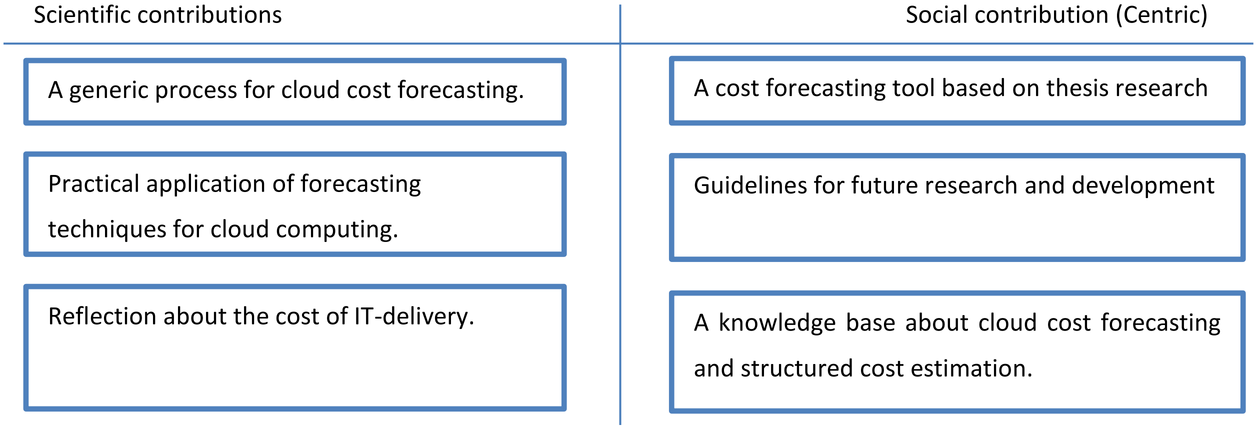
The technical solution as part of the design science research method is tested by means of a controlled experiment. Highly structured guidelines are implemented to increase both external and internal validity, and ensure proper research communication (Jedlitschka, Ciolkowski, and Pfahl 2008b). By applying such guidelines, experiments would lead us to 'gain more understanding of what makes software good and how to make software well' (Pfleeger 1999).

EXPERT REVIEW

An expert review is used to interpret the results from the experiments performed on the technical solution. Experts will interpret the results from the experiments and will determine the practical value of the suggested technical solution. For the selection of experts people with different backgrounds are selected. Ideally, the experts are people that are likely to work with these forecasts in a possible future. Furthermore, an expert is required to review the calculations and their results. The expert review conducted as a validation contains a face-to-face interview in which the interviewer describes and shows the research results and the expert provides feedback. The goal of the expert review is to receive feedback from the expert about the research results on which the forecasts method is created, the output of the forecast in terms of numbers, and the output of the forecast in terms of practical application. The interview protocol as well as their results can be found in Appendix H. The experts are requested consent to record the interview. Results are processed in a fashion that answers on the question are all included. However not all experts will likely be able to answer every question. Additionally, comments that aid interpretation of the results are processed as quotes.

1.4 Scientific and social contribution

The thesis project consists of deliverables for both academia and Centric. While there certainly will be some overlap between these deliverables it is important to recognize that each have its target audience, these are graphically depicted in Figure 3.



Scientific contributions

A generic process for cloud cost forecasting.

Practical application of forecasting techniques for cloud computing.

Reflection about the cost of IT-delivery.

Social contribution (Centric)

A cost forecasting tool based on thesis research

Guidelines for future research and development

A knowledge base about cloud cost forecasting and structured cost estimation.

FIGURE 3: SCIENTIFIC AND SOCIAL CONTRIBUTIONS

1.5 Outline

The thesis roughly follows a structure that covers the different deliverables in a single chapter. First, related literature on the topic of forecasting of cloud costs is discussed in chapter two. Chapter three covers the result of an extensive literature study about cloud costs, chapter four covers cloud forecasting techniques, and chapter five discusses a technical solution aimed at cloud cost forecasting. A discussion, conclusions and suggestions about future research are provided in chapter six.

Chapter 2. Related literature

This chapter will elaborate on some definitions that are used in the research in order to achieve a shared understanding as well as scope of these subjects. Because the research method includes a systematic literature review this chapter only covers a reflection of current literature that partly addresses the research question and related problem statement for this research.

Cloud computing is ‘a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction’ (Mell and Grance 2009). Cloud computing comprises three service models, which are *Software as a Service* (*SaaS*), *Platform as a Service* (*PaaS*), and *Infrastructure as a Service* (*IaaS*). Definitions of the three service models are provided below:

- *Software as a Service*. ‘The capability provided to the consumer is to use the provider’s applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user specific application configuration settings’ (Mell and Grance 2009).
- *Platform as a Service*. ‘The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment’ (Mell and Grance 2009).
- *Infrastructure as a Service*. ‘The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls)’ (Mell and Grance 2009).

The definition of architecture provided by (Mell and Grance 2009) leans to the perspective of the consumer. From an operator’s perspective, economies of scale are important as cloud IT involves high quality expensive products. Next to the three service models which can be seen as stackable abstraction layers there are also four deployment models, which are *private cloud, community cloud, public cloud, and hybrid cloud.* Definition of the four deployment models are provided below:

- *Private cloud*. 'The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises' (Mell and Grance 2009).
- *Community cloud. '*The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises*'* (Mell and Grance 2009).
- *Public cloud*. 'The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider' (Mell and Grance 2009).
- *Hybrid cloud. '*The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds)*'* (Mell and Grance 2009).

The research aims at cost calculation and the ability to forecast cost for cloud technology. Although concepts of cost calculation and the related total cost of ownership has been around for many years in IT and other industry a disruptive nature of cloud technology requires additional research. An important difference for the IT industry is the transition towards IT as a utility, which impacts both price and costs. The transition towards a pay-as-you-go service also introduces the need to predict costs (Truong Huu et al. 2010). Until now there has not been a significant contribution in literature that addresses the forecasting of costs for cloud services.

Until now, academics as well as organizations have praised cloud technology for its potential for cost saving. The majority of researchers include cost savings as a rationale for their research. So far, there have been several papers that include cost calculation on the infrastructure level (X. Li, Y. Li, and T. Liu 2009; Lindner, Galán, and Chapman 2010; Martens, Walterbusch, and Teuteberg 2012). However, there is little proof about how cost variables are derived. Also, current research focusses on the total cost of ownership (TCO) of the infrastructure limited to a cloud platform operator perspective. However, TCO is a concept that is deeply tied into traditional on premises infrastructure costing. It makes sense to rediscover the costing approach again for cloud costs. An integral cost calculation for software as a service that includes platform and infrastructure costs have not been described in literature.

Several cloud infrastructure providers have recently introduced cloud cost calculators of some sort to come up with an estimation of monthly costs. Among these are: Alinean (Corporation 2012), and IBM (IBM corporation 2012). An important assumption made by these cost calculators is that customers are required to know how much they will consume, as this is the required input for these cost calculations. However when a customer uses software as a service, it will have little concern about infrastructure cost incurred by the vendor.

Much effort has been made to make cloud technology more cost efficient, meaning getting more value for resource spend (Tian, Song, and Huh 2011). Contributions are aimed at increased utilization of servers which historically is quite low. *Economies of scale* have been a manner to increase cost-efficiency, but require scalability as a design constraint in both cloud software and infrastructure.

Forecasting of computing resources related to cloud has already been performed for workload scheduling (Caron, Frederic Desprez, and Muresan 2010; Caron, Frédéric Desprez, and Muresan 2011; S Chaisiri 2009). Another applications of workload scheduling is to control energy costs (C. Liu and H. Yi 2010; Mazzucco 2010). However, workload scheduling affects costs but this is not a manner to calculate or predict costs. The performance achieved in cloud environments can be simulated to aid budgeting and planning of research (Deelman et al. 2008; Hong-Linh Truonga,∗ 2010; Truong Huu et al. 2010). However this research aims at comparability between different scenarios. However, there is currently no comprehensive approach to forecast the costs of cloud.

Chapter 3. An overview of cloud costs

This chapter is aimed at answering the following research sub question:

a. What variables determine cloud costs and how do they relate to each other?

In order to answer the question, a comprehensive overview of cloud costs is composed which will be called the 'taxonomy of cloud costs'. The taxonomy is the product of a systematic literature study (Kitchenham 2007). In addition to the taxonomy composed from literature, cost drivers are also part of the research. The relationship between cost variables will be documented, and later on formalized in a cost model.

The research is aimed at providing a forecast for cloud costs. Different approaches for cost calculation exist, which differ in the scope of their definition. In order two answer the research question posed in this chapter the focus is set on cost variables that are affected by the daily usage or operation of a cloud product.

3.1 A SYSTEMATIC LITERATURE REVIEW ON CLOUD COSTS

There exists an abundance of literature on the topic of cloud computing. This section presents the result of an extensive literature review. The details of the literature review can be found in the research method section.

The systematic literature review process consists of five phases which are carried out sequentially in order to come up with high quality literature set. The different steps are described textually in detail below and the process and progress is graphically displayed in Figure 4.

1. Search execution: is performed between March 28, 2012 and April 10, 2012. The literature entries were stored into a literature database. At the end of the phase the literature set consisted of 489 results. Because the search interface is different for each publication database the search strategy differs for some of the sources. The IEEE database did not supported search through abstracts. After the consideration of a full text review, which resulted in 4.293 items, this is considered very ineffective. The only other option was a title-only search which resulted in 55 articles. A similar approach was used for the Scopus database which downsized the result from 1248 to 35.
2. Result preparation: consists of a check for completeness before the review process. There should at least be a title, abstract, keywords, and full text available. Duplicates are removed. At the end of the phase the literature set consisted of 419 results.
3. Phase 1 consists of filtering using the inclusion/exclusion criteria as proposed by the review protocol. At the end of the phase the literature set consisted of 373 results.
4. Phase 2 consists of filtering using the inclusion/exclusion criteria as proposed by the review protocol. At the end of the phase the literature set consisted of 139 results.
5. Phase 3 consists of filtering using the inclusion/exclusion criteria as proposed by the review protocol. At the end of the phase the literature set consisted of 60 results.

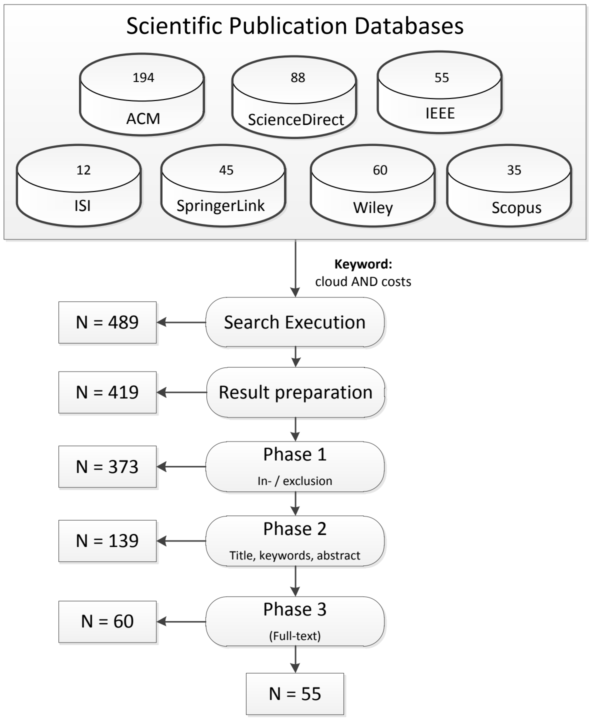


FIGURE 4: STRUCTURED LITERATURE REVIEW SEARCH PROCES

INSIGHTS INTO DISCARDED ARTICLES

In the process of structured literature review an enormous set of articles is processed and most of the articles are discarded from review. This section provides insights into the rationale of the major reasons for discarding articles. A small portion of the articles was not related to IT. Most of these articles discuss the atmospheric clouds or cloud patterns in 3D or medical science. Most articles that are discarded in the second and third check using inclusion and exclusion criteria are related to cloud computing and did mention costs. Because costs have been a rationale for migration towards the cloud, many articles use it as part of a research rationale. Most of these articles are about security, data storage and architecture. An efficient use of resource is a popular subject where costs is an active subject of the paper but most articles fail to provide an understanding about how costs are incurred but merely suggest methods for efficient resource utilization, hence avoiding costs.

This section covers a review of the cost variables found in the literature set. Through a systematic literature study a set of 55 papers are selected that actively discuss the relationship between clouds and costs. All papers are processed using a structured form for recording cost factors. This enables traceability for each factor to relevant literature. These cost factors are aggregated into the taxonomy of cloud costs. Some cost factors are bundled and others are unbundled to come up with a classification with distinct but related cost factors. Some variables are wrapped in each other which form bundled goods, like an instance (or virtual machine) is a bundle of memory and

CPU. Also there is a difference in the granularity at which the cost variables are described. Some papers just use network costs while others specify local and wide area networks. Also there are variables that are only described in a single paper. Using an approach from the grounded theory method (GTM) known as iterative classification a smaller set of the total taxonomy of costs is constructed (Glaser and Strauss 1967). From the collected papers key cost variables are selected and marked with a code. Codes are grouped into similar concepts. The process iterates until all material is classified. The concepts are categorized and each group forms a source for grounded theory. Grounded theory consists of common well described cost variables. A cost driver is a factor that causes a change in costs. Every cost variable has a cost driver that explains the level of costs. The cost variables that are discussed in at least five papers are discussed in Table 8, along with their corresponding cost drivers.

Cost variable	**Cost driver**
Virtual machine	Virtual machine costs are driven by the time it is used and the computational capacity it has.
Network	Network costs are driven by the capacity for throughput, actual throughput (measured in bytes). Also the amount of physical and virtual network addresses add to the costs.
Storage	Storage costs are driven by the amount of data stored, read and write activities on storage (also known as I/O), and the time data is stored.
Memory	Memory costs are driven by the amount of memory available in (measured in bytes).
Energy	Energy costs are driven by the amount of energy consumed (measured in watt).
CPU	CPU costs are driven by the processing capacity of the CPU, the amount of cycles it is run and the amount of processing cores used.
Service level agreements and objectives	Service costs are driven by the level of service supplied. These costs can be viewed as a meta-level cost driver that describes the quality of other cost variables. This makes it impossible to describe a single unit of measurement.

TABLE 8: COST VARIABLES AND COST DRIVERS

An overview of categories and their source in literature can be found in Appendix C: Literature research classification matrix. The cost variables where systematically gathered in the final review round of the SLR using a structured review form. Figure 5 depicts a holistic picture about cost variables processed in the literature study.

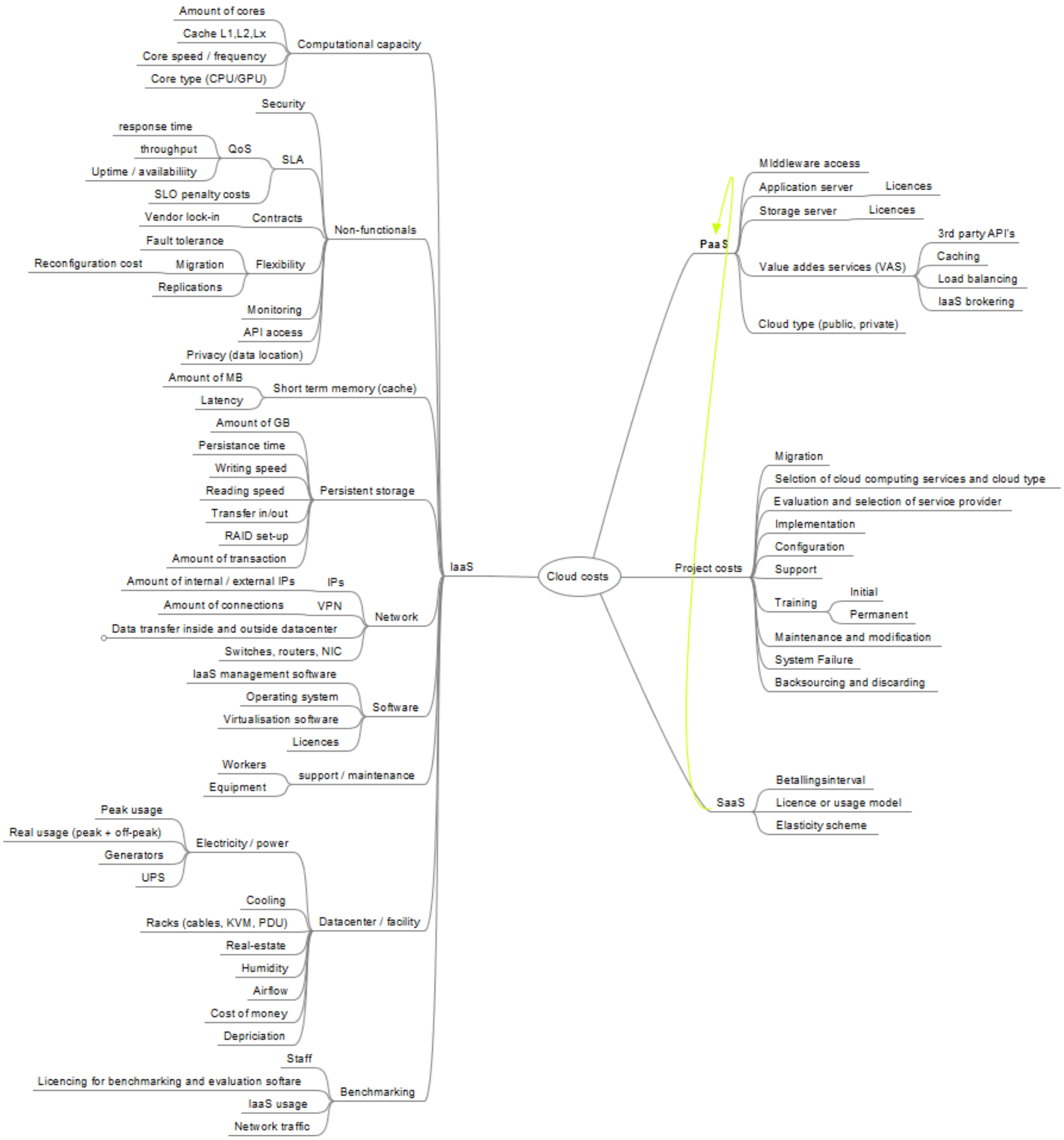


FIGURE 5: TAXONOMY OF CLOUD COSTS

3.2 LITERATURE REVIEW: CLOUD COSTS

The section covers a review of literature that explains costing from a cloud perspective. It covers the rationale behind costs related to cloud technology. There is an abundance of literature containing *solutions proposals* for a more efficient use of resources in the cloud. In general these papers discuss a way to avoid costs by making more efficient use of resources. However, these papers provide highly insightful information about where and how costs occur in the cloud value chain.

The literature is ordered using the three layers of cloud technology adopted from the NIST (Mell and Grance 2009). In addition, to this there is a generic section that covers literature unrelated to a specific level of service.

CLOUD COSTS FOR SOFTWARE AS A SERVICE

Cloud technology represents flexibility in demand of cloud technology. From the outside, cloud technology has become a rather homogeneous product with some variance in the added value services (for example: monitoring). A consumer can determine how much capacity and on at what quality (QoS) the service should be delivered.

In reaction to the difficulties surrounding cloud product selection, brokerage services have emerged to mediate between cloud customers and cloud operators (Lucas-Simarro et al. 2012). Cloud operators exhibit many differences regarding the functionality and usability of exposed cloud interfaces, methods for managing images, types of instances, level of customizations, pricing and charging models and corresponding pricing models. Cloud brokers can provide a uniform interface independently of a particular cloud. Services provided by cloud providers are: intermediation, aggregation, and arbitrage.

To assist a consumer on cloud decision making – considering an abundance of choices – when acquiring a cloud service, petri-nets can be used to model the decision process (Bai et al. 2011). Decisions can be modeled as transitions which have price parameters and states are the corresponding situations in the acquisition process. A benefit of such formal models is that it enables calculations like: profit maximization and cost minimization for a given pricing scheme.

CLOUD COSTS FOR PLATFORM AS A SERVICE

In the cloud, storage itself is rather cheap; however structured storage using databases management systems is highly costly. Traditionally database systems are developed to deliver performance while the hardware available and determined the actual performance. In the cloud the available hardware resources is virtually unlimited so these traditional systems have the tendency to consume an irrational part of the available budget. A manner to counter it, is cost amortization of database queries (Kantere et al. 2011). This means that data that is permanently stored can enjoy performance optimization but in time as data get less queried or changes this optimization will decrease. Eventually the amortization causes that less data is optimized at a given time which lowers the load on infrastructure, hence decreases or avoids costs.

Because of the enormous scale of cloud infrastructure the cloud received attention from companies like Google and Facebook that process enormous amounts of data in a parallel fashion. Organizations dealing with fluctuating performance requirements, or that perform batch processing, can potentially benefit from cloud technology by using an efficient scheduling strategy for computational jobs. In (M. D. Assunção, Costanzo, and Buyya 2010; M. D. D. Assunção, Costanzo, and Buyya 2009; Csorba, Meling, and Heegaard 2011; Mian, Martin, and Vazquez-Poletti 2012; Moschakis and Karatza 2010; Pandey 2011; Sharma et al. 2011; Truong and Dustdar 2010; Truong Huu et al. 2010; L. Wang and Zhan 2009; L. Wu, Kumar Garg, and Buyya 2011; Z. Wu et al. 2011; Q. Zheng and Veeravalli 2012) different scheduling strategies are described that have distinct cost/benefit qualities. The studies evaluate the cost of improving the scheduling performance of virtual machines by allocating additional resources from a cloud computing infrastructure. The main objective of scheduling is to maximize performance by avoiding unnecessary delays, and also maintaining a good leasing cost/benefit ratio. Additionally, heterogeneity or the lack of heterogeneity can pose a problem for batch processing in the cloud. Although the term batch job indicate a set of similar tasks in practice the task characteristics are quite different and follow a bimodal pattern (Boutaba, Cheng, and Q. Zhang 2011). Tasks show a bimodal distribution in length and size, there also is variance in the burstiness of job arrival rates. Existing scheduling techniques is that different – sometimes contrasting – factors are taken into account and often a trade-off is made while calculating a priority for a task. A problem arises when tasks are heterogeneous because most computational models are unequipped for handling a great diversity in tasks. Another characteristic that heterogeneous tasks can have is a burstiness in workload, which means that workloads can have a CPU utilization of 50 % but some tasks require demand several CPUs – creating a peak load of the infrastructure (Gmach, Rolia, and Cherkasova 2010). Burstiness of workload can benefit from having a more heterogeneous infrastructure, which is not common for large datacenters. Large datacenters achieve economies of scale is through large-scale acquisitions of hardware. Furthermore, individual low-level tasks that can be assigned to a non-high-end system can be performed less costly.

CLOUD COSTS FOR INFRASTRUCTURE AS A SERVICE

Cloud technology uses virtualization techniques to dynamically allocate physical resources and assign bundles to the operating system. Virtualization is the mapping of component—such as a processor, memory, or an I/O device—at a given abstraction level onto the interface and resources of an underlying, possibly different, real system (Smith and R. Nair 2005). In most cases the operating system itself is also virtualized and operates on a virtual machine. A single virtual machine (VM) can be used in practice for data processing. Advocates of cloud technology praise the dynamic allocation of VMs, and subsequently its low-level resources, because it is an enabling technology for a pay-per-use costing approach.

Cloud computing is economically tenable, on the operational – day to day – level when costs associated with cloud placement – or outsourcing – outweighs de associated deployment costs in the cloud (Y. Chen and Sion 2011). Different types of organizations and IT requirements all have a different tipping point for it. In order to calculate the cost for running the existing situation are crucial. A cost model that describes computation, storage and networking cost describes a significant portion of the real costs. Other, often related costs can be wrapped up like: floor space and energy costs. Outsourcing is feasible when the applications are computational intensive. It means that the cost-savings are sufficient to offset of client-cloud network costs. In practice this means a minimum of 1000 CPU cycles – single instructions carried out by the CPU – per each 32 bits of client-cloud transferred data. Furthermore, it is the tipping point for a single-client environment. On a larger scale or in a multi-client environment outsourcing will be feasible running 1000 instances when computational requirements reach 410 CPU cycles per 32 bit data or even less with guaranteed network service. For organizations operating 10.000 or more instances outsourcing will in general be economically viable. In (Deelman et al. 2008) an experiment is performed that explores the business model of cloud services. This research indicates that for applications that are data-intensive the cloud is cost-effective.

Today, the cloud operators have established their own vocabulary, which makes comparison and operator independence more difficult (A. Li et al. 2010). The research suggests benchmarking cloud providers and translation of benchmark measures to performance metrics which can be used in cost calculations and comparisons.

Cloud technology is aimed at on demand provisioning of computing resources. The contracts that describe theses services should be adjustable on demand as well. SLAs describe objectives of service usage between traders. However, the IT industry has a history of complex time consuming SLA negotiations, which are expensive and have a negative effect on the potential economic benefits of the cloud. Furthermore, a business requirement for on demand resources cannot be fulfilled when contract negotiations are in the way. One way to achieve more flexibility is using self-adaptable service level agreements (SLAs). Using generic contract templates for user requirements and a matching algorithm, SLAs are a matter of mapping. The autonomic mapping of user requirements to SLA templates can be performed by creating clusters of customer that have similar requirements (Breskovic et al. 2011). Cost-efficient clustering can be performed using the *k*-means algorithm. The approach increases net utility of traders and market in general through a decrease of maintenance costs. In cloud environments the distribution of workload can also degrade workflow outcomes, as the chance that one processing node fails increases which potentially causes SLA violations (Ramakrishnan and Reed 2009). Workflow distribution algorithms can solves this violation by monitoring workflow health and reacting upon failures.

Cost savings can be considered one of the premises of cloud technology. Much of current literature mentions a shift in how IT is paid for as part of the research rationale. It can be performed through scaling of resources – or elasticity – meaning the timely allocation and also de-allocation of resources for a service. While the technology itself is in principal not less expensive, a more efficient use of IT potentially decreases costs for a company. Moreover in the situation where demand fluctuates, scaling is the preferred approach to save costs, this is called avoiding overprovisioning (Ferrer et al. 2012). When demand is too high for the current capacity a delay occurs or requests are denied, both negatively affect the quality of service, this is called under provisioning. Cloud technology achieves flexibility through an on demand delivery model using resource scaling. However, a cloud client's – user or middleware – has to make decisions on when to scale up and down to balance cost and performance.

At the present time there is a non-negligible set-up time necessary to commit new resources to a customer. When an optimum efficiency can only be achieved by just-in-time scaling, prediction of workloads becomes a necessity (Caron, Frédéric Desprez, and Muresan 2011). One way to achieve it is identifying similar past occurrence in short term workload history. By identifying resource usage patterns that have occurred in the past that are similar to the present pattern decisions can be derived to down- or upscale resources. An algorithm predicts what likely will happen in the near future by interpolating what will happen in the historical data. Through monitoring and prediction and the use of cloud providers APIs dynamic autonomous resource scaling – or auto-scaling – has become a cost saver.

Another cost saver is scalability in capacity per virtual machine, also known as an instance. In cloud environments instances can be scaled in quantity but in most cases also capacity. In environments where demand in capacity is dynamic it is a potential cost saver. When a software architecture is optimized for parallelization having multiple low capacity instances can be more cost effective over heavy instances equaling the total capacity (Chard et al. 2011). Additionally , reconfiguration of virtual machines – resizing VMs or live migration – also consumes additional resources (Verma, Kumar, and Koller 2010).

Flexibility using instance scaling comes at a cost (Deelman et al. 2008). Cloud applications have startup costs: launching virtual machines, the teardown of virtual machines, and configuration of virtual machines. Also there are costs related to building the cost for building virtual images and keeping them up-to-date. A manner to decrease set-up costs is through auto-scaling and virtual machine queuing (Dougherty, White, and D. C. Schmidt 2012). Auto-scaling anticipates demand and prepares virtual machines in a queue which enables timely scaling. A queue of virtual machines that are out of production can be kept alive until demand increases. The problem is that today's cloud environment supports a wide range of configurations, thus a wide range of virtual machines. Furthermore, the costs of maintaining a queue are not negligible either. In large scale systems effective queuing of virtual machines using auto-scaling has major cost benefits. An improvement on auto-scaling for environments with high variance in environments and configurations is SCORCH (Dougherty, White, and D. C. Schmidt 2012). SCORCH uses feature models and *constraint satisfaction problems* to match environment requirements to queued virtual machines which significantly reduce operating costs. In practice, metering virtualized cloud instances is still immature. Cloud providers use metering for billing purposes but at the moment it is difficult to verify the accuracy of these metering tools (Sekar and Mantis 2011).

Load planning – or reservation – can lower cloud costs, especially in public clouds. While most cloud providers also offer resources on demand, they come at a higher price than planned resources. However, it is difficult to reserve the right amount of resources. An approach to determine the amount is optimal cloud resource provisioning (OCRP) (Sivadon Chaisiri, Lee, and Niyato 2011). OCRP uses a stochastic programming model is created which factors in demand and price uncertainty to come up with a cost-effective reservation plan.

While resource allocation can certainly play its part to cut costs, for applications that require less than one server the approach has limited options. For instance, a small database that requires a dedicated server but only uses a small portion of the server's capacity. In order to lower cost, decrease maintenance and overall make a proper use of resources a consolidation strategy is required that aims at matching server load with capacity (Ghanbari et al. 2012).

Scaling can also be performed at *on premises* hosted infrastructure, in combination with the cloud – which is called a hybrid cloud (O. Mazhelis, Tyrväinen, and A. Mazhelis 2011). Scaling to the cloud when extra capacity is necessary relieves workload of the traditional infrastructure. This enables an infrastructure set-up planned for a medium load scenario on an on premises infrastructure which is cost-effective while still being able to cope with peak loads using cloud resources. With decreased control over public-cloud resources a balance between privacy risk and data location costs should be taken into consideration (Tian, Song, and Huh 2011).

Public cloud operators like Amazon EC2 have sold spot instances for prices under 50% of the regular price for short term capacity (S. Yi 2011). For these so called *spot instances* prices are established by the degree of overcapacity of the datacenter and the price customers are willing to pay for the short-term capacity on that moment by bidding on it. When the overcapacity of the datacenter decreases spot instances can be withdrawn, this causes availability of service to fluctuate. To decrease networking costs data can be more dispersed by replicating data to multiple servers (Z. Wang et al. 2011). This eliminates bottlenecks in sometimes congested networks.

To make proper use of elastic cloud computing models, like Amazon EC2, a minimum of computing hosts – or virtual machines – is the cheapest solution (Byun et al. 2011). For a given workload, the *portioned balanced time scheduling* (PBTS) algorithm calculates a minimum of virtual machines is calculated for a user-specified finish time. Furthermore, the PBTS algorithm assigns workflow tasks in an effective manner without affecting performance. Additionally, the amount of instances have marginally decreasing effect on the total computational capacity (Napper and Bientinesi 2009). The decreasing return to scale of instance in the cloud has also been acknowledged in (Pandey 2011).

INFRASTRUCTURE COSTS AT FACILITY LEVEL

On the lowest level, facilities that support computational infrastructure comprise: housing, power supply and distribution, networking and rack housing. A rough guideline for large data center costs are 45 % on servers, 25% on infrastructure, 15 % on electricity, and 15% on network costs (Greenberg et al. 2008). Off course, efficiency and therefore the propositions are largely determined by the efficiency of a data center. Improving efficiency or utilization means getting more work accomplished per dollar invested, unfortunately in practice utilization is often around 10 %. A reason for this can be an improper *application fit*, which means that software does not make proportional use of all resources within one physical server, like memory, CPU, network, storage, or optionally a GPU. An additional inefficiency is *uncertainty in demand* forecasts which forces data centers to be able to cope with a peak well above the 95th percentile. *Long provisioning time scales* caused by the demand for economies of scale cause ordering process to take months. Moreover, infrastructure lasts quite long with a depreciation time of fifteen years for low-level equipment and 3-5 years for servers. *Risk management* tips the balance from under provisioning to under provisioning cutting profit margin. Finally, there still exist *virtualization short-falls* in equipment that insufficiently support virtualization, like load balancing equipment.

Costs for cloud computing can be analyzed using a *total cost of ownership* (TCO) model combined *with utilization costs* (X. Li, Y. Li, and T. Liu 2009; Martens, Walterbusch, and Teuteberg 2012). *Utilization costs* are the costs of real resources lockup up or committed to a particular user or application. Cloud TCO represents the costs spent to build and operate a cloud. A cloud TCO should consider amortization for different items over different lengths of time – for instance 10 years for buildings. Virtual machines represent packaged goods which are consumed by customers, it makes them suitable as input for a cost calculation. A TCO model offers a mathematically representation of the 'real world' in a simplified and abstract way. Quality attributes of a TCO model are:

- Transparency: the descriptiveness of the model and criteria;
- Applicability: the easy to which the TCO model can be used in a real life situation;
- Variability: the degree to which the model can foster variability in input and adaptation or extension;
- Comparability: the degree to which the outcomes of the model can be compared with each other;
- Decision support: the amount of information that the model offers to support a decision;
- Status quo: the model should be based on current business practices and the state-of-the-art literature.

TCO calculations can also incorporate migration costs (Tran and Keung 2011). For a large part migration cost stem from effort cost for redesigning software for cloud purposes. However software is not always adjusted for the cloud. Increasingly, there will be software that is compatible with cloud software delivery. The notion of migration cost existed before cloud technology, for example when switching software platforms, which makes it difficult to consider it as an intricate part of a cloud TCO or to leave it outside.

Cooling a data center can take a significant amount of energy. Although innovative ideas have come to pass, large scale data centers typically use electricity powered cooling systems that can take up to 45 % of the total power consumption (Le et al. 2011). Power costs can be reduced, through workload scheduling using geographically dispersed location with different power costs the power consumption costs (Mani and Rao 2011).

CPU's require electricity to operate which is a significant part of the operating costs of the infrastructure, with around 15 % (Greenberg et al. 2008). High performance computing relates to high energy consumption and high carbon emissions. Electricity has prices that depend on local markets which vary between countries. There is an increase in power consumption caused by cloud computing in the recent years because of the increased utility rate (Garg et al. 2011). Measurements indicate that it would be feasible to power down virtual machines when there is overprovisioning. The benefits of having near zero demand for electricity exceeds the electricity necessary for powering down and booting virtual machines (Lefèvre and Orgerie 2010). A more efficient use of electricity would be scheduling between datacenters. This would mean migrating virtual servers geographically to a data center with cheaper electricity. In addition, when servers do not run on peak load there is a possibility of dialing back the voltage of CPUs to further lower electricity usage with a total up to 33 %.

The growing interest in cloud services and the emphasize on utilization has created the need to accurately measure service performance (Haak and Menzel 2011). Especially when across different architecture levels self-optimization and self-configuration algorithms are trying to maximize performance and minimize costs. Data centers with physical or virtual hardware specifications use performance indicators and metrics which largely describe services as isolated parts, not how they work in practice.

3.3 DISCUSSION OF RESULTS

The research question that is answered in this chapter is “what variables determine cloud costs and how do they relate to each other?” A literature study is performed to answer the sub question. Out of an original 498 literature items about the concept of ‘cloud cost’ 55 papers are selected as useful. Through the systematic gathering of cost variables, a taxonomy of costs is constructed. In total, 83 variables are identified from this literature set. An important observation is the fact that the vast majority of the reviewed literature only covers a small portion of the cost variables of the taxonomy. Based on a threshold set, so each variable would at least be described in five articles, only eight independent cloud cost variables can be derived. However, these eight variables are quite generic, which means that much of the 83 original variables will be wrapped up in these eight variables. To answer the research question addressed in this chapter eight variables are identified in using a systematic literature review. Important constituents of cloud costs are: cost per virtual machine, network, storage memory, energy, CPU, service level agreements (SLAs) and objectives (SLOs).

Chapter 4. Cost forecasting techniques for cloud costs

This chapter is aimed at answering the following research sub question:

b. Which cost forecasting techniques are there and how can they be used for the prediction of cloud costs?

First, the concept of cost forecasting will be explained and different techniques will be discussed. Next, the applicability of these techniques to cloud cost forecasting will be evaluated. Finally, a suitable technique for cloud cost forecasting will be selected and an example implementation will be described in detail.

The field of forecasting is concerned with approaches that increase certainty to what the future will entail, as well as the presentation and use of forecasts in practice. Different terms are found in both literature and practice which are used interchangeably, being: *prediction*, *projection* and *prognosis* which are used to describe operations to obtain a picture about possible future developments (Young 1976). In the field as well as in this work the terms forecasting method, approach and technique are used interchangeably. The research field received significant attention the past half century. Most of the methods are developed in an early stage but the application and combination of these methods in new configurations and domains is still subject of research today (Armstrong and Green 2005). Historically, fields of application for forecasting are – but not limited to: econometrics, meteorology, marketing, and bulk production. Forecasting is commonly used in conjunction with a conditional statement. An example of such a statement is: when X occurs, it is likely Y will happen as well. Forecasting approaches are developed as products of empirical research and evaluated and validated in practice.

4.1 THE CASE FOR CLOUD COST FORECASTING

Until now, IT delivery is characterized by high capital expense (CAPEX) and low operational expense (OPEX). This is due to high investments in infrastructure hardware and / or high upfront software development costs. In contrast, cloud delivery has large operational expenses and zero or few upfront investments. Another important difference between IT and cloud delivery is the way variance in user load is accounted for. Traditionally, IT infrastructure is scaled to cope with peak performance. In practice, it means infrastructure is only used for 10 % (Greenberg et al. 2008), which also means there is 90 % overprovisioning. Cloud IT delivery is based on pay-per-use accounting. This provides a strong financial incentive to minimize overprovisioning. However, for many organizations the pay-per-use character of cloud IT delivery entails uncertainty in IT spending. This negatively affects the cloud IT delivery paradigm. Organizations may choose a high CAPEX over a high OPEX configuration, even when it means paying a vast amount more, even in the long-run. In addition, the over-payment can be seen as the cost of certainty. A way to decrease uncertainty is *cloud cost forecasting*. Quality characteristics for potential cost forecast figures are accuracy, relevance, and acceptance (Weida 1977). Forecasts should be *accurately* enough for a significant number of clients in regard to that there is a decrease of uncertainty not a transition into a different kind of uncertainty, being the uncertainty of the accuracy of the forecast. Furthermore, a forecast should be accurately enough that the variance of the prediction is not greater than the range of uncertainty. For example, when company X expects to be billed $10,000, with a variance of $ 4,000,- a forecast should provide a figure with less degrees of freedom. Forecast figures have input and output that should be relevant in respect to organizational characteristics. Ideally these figures are translated to the domain of an organization. An example of it is a forecast calculation for cloud based software for municipalities that handle passports. Municipalities would ideally produce a forecast about the amount of passports within a timeframe. These kinds of inputs are familiar and understandable for the target organization. However, there need to be a strong relationship between input characteristics and system load. *Acceptance* of forecasting output is important for both client and vendor. The people that are ultimately responsible for decision making in organizations that use forecasts should be aware of the considerations made during the forecast. Whether forecasts are made by experts or by mathematically sound models, both should be interpreted with the proper constraints in mind.

FORECASTING FOR EXISTING AND NEW CUSTOMERS

In practice, cloud cost forecasting will be helpful in two scenarios. The first scenario involves a customer which has not been on the cloud platform before, it is called an ex ante forecast. This means, there will not be historical data available for the customer. The second scenario involves ex post cloud cost forecasting, which will be a cost forecast from an existing customer for which historical data is available. It is important to note, the difference between the two scenarios. Historical data about system use and organizational characteristics need to be available. Both situations are graphically depicted in Figure 6.

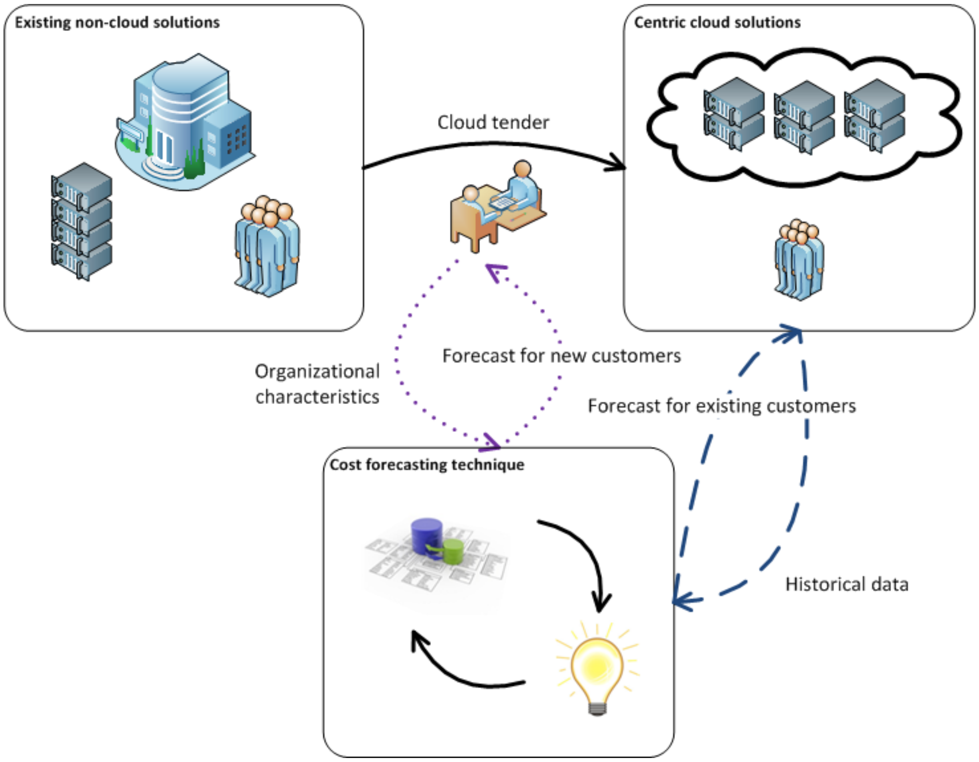

Existing non-cloud solutions
Cloud tender
Centric cloud solutions
Organizational characteristics
Forecast for new customers
Forecast for existing customers
Cost forecasting technique
Historical data

FIGURE 6: CLOUD COST FORECASTING SITUATIONS

APPLICATION OF FORECASTING

Looking back at the problem statement of the thesis research, uncertainty of future cloud costs is hindering adoption of cloud technology. Uncertainty should not be seen as a one-time occurrence, but rather as a recurring situation. The first time uncertainty, especially for customers that are not consuming a pay-per-use type of IT delivery yet can be addressed separately. In order to address the ongoing uncertainty issue for these customers, one has to offer forecasting as a service on an ongoing base. For a platform operator there can also be a desire for forecasting, as it provides the costs incurred by a customer or customer group for a software service. Furthermore, forecasts can be used as a guideline for infrastructure sizing.

CLOUD COST FORECASTING FOR A PLATFORM VENDOR

The term internal cloud cost forecasting is used when the customer that pays for software services is not exposed to cost that occurs on the infrastructure layer. A situation where this is the case, is a payroll application for which the customer is only billed for every processed pay slip. The fee per pay slip is unrelated to the influence processing of pay slips has on the infrastructure. The risk of infrastructure costs belongs to the vendor. However, the vendor that takes the risk is exposed to fluctuations in infrastructure loads. This means that, the vendor must properly size its infrastructure to be able to cope with fluctuations in demand. Over time, a vendor has fluctuating cost of infrastructure, unrelated to the level of income. These fluctuations cause a decrease in margin. Due to an increase in demand infrastructure needs to size up and costs will increase. However, downsizing infrastructure poses a problem. Vendors have to buy infrastructure components themselves which, cause long-term investments. Even when vendors do not invest in infrastructure themselves, they always respond to down-sizing reactively, which implies there will be a period where there is too much overprovisioning. The results, summarized over 52 weeks in a year, are graphically depicted in Figure 7. Infrastructure load (IaaS costs) is depicted by the blue dotted line. It displays the amount of the current available capacity that is being used expressed in a percentage, and has a corresponding scale on the left vertical axis. Infrastructure costs are depicted by the green dashed line. It expresses the amount of costs incurred on the infrastructure level to provide the sized amount of capacity by the vendor. It has a corresponding axis expressed in euros on the right vertical axis. The software as a service billed amount is depicted by the purple solid line and depicts the amount of money billed to the customer.

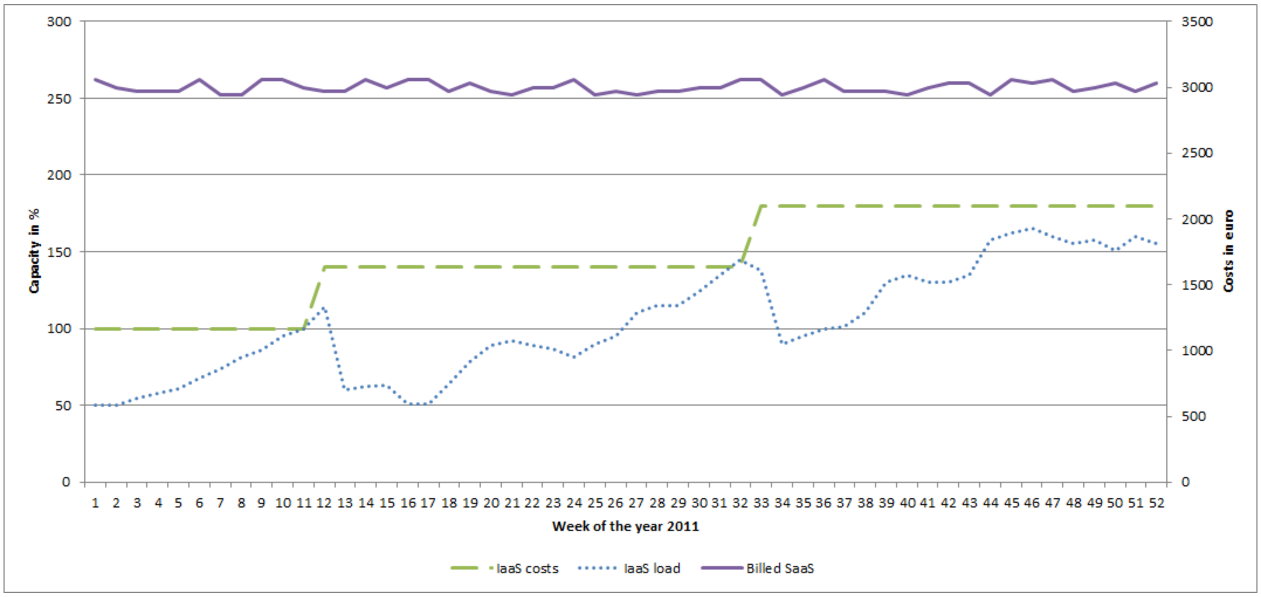


FIGURE 7: EXAMPLE SYSTEM LOAD AND SERVICE BILLING

The graph depicted in Figure 7, is based on a simplified version of reality. However, some interesting observations can still be made. Margin is decreasing, which is indicated by the two-step increase in costs, combined with a relatively consistent amount of income through *billed SaaS*. In practice this would be common. For most systems there is a constant increase of demand in storage capacity. On average, there is an average of 29 % overprovisioning. When it can be decreased in a ceteris paribus situation (when environmental conditions remain the same), it would not affect service, but will decrease costs with a similar turnover. Therefore, profit margin is increased. Re-sizing of the capacity is only performed scaling upwards. It is executed reactively, only when the actual load reaches capacity limits. Downscaling can only be performed reactively, for which delay causes a decrease in margin.

4.2 SOFTWARE BEHAVIOR ON INFRASTRUCTURE AND LOAD FOOTPRINTS

In general terms, the use or execution of SaaS for a customer causes load throughout the cloud stack. Examples of usage are a request for a sales report, but also something as trivial as a help page. The use is part of the pay-per-use agreement and is billed at an agreed rate. Only, not all SaaS applications cause an equal amount of load on the underlying infrastructure. Footprint is the amount of capacity used by software when running. Some applications have a high footprint, which indicates that the application consumes many resources, while others have a small footprint. Furthermore, applications respond to growth differently. An application with a small footprint causes a minimal load on the infrastructure but can have a steep growth model. Conversely, an application with a large footprint on the infrastructure can have a moderate growth model. The concept can be viewed as the *footprint elasticity*. This research introduces the concept of *footprint elasticity*, which can be viewed as the ratio of the percentage change in one variable on another. The relationship is shown formally in Equation 1.

$$E_{x,y} = \frac{\partial y}{\partial x} = \frac{\partial footprint}{\partial usage}$$

EQUATION 1: FOOTPRINT ELASTICITY

There is also a difference amongst customers that use a similar application. Moreover, multiple customers using the same instance of an application can cause a different load footprint. In this respect one could say that the load footprint of an application is homogenous when there is no or little difference between customers, or heterogeneous when there is much difference. While this insight is not groundbreaking, the homogeneity influences what forecasting techniques can be employed as it determines to what respect the data of one customer might also be generalizable to other customers.

4.3 Different cost forecasting techniques

As the field of forecasting can be seen as mature, it is useful to adopt a classification of different forecasting techniques as a frame of reference. The classification of Young (Young 1976) are generic, they classify forecasting techniques in two dimensions, these are the function or purpose of the forecasting technique, and second is orientation or behavior. The function can either be operational, where the output of the forecast is contingent to a range of possible outcomes or analytical for which the output is expectation (like with trend prediction). The orientation can either be normative where the output is accepted based on the forecast itself or positive where the outcome is evaluated under a set of ceteris paribus assumptions. A drawback of the classification is that it seems constructed to map existing techniques, however offers little help in the selection of a technique. A different taxonomy that is not limited to two dimensions and provides support for forecasting selection is that proposed in (Armstrong and Green 2005), which is also reviewed as the better classification in a comparison of five popular classifications (Gentry, Calantone, and Cui 2006). For the research, this classification is adopted in order to explain how different forecasting techniques might be used to predict cloud costs.

There are two distinct differences in forecasting techniques (Armstrong and Green 2005). Figure 8 depicts eighteen forecasting techniques in a classification of type and application. Judgmental forecasting is performed when there is no quantitative data available or qualitative data is likely to produce a more accurate or acceptable result. The focus of this chapter is on forecasting techniques based on data. These techniques are marked by the blue box. The quantitative analysis of dependencies between variables is called regression analysis. The discovery of the relation is done through observation of changing *dependent variables* by influencing one or more *independent variables*. In this research, the dependent variable is explained as the cost of the used service.

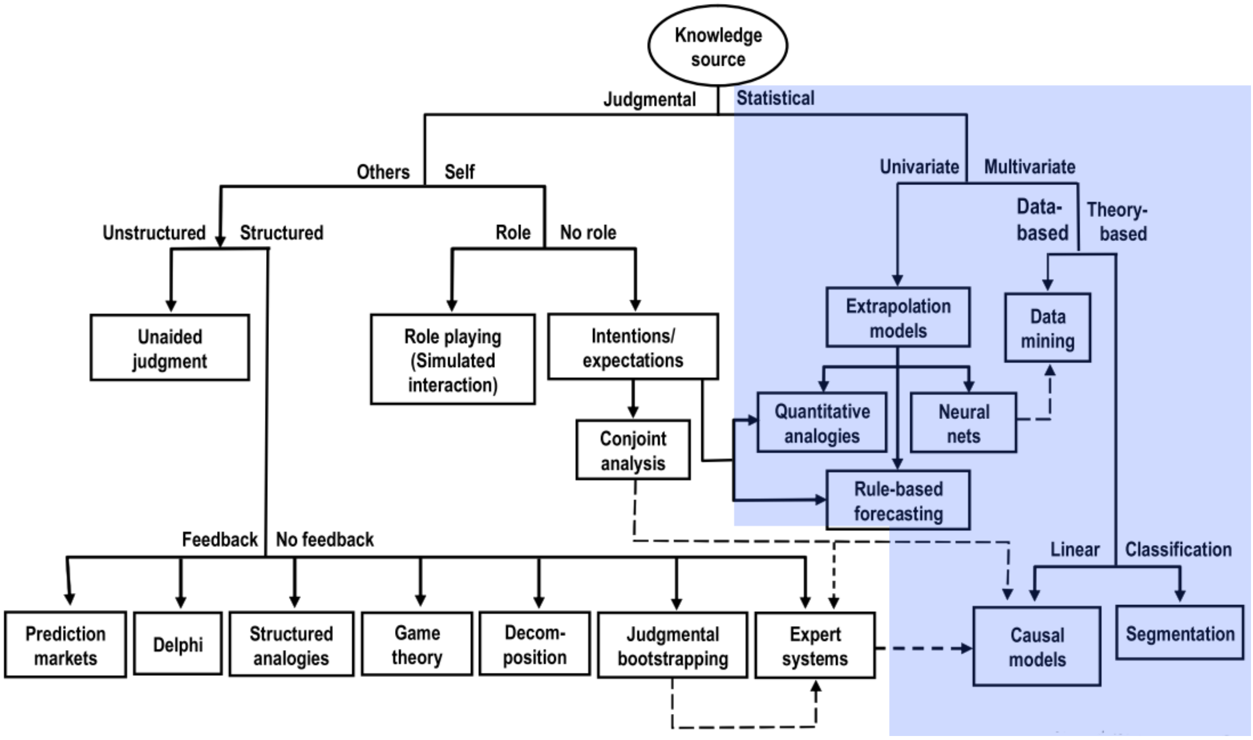


FIGURE 8: TAXONOMY OF FORECASTING TECHNIQUES

Data used for forecasting can be gathered from within the organization or within the market. An *inward focus* on data gathering means that historical data or organizational knowledge is being used in the forecast. An example of it is the historical sales volumes of a product. An important notion here is that internal data must have a causal relationship on the predicted dependent variable. However, it is not always the case. An *outward focus* on data gathering means that environmental factors are being used in the forecast. An example of this can be market growth figures or demographic figures. In order to contribute to accurate forecasting the data should have a causal relationship on the predicted dependent variable. The concept of *time-series* recognized as important among the data intensive forecasting techniques (Brillinger 2001). Time series represent a sequence of data points observed over a given time interval. Typical application of time-series is the creation of a trend line, based on passed occurrences.

EXTRAPOLATION MODELS

Extrapolation models use historical data on that which one wishes to forecast. Extrapolation means to construct new data points outside existing ones (Makridakis, Wheelwright, and R.J. Hyndman 1998). The results of these new data points are as strong as the relationship between past, present, and future occurrences. Exponential smoothening is a popular approach for situations, where the present past is more representative than the more distant past. Exponential smoothing *weighs* recent data more heavily in respect to old data. It has a flattening effect on fluctuations on data from a more distant past.

For the application of *extrapolation models* for cloud cost forecasting historical data needs to be available. A precondition for the application of extrapolation models is that there needs to be a strong relation between past occurrences – and future occurrences. An example of it can be: seasonality in system loads or recurring quarterly number crunching for financial reports. For new customers there is often too little data available about past service usage. Furthermore, when a customer transits from an on premises situation to the cloud, it is not likely that the historical data will be representative for, and compatible with, the new cloud situation.

For existing customers historical data can have a significant contribution to predictions. Due to the pay-per-use character of cloud services bills will likely reflect the usage flow. Whenever the flow has a recurring pattern over the long term, or a steady short term trend line, historical data can provide accurate forecasting information. In practice, short-term trend analytics are already being used in load-balancing software (Ardagna et al. 2012; Bossche 2010; Gmach and Krompass 2005; Mian, Martin, and Vazquez-Poletti 2012).

Extrapolation models cannot be used for long-term prediction when there is only little historical data. As a rule of thumb, for any given time frame, there need to be at least an equal amount of historical data. When forecasting seasonal influences, there need to be at least two seasonal cycles pattern in the data set. However, in regard to the exponential smoothening technique one has to be aware that seasonality in a distant past will be weighed lighter than recent seasonal differences. However, extrapolation models are not limited to the data generated by the customer itself, as data can also be derived from other customers. This idea will be discovered further in the *quantitative analogies* section.

Quantitative analogies

Quantitative analogies represent situations that are analogous to a given situation. Sometimes, experts are used to identify analogous situations. Quantitative analogies are useful when there is no, or little, data available about a given situation but extensive data on a similar situation. An example of it is the case where there is a loss in sales in a given country that is analogous to a previous situation in a different country. Another example of this is an organization that produces promotional goods for soccer matches in The Netherlands. The organization will likely have production that correlates with significant event like the European and World championships as well as the Olympics. Furthermore, these external data sources can be gathered but some can be bought from for example market researchers.

The basic assumption in the use of quantitative analogies is that a blueprint of a given situation can be matched upon a generalized and representative (group of) individuals. The trick is to construct the proper blueprint by using the suitable analogies. The identification of analogies is an important step in the process. One way to discover suitable quantitative analogies can be through experts. It seems natural as a common first approach to come up with a reasonable cost price seem to be to ask experienced people whom will remember past and similar situations. A different manner to discover non-evident connections is through *data mining* techniques, which is discussed later on in the section.

For the application of *quantitative analogies* for cloud cost forecasting similar occurrences have to be found. These analogies can be drawn based on numerous events. However, a significant correlation with historical and present events that can be used to predict the future is likely to be not that far-fetched. However, it is more likely to exist between business units, competitors in the same markets, and users of the same and similar software service. For a cloud service vendor it has the potential to offer forecasting services based on its user base as an added value service. Users of the same software services will presumably have some common ground. Especially in the case where services are specialized to target market or customer groups. For such a market, forecasts can be based on information normally not available to a single customer.

Rule-based forecasting

Rule-based forecasting (RBF) allows heuristics and expectations to be incorporated into time-series data analysis, effectively decreasing the degrees of freedom in the prediction. The advantage of rule-based forecasts is, that it enables an organization to translate domain knowledge or expert knowledge into a set of rules. Rule-based forecasting can yield substantial accuracy when there is of domain knowledge available, patterns are tangible, trends are strong and forecasts are made over substantial time. Rule-based forecasting within these conditions have shown significant less susceptible to errors than combinations of multiple forecasts (Adya et al. 2001).

The application of RBF for cloud cost forecasting has the potential to reduce errors when there is substantial knowledge that can be transformed into a set of formal rules. The rules can be derived from experts, this way existing knowledge from within an organization can be put to good use and guide the forecasting process. In practice RBF can perform as a means of trend analysis, however more importantly it can be used as an addition to other forecasting techniques.

An example of RBF rules can be a minimum server capacity that is required to keep the software alive, limiting the degrees of freedom of a forecast. In practice RBF prediction would respect this rule and will not forecast below the minimum server capacity, even when input figures would otherwise implicate this.

Neural nets

Neural networks are computation intensive methods that mimic the decisions process analogous to those of humans. Input changes the structure of the net. This is known as *learning.* Just like it would happen with human neural networks (G. Zhang and Patuwo 1998). The main difficulty with neural network is that the learning process causes difficulty in understanding how predictions come to be. Neural nets also need stimuli to be useful which might not always be available. Especially when input factors can be recorded or replicated there is little chance of creating an identical neural net. Thus, replication of conditions that led to a prediction. At the moment it disqualifies neural nets for the application of cloud cost forecasting.

Data mining

Data mining is a popular approach for identifying relationships or patterns using sophisticated statistical analysis in large data sets. Data mining ignores theory and prior knowledge in the search for patterns and lets numbers and mathematics do the work. In recent years, data mining had gained significant interest from organizations that process and store large amounts of data like Facebook and Google (Witten, Frank, and Hall 2011). An important concern for data mining techniques in relation to the ability to forecast is the lack of real evidence of success in the field (Keogh and Kasetty 2003). Moreover, most data mining techniques are rarely validated against significant amount of datasets. Data mining is an approach to creating useful insights out of existing data, for which forecasting is one of the possible applications (Fayyad et al. 1996). Knowledge discovery through data mining can be performed using different techniques discussed in this section, but it is important to recognize that it is not a distinct forecasting technique itself.

Causal models

Causal models are based on prior knowledge and theory about the causal and deterministic relationship between two events in an abstract nature (Pearl 2003). The key is to limit the degrees of freedom and increases the accuracy of the prediction. Causal models are most useful when strong causal relationships are expected, the direction of the relationship is known, causality can be determined and large changes are expected to incur in the causal or independent variable (Armstrong and Green 2005).

In practice, *causal models* can be used to predict statistical associations. In other words, the association should reflect a cause-effect type relation where the cause explains the effect. It works best when time is one of the variables for which you can just put in a date for the forecast. In general, business applications like ERP systems show peaks at office hours, quarterly reports and lows during the holiday season (Lindner, Galán, and Chapman 2010). A practical application of an association would be to use time of day and *office hours* as statistical associations.

When time is not a variable in the statistical association *causal models* can still be used. An example of a causal model could be an application for which user licenses need to be paid to an external party. The amount of people that have access to the application would then be an independent variable to forecast high license fees as the dependent variable.

4.4 MEASURING AND IMPROVING FORECASTING PERFORMANCE

Research has led to advanced forecasting techniques with significant success. However, forecasting remains a practice of estimation with uncertainty. In practice, almost all forecasts will be of to some degree. Important aspects in this respect are tools that measure variance and error in forecasts. Forecasting performance can be defined using several measures. These all calculate a metric about the difference between the observed and the actual value, using a_t as actual value and f_t as forecast value. The formula to calculate it is:

$$Error\ (E_t) = a_t - f_t$$

Mean error (ME*)* is the simplest rate of error, which depicts the average error of the forecasts. The formula to calculate it is:

$$ME = \frac{\sum_{t=1}^{n} E_t}{n}$$

Root mean squared error (RMSE) is the square root of the variance measured by the *mean absolute error*. A RMSE of zero means a perfect forecast, although in reality this is unlikely to happen. This standardizes the value which makes it comparable. The formula to calculate it is:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} {E_t}^2}{n}}$$

Mean absolute error (MAE) is the average of the absolute value of the difference between forecasted value and the actual value. The formula to calculate it is:

$$MAE = \frac{\sum_{t=1}^{n} |E_t|}{n}$$

Mean percentage error (MPE) is the computed average of percentage errors by which forecasts values differ from actual values. The formula to calculate it is:

$$MPE = \frac{1}{n}\sum_{t=1}^{n} \frac{f_t - a_t}{a_t}$$

Mean absolute percent error (MAPE) is the average of the absolute value of the difference between the forecasted value and the actual value explained as a percentage of the actual value. The formula to calculate it is:

$$MAPE = \frac{\sum_{t=1}^{n} \left|\frac{E_t}{a_t}\right|}{n}$$

Mean absolute scaled error (MASE) is the average error independent of scale. While most error rates indicated errors based on the size of the amount of observed values E_t this makes comparing time-series with differing lengths difficult (Rob J Hyndman 2006). The MASE value can be used to do precisely this. The formula to calculate it is:

$$MASE = mean(\left|\frac{E_t}{\frac{1}{n}\sum_{i=2}^{n}|a_i - a_{i=1}|}\right|)$$

It stems to reason that the forecasting accuracy is related to the granularity and availability of both a cost model and historical data. When data is non-specific forecasts become meaningless which will result in unusable estimates. Having too much data is a lesser problem. Most forecasting techniques are capable of smoothening data and are also able to recognize unwanted side-effects of large datasets, like trends convergence towards the mean for large datasets. It should be the goal of the cloud operator to retrieve specific information about system usage in order forecast costs more accurately. However, in most cases the range of the forecasted, results in a prediction of an increase or decrease of one or more virtual servers.

SEGMENTATION

Segmentation involves the independent breakdown of the problem into independent parts (Armstrong and Green 2005) using prior knowledge or theory about the data. Using segmentation it is possible to diminish the effect of variables that obscure an otherwise strong correlation. An example of a strong correlation is demand forecasting for surfboards, where age as well as proximity to the beach are likely to both have influence. However, the two variables do not necessarily agree with each other. Segmentations can have a priority with cut-points. The more data is available, the more core cut-points should be made. For each variable, cut-points are determined in a manner that the stronger the relationship with the dependent variable, the greater the non-linearity in the relationship. Forecast should be made for the population of each segment and the corresponding behavior of the independent variable.

4.5 CHOOSING THE PROPER FORECASTING MODEL

It seems apparent that for a concept as broad as cloud IT delivery many different forecasting techniques can be applied. However, in a given situation some will be more appropriate than others. Furthermore, the requirements for cloud cost forecasting are the availability of a cost model, and historical metering data. The cost model should incorporate all variables that are used in the costing process of the vendor or operator of the cloud service. Historical metering data reflects past cloud service usage. As a general rule the forecasting horizon should not be greater than the set of historical data. Seasonal effects in data will start to emerge after two full seasons (Kalekar 2004). Which forecasting techniques is used, depends on the type of application, and the amount of data available. Table 9 describes the requirements on the data for every forecasting technique.

	Extrapolation models	**Quantitative analogies**	**Rule-based forecasting**	**Segmentation**	**Causal models**
Forecasts based on	Historical data	Similar past and current occurrences	Expert domain knowledge translated into rules	Comparison and grouping	Cause and effect type relations (like bundles goods)
Historical data required as input	Time-series. More is better.	Time-series of analogous situation.	Mostly the current state.	Multiple environments, more is better.	Time-series of multiple environments.

TABLE 9: COMPARISON OF FORECASTING TECHNIQUES

Applications that have a similar load footprint at different customers (homogeneous) have the advantage that other customers can be used as a benchmark for the customer at hand. In contrary, an application for which the footprint is influenced heavily by the customer's usage has its own unique situation, with a corresponding unique footprint.

The matrix depicted in Figure 9 can be used for decision making purposes. For each application and customer one should classify the availability and relevance of historical data for its particular situation and the type of application.

	Homogenous application load footprint	Heterogeneous application load footprint
No historical data for customers	Quantitative analogies	Rule based forecasting Segmentation forecasting
Existing data for customers	Quantitative analogies Causal models (multiple customers) Extrapolation models	Causal models Extrapolation models

FIGURE 9: FORECASTING TECHNIQUE DESCISION MATRIX

Sometimes multiple forecasting techniques can be applied. As forecasting is the result of an empirical learning process, continuous improvement is an important aspect. Another manner to improve forecasting accuracy, and reduce error is the combination of different forecasting methods. In 30 empirical comparisons an average increase of 12,5 % was achieved (Armstrong 2001). Furthermore, the application of multiple forecasting techniques which differ substantially from each other, will lead to an additional increase off accuracy.

4.6 DISCUSSION OF RESULTS

This chapter covers a range of forecasting techniques and their application for cloud cost forecasting. This chapter discusses literature about forecasting techniques, covering the last six decades. An assessment of the applicability for each forecasting technique, including its potential for the prediction of cloud costs is provided. Forecasting examples are provided for two customer groups, being both new customers and existing ones. For some techniques different types of application are provided. Accuracy relevance and acceptance of forecasts are taken into consideration. As well as how forecasting performance, and errors can be measured. Finally, a decision matrix is constructed that matches forecasting techniques to its corresponding situation and application.

Chapter 5. Cloud cost forecasting in practice

This chapter is aimed at answering the following research sub question:

c. How can cloud cost variables and a forecasting technique be integrated in a technical solution that predicts cloud costs?

This chapter discusses design elements for a prototype cloud cost forecasting tool that incorporates three components: a cost model, historical metering data, and one or multiple forecasting techniques. The primary objective of the technical solution is to provide a prediction for a given forecasting horizon.

The structured literature review in chapter two has shown that a holistic cloud cost forecasting solution does not yet exists. The aim of this chapter is to provide a preliminary technical solution that can be used in practice. The technical solution is designed as an automatable solution because cloud technology is built to thrive on economies of scale (Asoke K. Talukder 2010; J Hamilton 2010; Sekar and Mantis 2011). This means that a forecasting solution would be made available to both the platform operator and the customer. The relatively immature and practical nature of this solution poses limitations which are discussed in their respective sections.

Cloud products cannot yet be considered as mainstream (Pettey and Meulen 2012) which means that cloud operators are still establishing best practices and standardizing their products. As cloud products do not have a long track record, datasets with historical information tend to be limited to months instead of years. This makes it difficult to create a forecasting solution based on data from a large customer group. For a platform vendor who is hosting numerous applications that are heterogeneous in nature, little comparison between applications and customers is possible. The lack of comparability causes forecasting to be based solely on historical data from a single customer. Therefore, only the application of the extrapolation forecasting technique is discussed in this section. Another difficulty due to the limited availability of historical data is that forecasting horizons will be limited to the near future.

5.1 A COST MODEL IN A TOOL

Chapter three indicates that in cloud environments the majority of the costs are derived from usage. The majority of the cloud products are offered as a pay per use product (Ibrahim, He, and Jin 2011). Usage can be defined as the capacity required while running software on the infrastructure. However, not all cost will be related to the actual physical use of infrastructure. An example is the variety in software licensing models: license fees may be based on the usage of hardware such as processors cores or on the number of concurrent users. This makes licensing costs one of the most difficult sources of cost to predict. Software license constructions need to be reevaluated because “the licensing model for commercial software is not a good match to Utility Computing” (Armbrust et al. 2010). It also forced organizations to favor open source software in contrast to commercial products. In an early stage of cloud acceptance it will be likely for “commercial software companies to change their licensing structure to better fit Cloud Computing" (Armbrust et al. 2010). Therefore, costs that cannot be calculated through a cost driver related to infrastructure are excluded from the technical solution.

The costs incurred by a platform vendor for operating cloud services can be described for the IaaS, PaaS and SaaS levels. These levels are built on top of each other. The decomposable architecture makes it possible to determine costs for each layer. The costs for each layer on its turn are passed through to the top layer or *rolled up*.

The costs for each layer comprises a recurring set with five generic sources of costs; these are *costs of goods sold*, *labour*, *investments, rolled up costs*, and *indirect costs*. Costs of goods sold are the costs of the actual purchase of the resold products (e.g. licences, CPU's, server chassis). Labour costs comprise of personnel cost (e.g. cost of operating and maintaining the platform). Investments comprise costs that are purchased and depreciated over a fixed time period (e.g. hardware). Rolled up costs are costs rolled up from a layer below. Indirect costs are the costs which are difficult to allocate to a specific driver for which a percentage calculated (e.g. housing). The multi-layered cost structure can be translated into a programmable structure. A class diagram that depicts the structure is depicted in Figure 10.

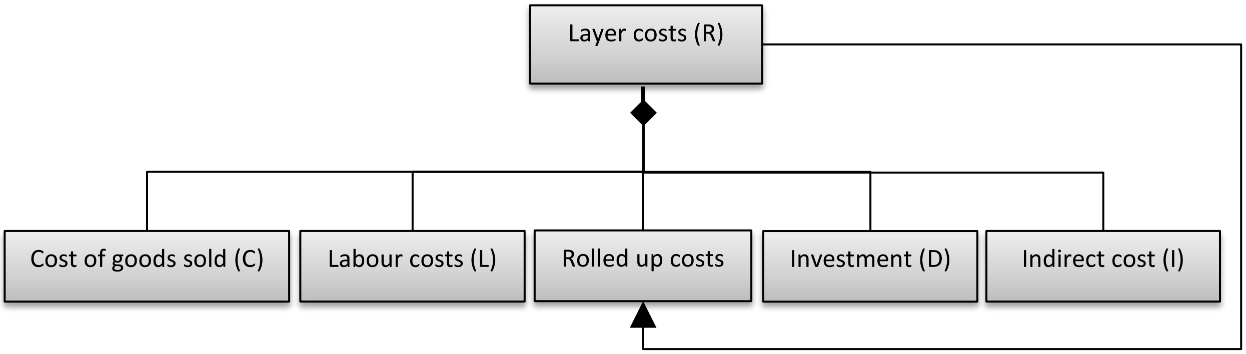


FIGURE 10: CLASS DIAGRAM FOR A MULTI-LAYER COST STRUCTURE

When a cloud operator sells software as a service and has its own platform and infrastructure layer in-house three cost layers are constructed that together depict the costs of the software as a service. An equation to calculate this is depicted in Equation 2. Whereas *C,L,D,I* are the cost variables and *R* is the portion of the sub layer costs. Where superscripted *S*,*P*, and *I* denote the different layers in the example equation.

$$SaaS\ costs^{S} = C^{S} + L^{S} + D^{S} + I^{S} + R^{P}\ (C^{P} + L^{P} + D^{p} + I^{p} + R^{I}\ (C^{I} + L^{I} + D^{I} + I^{I})\ )$$

EQUATION 2: COST MODEL

5.2 HISTORICAL (METERING) DATA IN A TOOL

The second component in a cloud cost forecasting tool is the ability to store historical data. As a bare minimum, data needs to be stored about virtual machines and storage. A technical specification of every virtual machine with the amount of cores, their respective speed, combined with performance data of past usage. A possible way to retrieve these figures is through the virtualization software. Most of the players like VMware, and Oracle VM VirtualBox have options to make performance statistics available on the physical and virtualized level (Oracle 2011; VMware 2007).

In addition to measuring on the infrastructure level, it is also important to measure activity on the software level. In some cases it would also be the figure on which the billing is based. This activity should be reflected in a figure that represent the service the software is offering, like a transaction in a case application or a hit on a request from a web browser for a website. For a pay slip application, it can be the amount of pay slips processed.

The intervals at which these statistics are derived have great influence on the predictability of the independent variable. In practice, most tooling uses consolidation algorithms to consolidate historical data as these metering data can grow rapidly. Nevertheless, despite these algorithms the statistics of a single host in VMware can grow up to one gigabyte for one year (Hsiao and Y.-Z. Wang 2007). There will be a trade-off as far as to the precision of the interval for which historical data is kept. VMware uses a five minutes interval on daily base, a 30 minutes interval for weekly data, a two hours interval for monthly data and a daily interval for data older than one year. This consolidation process is necessary, because unconsolidated statistics will negatively affect VMware vSphere's performance. However, this influences the forecasting abilities, as the granularity of the prediction is influenced by the availability of data. An example of it would be a municipality that distributes passports to citizens. In Holland, there is a large peak in requests for passports before the summer holiday. If one would forecast on a weekly level there would at least be daily data available. However, the consolidated data neglects that municipalities primarily work during office hours. A mathematical average does not always provide enough information. In some cases an hourly interval is required for a prediction. However, the effect will decrease when forecasting is performed for larger clusters, covering several hosts, because the peaks in loads on individual servers will even out over the whole.

A forecast using a granularity at an hourly level is a challenge. The amount of data required to do so is considered costly to store and process. The benefits achieved with an hourly forecast are limited as infrastructure also has overhead time to instantiate (L. Wang and Zhan 2009). In theory it would be possible to store each month's hourly statistics in an external database and use the data for forecasting purposes. However, for cost forecasting purposes a daily level would be sufficient.

In this research, several cost variables have been identified as determine an important portion of the costs for cloud systems. Based on these variables, corresponding measurements from the VMware statistics database are selected. These measurements are then consolidated into clusters of virtual servers with identical hardware. vSphere statistics have a name, interval, unit of measurement, and value (VMware 2007). A detailed design of the relevant tables from the VMware database can be found in Appendix F.

5.3 FORECASTING TECHNIQUES IN A TOOL

Chapter 4 covers an extensive overview about the application of a diversity of forecasting techniques for the prediction of cloud technology. Although a diversity of forecasting techniques have potential, extrapolation forecast has the potential to forecast costs for existing customers for any type of applications that is run in the cloud. Due to the broad applicability of extrapolation forecasting it is chosen to be included in the technical solution. Historical data, measured on time intervals, called time-series are the input for extrapolation forecasts.

In order to assess the influence of trends and seasonality on a forecast, two options are available. Either an expert assesses the data for seasonal and trend influences, or an algorithm is used to make a computational assessment about the dataset at hand. Fully automatically forecasts can also be performed on datasets using algorithms that detect seasonality and trends (R.J. Hyndman 2008; Rob J. Hyndman et al. 2002). The method described by Hyndman et al. (Rob J. Hyndman et al. 2002) uses a state space framework for the automatic selection of exponential smoothening techniques for forecasting. The framework makes an assessment of best fit – comparing error values – by choosing amongst twelve different types of exponential smoothing using multiplicative and additive error models. In effect, the 'best' technique amongst 24 different options is chosen and applied. Although automated forecasting will never provide the best available forecast for every situation, it has proven itself in the 3003-series of the M-competition in a comparison where 'our method performs better than the others shown for shorter forecast horizons, but not so well for longer forecast horizons' (Rob J. Hyndman et al. 2002). Although the comparison is the result of a large scale study, it does not disqualify the algorithm for use in practice. Moreover, when forecasting is performed solely on distant forecasting horizons, there are other algorithms to consider.

Both manual and automatic selection of a forecasting algorithm is also possible (Rob J. Hyndman et al. 2002). Three options need to be evaluated, being seasonality, trend and the manner how errors are handled. For datasets spanning several months, seasonality might influence system load. However, seasonality in data will be influenced by the type of organization and software that is ran on the infrastructure. When seasonal influences are not applicable or should not be taken into account, simple linear extrapolation with exponential smoothening is the right pick using the *ets* function in the R software (Ryan and Ulrich 2012).

The final consideration that needs to be made is determining the forecasting horizon. Although in theory there is no limit to the forecasting horizon, the accuracy of the prediction is decreases when the forecasting horizon increases. Forecast horizons covering periods greater than the data collection interval will therefore be unreliable, as trends and seasonality take time to manifest itself in the data. The evaluation section below, different measurements of accuracy are presented. The output of a forecast can be just a single number for a specific date, a sequence of numbers for consecutive dates but also forecast plots. All have descriptive output like multiple predictions with difference confidence intervals, the chosen extrapolation algorithm, multiple error measurements, and parameters which are used for smoothening and trends. The output can be part of a periodical reporting about the rendered services.

5.4 A CONTROLLED EXPERIMENT TO FORECAST CLOUD COSTS

This section discusses a controlled experiment in order to derive quantitative results about the power of forecasts for cloud costs. Guidelines aimed at heterogeneous reporting of experiment are applied in this section (Jedlitschka, Ciolkowski, and Pfahl 2008a). However, since this section is part of a much larger thesis only the reporting guidelines on the content of the experiment are adhered.

CONTEXT OF THE EXPERIMENT

The experiment should be performed with data from a system that is in production as the purpose of the experiment is to prove or disprove the applicability of forecasting in practice. Access to such a production environment should be granted and details about the software installed on the servers should be made available. In order to run the experiments the environment should be live for some time which can be anything over three months. However, the more samples gathered the better. The data used in the experiments should be 'frozen', in this respect the experiments should be performed on a dataset that does not change anymore.

EXPERIMENT PLANNING

GOALS

The objective of the experiment is to analyze the accuracy of forecasted cloud usage for the purpose of assessing the predictive power with respect to the used dataset and forecasting technique from the point of view of infrastructure usage in the context of infrastructure as a service.

TASKS

To enable replication of the experiment by others (Jedlitschka, Ciolkowski, and Pfahl 2008a) a research protocol was created that is comprised of the following consecutive steps:

1. A dataset needs to be obtained that describes infrastructure usage. The environment should be described in detail.
2. A daily average can serve as a guideline for a sampling strategy. When the dataset contains more fine-grained statistics (e.g. on an hourly level) averages can be created in the preparation phase.
3. The datasets need to be transformed to key-value 'comma separated value' (CSV) files using the data as key and the average measurement as a value.
4. The datasets are imported as data source in R. Forecasting libraries are downloaded from the central repositories and imported.
5. Forecasts are performed using the *R forecast package* (R. Hyndman, Razbash, and D. Schmidt 2012).
6. Forecasts are analyzed visually as well as numerically.
7. Interpretation of the results.

All steps should be performed sequentially. For the purpose of this experiment hypothesis are created which should guide the interpretation of the experiment.

Hypothesis and variables

The purpose of the experiment described in this chapter is twofold. First, the experiment involves the creation of an IT artifact which serves as a catalyst for knowledge creation as part of the design science research strategy (Hevner and Salvatore T March 2003). Secondly, the purpose of the experiment is to compose and test fundamental forecasting theory in the context of cloud costs. The theory is formulated as a set of three hypotheses:

- $H1_0$: The accuracy of cloud costs forecasts decreases with increasing forecasting horizon but is adequate to predict two months ahead.
- $H2_0$: The automatic forecast model selection provided by the forecasting function of the R forecast package returns the most accurate forecasting model for cloud cost data.
- $H3_0$: The accuracy of forecasts increases with the size of the datasets.

The variables that are monitored are described using a variable table. Definitions of the statistics used in the experiment are depicted in Table 10.

Name	Type of variable	Abbreviation	vSphereID	Scale type	Unit	Range
CPU usage	Independent	CPU	6	Numeric	Megahertz	$\|\mathbb{N}\|$
Memory consumed usage	Independent	MEM	98	Numeric	MegaBytes	$\|\mathbb{N}\|$
Storage usage	Independent	STORE	125	Numeric	KiloBytes	$\|\mathbb{N}\|$
Network usage	Independent	NET	143	Numeric	KiloBytes	$\|\mathbb{N}\|$
Forecast horizon	Independent	FH	N/A	Numberic	days	$\|\mathbb{N}\|$
Forecast CPU	Dependent	Fc(CPU)	N/A	Numeric	Megahertz	$\|\mathbb{N}\|$
Forecast MEM	Dependent	Fc(MEM)	N/A	Numeric	MegaBytes	$\|\mathbb{N}\|$
Forecast STORE	Dependent	Fc(STORE)	N/A	Numeric	KiloBytes	$\|\mathbb{N}\|$
Forecast NET	Dependent	Fc(NET)	N/A	Numeric	KiloBytes	$\|\mathbb{N}\|$

TABLE 10: SCHEMA OF VARIABLES

Analysis procedure

In addition to the tasks of the experiment an overview of the analysis procedure is provided below. Additional contextual information about the dataset should be described. The purpose for this is that a reader should be able to assert the external validity of the findings (Jedlitschka, Ciolkowski, and Pfahl 2008a). An overview of descriptive statistics should be compiled that describes the dataset under consideration. For each forecast, a graph is constructed on which the forecast is plotted, including the data on which is it based. The forecast model on which the forecast is based should be explained. The Forecasts graphs are analyzed visually. The plotted dataset is described as well as the forecast. The forecasts are analyzed numerically in terms of error measurements.

Comparisons of forecast are structured using tables to aid analysis. The final analysis step is the evaluation of the hypothesis proposed in this experiment.

Experiment execution

Preparation

The dataset used in the experiment is derived from a production environment using seven physical hosts on which 51 virtual machines are installed. Four hosts have two Intel® Xeon® L5630 quad-core CPUs, and the other three have one. The infrastructure houses an application that performs tax processing. This application can be considered as a business application. For confidentially reasons, a full report on the customer and the internal workings of the applications has been omitted from the report.

There is a great diversity in software products that are capable of producing extrapolated measurements. For the purpose of the experiment, the scientific mathematics software R is used (R Foundation 2012). This software is extend with a forecast library (R. Hyndman, Razbash, and D. Schmidt 2012). For the purpose of this study an export of the VMware vSphere database is created using a sequence of SQL scripts. These scripts and application guidelines can be found in Appendix D: Forecasting data retrieval. The resulting dataset is then imported into the R workspace. Dates are converted to the format recognized by R. Next, a time series object needs to be created. It is performed by providing a key/ value formatted list and parse it to a time-series object. This is performed by using an additional library which packages the *xts* function (Ryan and Ulrich 2012). The time-series object needs a start date which is set by passing the year, and day of the year of the first data point of the time-series object. In order to complete the time-series object it requires a notion of frequency of observation per unit of time, an example is 24 observations per day.

In order to test the automatic selection of forecast models, an experiment is performed using the manual selection of nine different extrapolation models. By using the table, a comparison can be made between the different extrapolation models. The table displays the descriptive error measurements common for forecasting. For the purpose of the experiment the forecasting model selection for the CPU usage forecast is compared. Only multiplicative errors are assumed as a valid candidate. The reason for this is that errors should be expressed as a percentage of the true value, not as a constant. The R scripts used in the experiment can be found in Appendix E.

In order to test the accuracy of forecasts, forecasts are performed using different forecasting horizons. Accuracy describes the degree to which the forecasted values differ with the actual observed value. Using the identical dataset as described previously in this section, the final month is used for actual observations and forecasts are performed on the dataset covering March, until August the 10th. Forecasts are performed using different forecasting horizons (7 days, 14 days, 21 days, 28 days, and 31 days). In order to compare the effect of the dataset size the experiment is performed a second time only this time two months are removed from the initial dataset. Forecasts are performed using different forecasting horizons (7 days, 14 days, 21 days, 28 days, 31 days, 50 days, and 62 days). The forecasted values are recorded and compared with the actual values of the dataset. The R scripts used in the experiment can be found in Appendix E.

Deviations from the experiment plan

The first two weeks of the data have been removed from the dataset, as the environment was not yet in production at that time. This can be considered part of the dataset cleaning process. The statistics measured during the pre-production phase do not affect the experiment.

Analysis

The analysis is performed according to the proposed procedure. The resulting dataset spans a period of 196 consecutive days of recorded statistics by VMware vSphere 5. Descriptive statistics of the dataset over these statistics can be found in Table 11.

	CPU	MEM	STORE	NET
Minimum	1253	165100	97	8
1st Quartile	6974	221000	6620	4704
Median	11380	230300	12770	8438
Mean	9676	227700	14300	9866
3rd Quartile	12390	239200	17820	10700
Maximum	15240	258200	54040	34300

TABLE 11: DESCRIPTIVE STATISTICS

For each variable – described in the columns of Table 11 – a forecast model is constructed using a 31 days forecasting horizon. The forecasting was configured to choose the best forecast method for the data automatically (Ryan and Ulrich 2012). The frequency variable is set to one, as the dataset has a daily interval. The forecast is indicated by the blue line surrounded with orange and yellow area.

Forecast on CPU

The graph depicted in Figure 11 depicts usage of CPU on the vertical axis and the days within the year 2012 on the horizontal axis. In the forecast, trend and seasonal effects are taken into account. Error handling is performed multiplicative which means that variations in predictions occur according to a statistical normal distribution and are proportional to the data. The equation used to calculate the forecast is described in Equation 3. In each case, l_t denotes the series level at time t, b_t denotes the slope at time t. α And β^* are constants.

$$l_t = \alpha y_t + (1 - \alpha)\, l_{t-1} b_{t-1}$$

$$b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)\, b_{t-1}$$

$$\hat{y}_{t+h|t} = l_t + h b_t$$

EQUATION 3: FORECAST WITH MULTIPLICATIVE TREND

Visual inspection of the graph shows a positive trend in combination with a possible seasonal pattern which is quite weak and therefore is ignored. The blue line represents the actual forecast with an increase in slope. The CPU usage in the dataset clearly indicates a steady trend, although there are fluctuations which seem to be of a repetitive kind spaced weekly.

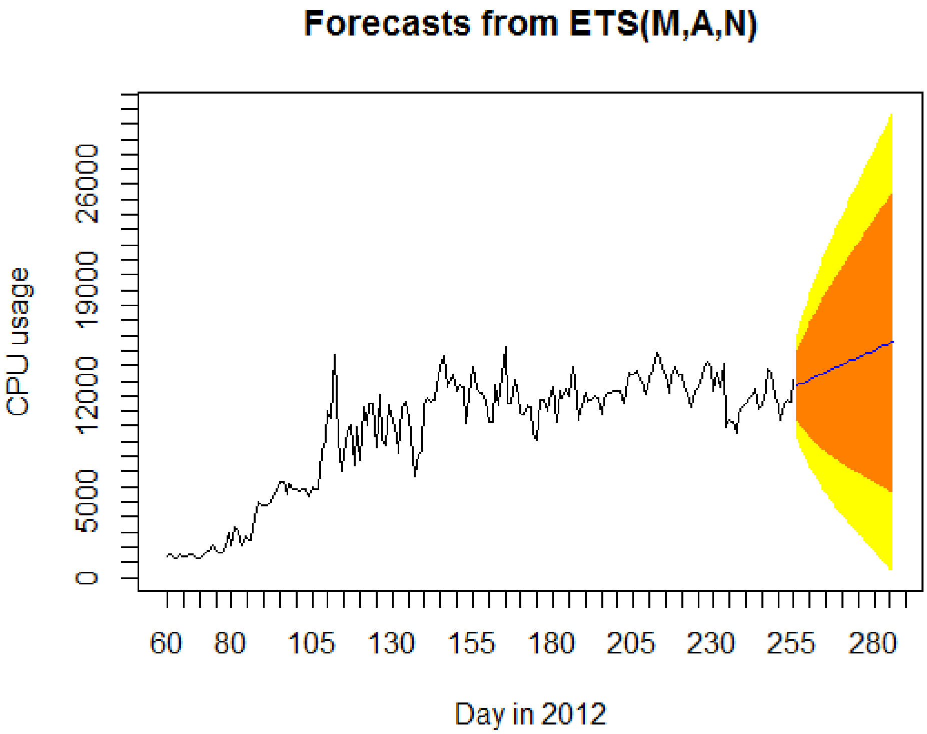


FIGURE 11: FORECAST ON CPU

Forecast on memory

The graph depicted in Figure 12 depicts usage of memory on the vertical axis and the days within the year 2012 on the horizontal axis. In the automatic forecast, trend and seasonal effects are taken into account. Error handling is performed multiplicative as well as trend, which means that variations in predictions occur according to a statistical normal distribution and are proportional to the data. The equation used to calculate the forecast is described in Equation 4. In each case, l_t denotes the series level at time t, b_t denotes the slope at time $t.$ α And β^* are constants, together with the dampened trend $\emptyset h = \emptyset + \emptyset^2 + \cdots + \emptyset^h$.

$$l_t = \alpha y_t + (1-\alpha)\, l_{t-1} b^{\emptyset}{}_{t-1}$$

$$b_t = \beta^* \left(\frac{l_t}{l_{t-1}}\right) + (1 - \beta^*)\, b^{\emptyset}{}_{t-1}$$

$$\hat{y}_{t+h|t} = l_t b^{\emptyset h}{}_{t-1}$$

EQUATION 4: FORECAST WITH MULTIPLICATIVE TREND

Visual inspection of the graph indicates a trend without a seasonal pattern. It should be observed that the y-axis do not start at zero. The first month shows a steady increase which is concurrent with the CPU usage, afterwards the trend stabilizes.

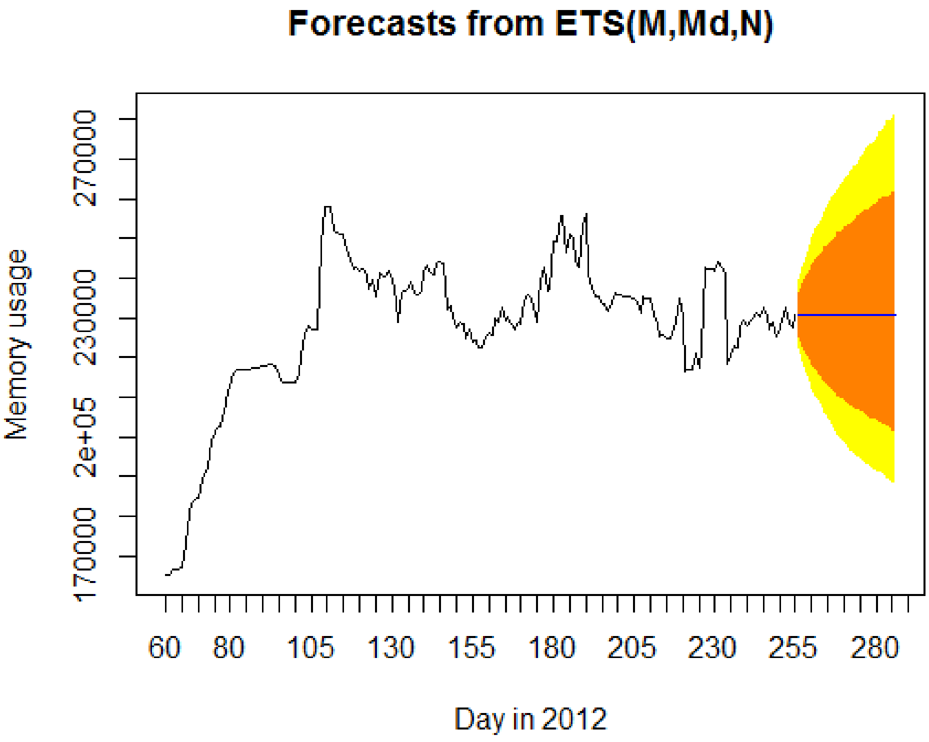


FIGURE 12: FORECAST ON MEMORY

FORECAST ON STORAGE

The graph depicted in Figure 13 depicts usage of storage on the vertical axis and the days within the year 2012 on the horizontal axis. In the automatic forecast, trend and seasonal effects are taken into account. Error handling is performed multiplicative which means that variations in predictions occur according to a statistical normal distribution and are proportional to the data. The equation for the calculation of the forecast is depicted in Equation 4.

Visual inspection of the graph indicates a positive trend without a seasonal pattern. The figure indicates much fluctuation in the storage usage. This results in wide minimum and maximum forecasted values. The fluctuations seem to concur with the fluctuations found in the CPU data, depicted in Figure 11. The trend seems to be a reasonable fit for a forecast. However the trend does not include the fluctuations which must also be assumed. This would imply that short-term forecast could be inaccurate when the real data shows an unexpected low or high. However, long-term forecast would not suffer so much, as the mean over a long period doesn't suffer as much from fluctuations.

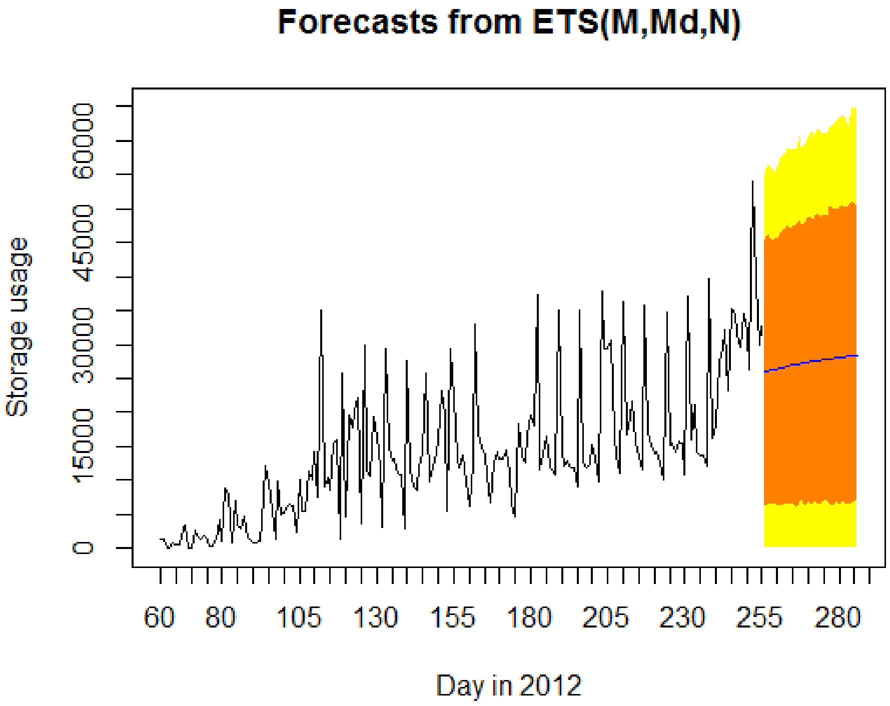


FIGURE 13: FORECAST ON STORAGE

Forecast on network

The graph depicted in Figure 14 depicts usage of network on the vertical axis and the days within the year 2012 on the horizontal axis. In the automatic forecast, trend and seasonal effects are taken into account. Error handling is performed multiplicative as well as trend, which means that variations in predictions occur according to a statistical normal distribution and are proportional to the data. The equation for the calculation of the forecast is depicted in Equation 4.

Visual inspection of the graph indicates a positive trend without a seasonal pattern. The figure indicates much fluctuation in the network usage. This results in wide minimum and maximum forecasted values. The fluctuations seem to concur with the fluctuations found in the CPU data, depicted in Figure 11. The trend seems to be a reasonable fit for a forecast. However the trend does not include the fluctuations which must also be assumed. This would imply that short-term forecast could be inaccurate when the real data shows an unexpected low or high. However, long-term forecast would not suffer so much, as the mean over a long period doesn't suffer as much from fluctuations. A disturbance in the fluctuations can be found spanning two weeks around the 170th day. This disturbance can also be observed in the storage graph, depicted in Figure 13.

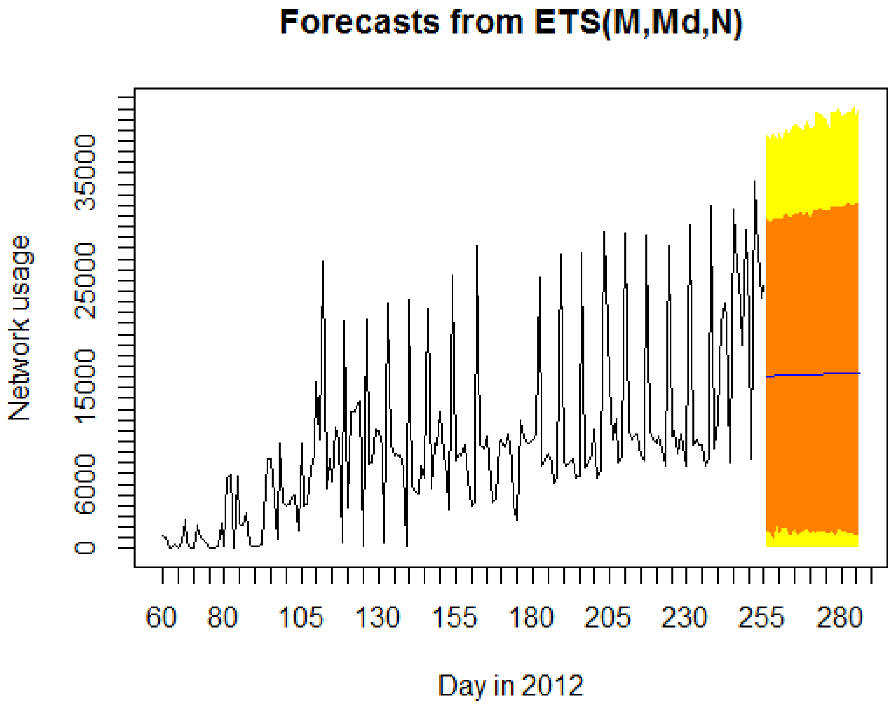


FIGURE 14: FORECAST ON NETWORK

Automatic selection of a forecast model

The comparison depicted Table 12, shows the error measurements of nine different forecast models compared to the automatic forecast. The first column shows the applied forecast model. The others show error values. The automatic forecast selected the forecasting model with has the lowest forecasting error. However different error measurements can provide contradictive information. The automatic forecast has the same values as the M.A.N. forecast model. This model is known as the Holt-Winter forecast model (Kalekar 2004). The error measurements for the M.A.N. model indicate the lowest mean error on al measurements except the mean absolute percentage error. However, this is related to the fact that RMSE is more sensitive because of the square operation.

Forecast model	ME	RMSE	MAE	MPE	MAPE	MASE
Automatic (R.J. Hyndman 2008)	-63.15 %	1228.37	881.58	-2.05 %	10.17 %	96.3 %
M.N.N.	82.94 %	1232.81	883.72	0.36 %	9.93 %	96.5 %
M.N.A.	82.94 %	1232.81	883.72	0.36 %	9.93 %	96.5 %
M.N.M.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %
M.A.N.	-63.15 %	1228.37	881.58	-2.05 %	10.17 %	96.3 %
M.A.A.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %
M.A.M.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %
M.M.N.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %
M.M.A.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %
M.M.M.	-65.74 %	1232.95	888.75	-2.20 %	10.31 %	97.1 %

TABLE 12: COMPARISON OF FORECAST ERROR RATES

Accuracy of forecasts

The forecast values and the comparison with the actual values are depicted in Table 13. The first column shows the forecast horizon. All forecast are performed using the automatic forecast selection, which resulted in the application of the M.A.N. forecast model (Holt-Winters forecast model) on all occasions. The forecast column indicates the actual expected amount of CPU usage. The second and third column show the minimum and maximum value which can be expected with 80% certainty. The fourth and fifth column show the minimum and maximum value which can be expected with 95% certainty. The sixth column shows the actual values measured by vSphere 5 on that particular day. The final column indicates the difference between the actual and the forecasted value.

The difference between the actual and forecasted value increases as the forecasting horizon becomes greater. This is compliant with current observations in literature.

Forecast horizon	Forecast	Low 80	High 80	Low 95	High 95	Actual	Δ in %
7 days	13119.35	8582.403	17656.31	6180.685	20058.02	13911	6.00 %
14 days	13794.53	7397.051	20192.01	4010.429	23578.63	10245	34.56 %
21 days	14469.70	6437.737	22501.67	2185.870	26753.54	12470	16.03 %
28 days	15144.88	5571.982	24717.78	504.3957	29785.36	11342	33.53 %
31 days	15434.24	5215.130	25653.35	0	31063.02	11788	30.93 %

TABLE 13: EVALUATION OF FORECASTS USING DIFFERENT FORECASTING HORIZONS (AUGUST UNTIL 10TH SEPTEMBER 2012)

For second comparison forecast are produced using a smaller dataset and a larger forecasting horizon. The forecast values and the comparison with the actual values are depicted in Table 14. The table is structured similarly as Table 13. The experiment uses a similar approach as described before, only this time, the dataset was limited to March until June. The predicted period was set at similar intervals. Furthermore, the intervals are extended to include 50 days, and 62 days. The difference between the actual and forecasted value increases as the forecasting horizon becomes greater. However, the delta values are much lower than those of the forecast described in Table 13. In the first month the forecasts seems steady. However towards the horizon the actual values suddenly drop.

Forecast horizon	Forecast	Low 80	High 80	Low 95	High 95	Actual	Δ in %
7 days	11695.15	7093.196	16297.09	4657.072	18733.22	12390	5.49 %
14 days	12336.79	5800.574	18873.02	2340.507	22333.08	10762	14.63 %
21 days	12886.78	4867.848	20905.71	622.882	25150.68	11454	12.05 %
28 days	13528.43	3858.910	23197.95	0	28316.68	12113	11.69 %
31 days	13803.42	3437.835	24169.01	0	29656.22	14812	- 7.30 %
50 days	15545.04	608.475	30664.94	0	38620.40	13579	14.48 %
62 days	16736.68	0	34678.74	0	44176.70	12470	34.21 %

TABLE 14: EVALUATION OF FORECASTS USING DIFFERENT FORECASTING HORIZONS (JULI & AUGUST 2012)

A graph that combines the forecast over the month's July and August with the actual values (depicted by the green line) is depicted in Figure 15.

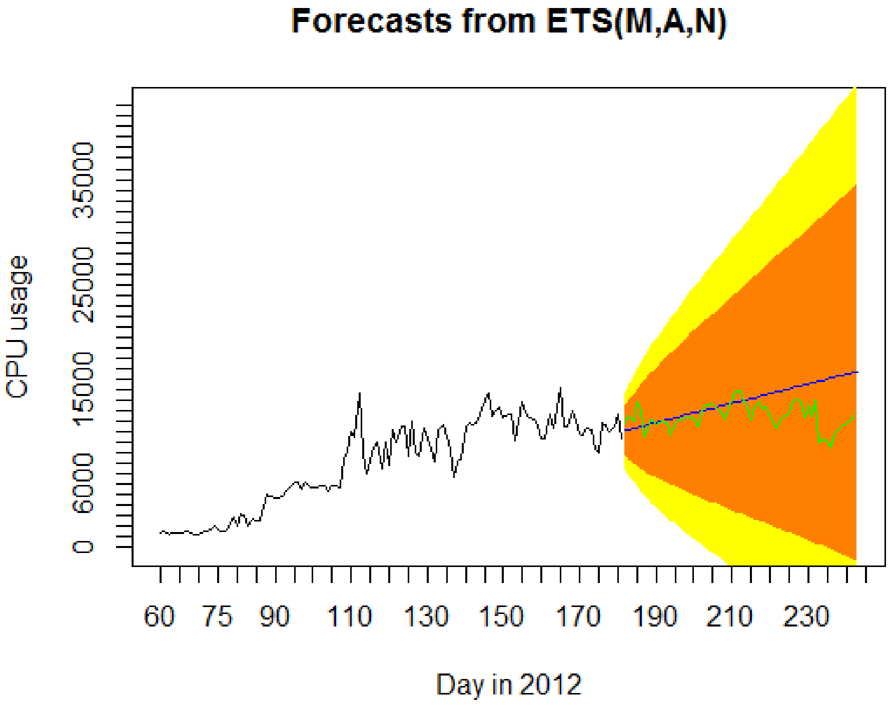


FIGURE 15: FORECAST ON CPU JULY AND AUGUST

The results as displayed in Table 14 are considered as a big improvement as the accuracy doubled. This enabled a test using a more distant forecasting horizon. The test results are depicted in Figure 15. The 50 day forecast has a relatively high accuracy with 14,48 difference between the forecasted and observed value. However, on the 62th day the accuracy decreased drastically. This is likely caused by the holiday season as observed earlier.

The differences between the actual and the forecast values described in Table 13 might be caused due to the holiday season effect. The training set of the forecast does not cover a whole year of data. As a result the forecast is based on a non-seasonal trend. In reality the actual usage of the infrastructure declines over the period of 8th of August until the 8th of September.

Hypothesis evaluation

$H1_0$: The accuracy of cloud costs forecasts decreases with increasing forecasting horizon but is adequate to predict two months ahead.

The acceptance of accuracy of forecasts is difficult to put into a single number. The mean absolute percent error of the forecast on CPU is around 10 %. This number is considered a good performance of a forecast. However, it is more a measurement of fit with the dataset. The forecasts depicted in Figure 15 indicate that the actual values are closely aligned with the forecasted values. The first month shows a similar trend in both values. However, at the end of the second month differences become larger. Still, the surface of the graph of two months determines the costs of the CPU usage. The graph indicates that the cloud costs are overestimated; this would mean there is some overprovisioning of infrastructure resources. This is very common for infrastructure. Furthermore, because provisioning of infrastructure resources is determines by offsets of capacity (per virtual machine), it is likely to round required capacity up. The minimum and maximum values at a confidence level of 95% on the forecasting horizon indicate that the forecast is progressively becoming uncertain towards the forecasting horizon. Especially the minimum and maximum for the predictions of the network and storage are double and halve the predicted value. However, memory is predicted with much more confidence. Based on the findings of this experiment the hypotheses cannot be rejected. However, based on the differences in success of forecasts related to the confidence of the forecasts further research should be performed to reach full acceptance of this hypotheses.

$H2_0$: The automatic forecast model selection provided by the … function of the R forecast package returns the most accurate forecasting model for cloud cost data.

The experiment indicated that the selection of a forecasting model is based on error measurements of error in accuracy. By creating ten separate forecasts and extracting the error measurements it was possible to compare these measurements. The automatic selection algorithm had already proved itself on the M3 challenge (R.J. Hyndman 2008). However the side note of this approach was that it performed well in short term (18 steps) forecasts. The experiment in this chapter expands this to 60 steps in this context. Based on the findings of the experiment this hypothesis is accepted.

$H3_0$: The accuracy of forecasts increases with the size of the datasets..

The experiment involved a one and two month forecast based on one dataset that have differing lengths. The differences between the actual and forecasted numbers described by Table 13 and Table 14 suggest that the latter one that is based on smaller dataset has a more accurate prediction compared to the former one. The forecast based on the smaller dataset is able to predict almost twice as far ahead with more accuracy compared to the larger dataset forecast. However, close inspection of the results indicate that this effect might be caused by the holiday season. This is not unlikely to happen in august. Both of the forecasts show a sudden drop in accuracy by the end of July. Because of the relatively small size of the dataset seasonality cannot be detected. The results of the experiment indicated that forecasts based on small datasets are very easily influenced by last month of the actual values. This can be explained by the exponential smoothening approach that weighs recent fluctuations more heavily over old ones. Based on the findings of the experiment the hypothesis is rejected for datasets shorter than two years (undetectable seasonal pattern).

INTERPRETATION OF RESULTS

Each of the four forecast used in this experiment produced insights into how usage will progress in the future. The blue trend lines indicate the value that is actually expected. The orange area indicates the possible outcome within 80 % confidence interval. The yellow area indicates possible outcomes within the 95 % confidence interval. The four metric which were forecasted on did not reveal a seasonal pattern. However, it does not indicate a lack of seasonality, but more likely indicates the dataset is too limited to detect seasonal patterns. Each graph shows a cyclical pattern which is likely to be observed in business applications. In order to detect a seasonal pattern, two years of data is required (Kalekar 2004). However, most cloud products are relatively young which makes it difficult to extract such a dataset.

The experiments conducted in this section prove that it is possible to conduct forecasts based on infrastructure usage. Usage forecasts can be inserted into a cost model in order to calculate expected costs. The technical solution constructed for these experiments reveals that by integrating a cost model and usage statistics it is possible to construct a technical solution that forecasts cloud cost. By doing so the potential for forecasting is demonstrated.

5.5 AN EXPERT REVIEW ON CLOUD COST FORECASTING

An expert review is conducted with five experts from the field. The review set-up is described in chapter one. The review is conducted as a structured interview with four experts. The fifth expert reviewed the rigor of the forecasting approach. The guidelines for the interview can be found in Appendix H: Expert review protocol. Experts with different background are selected. The first expert is an implementation project manager who has performed numerous projects in the public sector as well as the financial sector. The second review is conducted with a senior infrastructure consultant. The third review is conducted with an infrastructure specialist. The fourth review is conducted with an infrastructure sizing and metering expert. The final review is conducted with a business intelligence consultant. An overview of the experts, their affiliation and review focus is depicted in Table 15.

Expert	Background	Affiliation	Review focus
1	Project manager	Centric	Practical application of cloud cost forecasting
2	Senior infrastructure consultant	Centric	The output, and practical application of cloud cost forecasting
3	Infrastructure specialist	Centric	Interpretation of input, output of forecasts
4	Infrastructure sizing and metering specialist	Centric	Interpretation of input, output of forecasts
5	Business intelligence consultant	CSB-systems	Proper application of forecasting techniques

TABLE 15: EXPERT REVIEW SELECTION

REVIEW ON THE PRACTICAL APPLICATION OF THE FORECASTING SOLUTION

Overall the experts are confident that the proposed forecast technique has obtained new insights into how costs can be forecasted. However expert one and two agree that it when customers are given a choice, they are willing to pay more for certainty using fixed pricings. Still, forecasting combined with the proper business model would enable pay per use pricing combined with a forecasting technique. Expert one suggested that the forces that cause infrastructure load are determined by how the does business which, can also change over time. Therefore it is difficult to rely on the past to properly predict the future. Expert four suggested 'a situation where customers decide their level of pay-per-use as well as fixed'. Expert two suggested that short-term forecasts up to three months are interesting for sizing purposes. Expert one to four agree that the creation of automated cost estimations for customers are interesting, however this should be well tested. Expert one noted that 'quantitative techniques are considered as high potential techniques, but they require a degree of formalism in the cloud operator's organization'.

REVIEW ON THE INPUT OF THE FORECASTING SOLUTIONS

Expert three and four concurred that VMware statistics are promising for such applications. There are some small projects where similar information is already used for metering and billing. Expert four indicated that 'it would be very difficult to deduce the amount of required virtual machines from CPU and memory usage'. However, 'expressed as a percent change relative to the past it is likely to work'. The CPU forecast is considered as the most important forecast for determining costs by expert two, three, and four. Expert four also noted that 'the forecast on storage usage is not very effective'. The reason for this is that a virtual machine will always perform operations on about the same size of storage, but from a different month. Moreover, storage can grow, but virtual machines will essentially process the same amount of data every month. Expert four suggested that 'a forecast on IOPS (input output operations per second) would be more useful'. However, the current business model does not include IOPS for billing.

REVIEW ON THE NUMERIC OUTPUT OF THE FORECASTING SOLUTION

The experts agree that a forecast should be traceable in some way to an amount of the virtual machines. The virtual machine is a packaged good. It is what the customer is ultimately paying for, regardless this is on an hourly, daily or monthly basis. The fifth expert reviewed the technical application of the forecasting techniques. The expert concluded that 'forecasting techniques where properly applied'. However, the fifth expert noted that 'the time series are too small (or short) to be able to detect seasonal patterns'.

5.6 DISCUSSION OF RESULTS

This chapter introduces a technical solution which results in the forecast of cloud costs. The technical solution includes usage data, a cost model and a forecasting technique. Modern infrastructure virtualization techniques record usage for the purpose of system administration, monitoring, and – more recently – billing. Because of their market leader position in virtualization, VMware has been selected to serve as a source of data for the controlled experiment. As discussed in this chapter the default setting of the VMware management software incorporates sufficient statistics for the forecasting technique discussed in this section. However, it should be noted that the consolidation process which compresses statistics older than one moth to keep the database small, does cause a loss of data which otherwise would enable a forecast of traffic over a 24 hour time window. The final part of the technical solution is the application of the forecasting technique itself. For this research the mathematical analyses software R was chosen. R has a track record in a broad diversity of scientific domains, and also important is included in commercial products as well (Foundation 2012). R includes a forecasting package which support a variety of forecasting models using linear extrapolation (R. Hyndman, Razbash, and D. Schmidt 2012; R.J. Hyndman 2008; Makridakis, Wheelwright, and R.J. Hyndman 1998). The forecasting package includes an automatic forecasting model selection technique which has proven itself in a grand international comparison, but evenly important, similar results are found in experiment described in this chapter (Rob J. Hyndman et al. 2002). As the automatic forecasting model selection technique is known to work particularly well on relative short forecasting horizons (Rob J. Hyndman et al. 2002).

Real data was used from a cloud production environment. Forecasting calculations are made with varying forecasting horizons. However, the dataset used in this experiment is too small to create a long term forecast (i.e. a forecast over a year). It can be expected that when cloud technology matures, forecasts become more accurate, and forecasting horizons become more distant. The experiment has showed us that forecasting should be viewed as an ongoing effort to capture practice in a formal manner, which requires continuous improvement.

The expert review indicated that the creation of the forecasting solution led to new insights into cloud costs. This reflection ultimately is the goal of design science research which proved to be an appropriate research method for this research. The ability of the technical solution to create a forecast of one or two months can be a useful benchmark for existing judgment based forecast. All experts agree that the suggested approach has potential but is not ready to be incorporated in a business model.

Forecasts on usage can also be used in other manners. For the cloud operator, forecasting information can be used to in an automated cloud environment to create a feedback loop directing the virtualization software to re-size the infrastructure for the forecasted figures. Forecast figures can also be used to plan the acquisition of additional infrastructure capacity. Furthermore these figures can be used by the sales department for constructing propositions. For customers, the information can also be used for management purposes. Some companies also provide periodical reporting about the level of service rendered, for example uptime and updates. Forecasts would be a good candidate to be included in such reports.

Chapter 6. Discussion & conclusion

This section describes a discussion about the research finding and their limitations, the conclusion, and finally an indication of future research directions is provided.

6.1 Discussion and research limitations

Reflecting on the problem statement, this thesis is an effort to address the uncertainty of costs (Ibrahim, He, and Jin 2011). This effort has great potential for organizations transitioning from a traditional IT environment founded through investments, towards a pay-per-use IT environment. The discussion section is organized as follows. First, the research method is discussed, after this the discussion is organized in three parts, each addressing a sub question and of this thesis.

The research method applied in the research is called design science (Hevner et al. 2004). It uses an IT artifact as a source of knowledge discovery. The IT artifact is both subject and the goal of the research. It is important that observations made during the construction of the IT artifact constitute the main scientific contributions. The IT artifact in this research is a technical solution for cloud cost forecasting. The main research question is divided into three sub questions; their respective results are covert in chapter three to five. Conclusions about the findings are presented in this chapter.

Cloud costs

In order to forecast cloud costs, first the question was posed on what constitutes these costs. This question was answered using a systematic literature review (SLR) (Kitchenham 2007). The reason for choosing such a time consuming approach was that cloud technology is a relatively new research field. Therefore, the concepts are likely to be not so well defined, and little literature can be considered *established*. Supporting this argument is the lack of journal publications included in the systematic literature review. The guidelines on performing structured literature research described in the work of Kitchenham (Kitchenham 2007) are closely followed. One of the limitations of the literature study is that during the exploration phase, some literature databases differed substantially in regard to their search features. It caused a title-only search neglecting the abstracts of these articles. This might have caused a disqualification of important articles, which potentially impacts the conclusion of the first research question. In order to minimize the risk an approach similar to snowballing was used to discover *high potential* literature. It is also likely that high quality literature would be referenced by the literature included in the structured literature review. However, this did not lead to additional material. The result of the SLR provides short lists of cloud cost variables and describes the context about how these costs occur. However, the review indicates that most literature lack a detailed explanation about these cost variables. The lack of detailed explanation of costs variables was determined to be unsatisfactory, especially considering that the majority of the literature about cloud technology does mention costs as one of the motivations to adopt it. Another interesting observation is that recent literature seems to impeach cost advantages of cloud products but instead focuses on high quality products and value added services.

Cloud technology causes experts from infrastructure and software to become more co-dependent on each other, as both play their roll in cost estimation. Separation of business units, functions, and departments is a struggle for something as holistic as research on cloud costs. It is difficult to find an expert who both understands the infrastructure and the software that runs on them. In order to perform a proper expert review, it requires the selection of different perspectives on cloud technology.

COST FORECASTING TECHNIQUES

In order to predict costs, one would need to know costs, and also have a technique that is able to process data about these costs and provide an accurate forecast based on data. There already have been applications of forecasting techniques in load balancing software, as indicated by the *structured literature review* (SLR) from the previous section. The field of forecasting can be considered as an active research field for which most of the fundamental theoretical research is performed between 1960 and 1980. To this day, it is an active research field, where new applications of forecasting are still being researched. Because of the relatively established nature of the field the research approach that was chosen for this section was a literature study. The literature study can be considered as less formal compared to a SLR. However, relevant literature on the intersection of cloud costs and forecasting would still have been found in the SLR conducted for the first sub question.

Early on in the research, quantitative methods for forecasting were considered as a superior candidate over judgment based forecasts, which require human intervention. The reason for this is that cloud technology thrives on economies of scale. It means that in order to create a forecasting solution which involved the structured intervention of people; it would negatively impact these economies of scale. However, it should be mentioned that judgment based forecasts are not disqualified because of reasons other than scale. Especially forecasts that combine both human judgment with statistical analysis have proven to be successful (Gigerenzer 1990).

A diversity of forecasting techniques can be applied in the prediction of cloud costs. While several applications are discussed, most suffer from the constraints of practice and remain theoretical. For this research only the extrapolation models are validated in practice. The primary reason for this is that most quantitative techniques imply that there is an abundance of sound historical data readily available. In practice this is not always the case. However, the other techniques do yield substantial potency to base further research on. Moreover, these models should be taken into consideration while designing new cloud products or when revising existing ones.

As a final consideration of the sub question, it should be noted that forecasting is a discipline which is based on heuristics and significant relationships, which in practice is not always the case. Therefore forecasting should be seen as a practice of continuous improvement.

CLOUD COST FORECASTING TECHNIQUES IN PRACTICE

Chapter five combined insights of chapter three and four and integrated them into a technical solution which forecasts cloud costs. An experiment is used in order to quantitatively research the validity of the proposed forecasting solution.

The experiments that are executed in order to validate the technical solution are performed on a single dataset. The dataset is based on a requirement that there needs to be a minimum of three virtual machines in order to make forecasting work. In practice this assumption would not disqualify an average cloud customer. However, whether or not there is a maximum of environments to base forecasts on was not part of the research. The dataset used in the experiment describes measurements on a daily level. However in reality infrastructure load fluctuates during the day. While it is true that some forecasts provide a proper indication for sizing purposes, one must assume that infrastructure load fluctuates over time. For example, most office automation systems show a peak in infrastructure load during office hours, and are nearly idle at night. The daily interval of statistics is the result of a consolidation process that keeps the statistics database to an acceptable storage size. At this moment, it would be quite costly to store statistics on an hourly timescale. The lack of granularity can be considered as a weakness. A non-invasive solution for it would be, to store the pattern of distribution along with the actual values. This distribution would help to recover how the infrastructure load was distributed during the day while still respecting the requirement to consolidate this data to small data points.

The implemented forecasting solution is chosen based on positive results in the M3 challenge, which serves a comparison in performance of forecasting algorithms (Rob J. Hyndman et al. 2002). It proved to work particularly well on relatively short forecasting horizons. The experiment performed on real data, indicated a forecasting horizon of one month provided a prediction within the 95 % confidence level. However, it is important to verify the findings in different situations. This should include trying forecasting on more or less servers, and a completely different type of software (i.e. image processing software).

The proposed forecasting solution involves a technique that reads the statistics of the virtualization layer. The costs forecast could potentially be used for planning and sizing purposes. Although it sounds promising, it should be handled with great care. Modern applications contain a great diversity of optimization solutions on different architecture levels. Even so, expansion of capacity already indicate decreasing returns to scale (Napper and Bientinesi 2009). The virtualization software itself, the operating system, application server, and software itself will contain intelligent solutions for handling requests in an efficient manner (Caron, Frédéric Desprez, and Muresan 2011). Different optimizations can also create a resonance in the VMware statistics which disrupts forecasts. For example, resonance could be mistaken for seasonality. When implementing cloud cost forecasting, it is important to interpret the forecasts as a means of control.

The expert review performed as a means of validation of the technical solution provided context to increase understanding of the research results. Five experts are selected with different backgrounds. Four are employees of Centric. Three come in contact with cloud technology on a daily basis. In addition to this a business intelligence expert reviewed the application of forecasting techniques. One could argue that an expert from an organization other than Centric would be beneficial to the external validity of their findings. However, two experts have experience in consulting projects that qualifies a broad understanding of the cloud industry.

6.2 CONCLUSION

This section is aimed at disclosing the research results as well as discussing the findings based on the IT artifact belonging to the design science project. The purpose of design science is a two-step process that first enables the researcher to construct a design process to produce 'something new', an artifact. The second, however equally important, design process involves the evaluation of that artifact to provide feedback and generate new knowledge about the problem at hand. The conclusions are presented for each sub question which together answers the main research question.

For the purpose of this research a technical solution is created that consists of a cloud cost model, historical cloud usage information, and a forecasting technique. The result of the technical solutions is a learning process which resulted in this master's thesis report. Furthermore, a proof of concept automatable technical solution is created that is able to predict the costs of a cloud solution which increases certainty to a customer about what future cloud costs will be.

CLOUD COSTS

The research question that is answered in chapter three is "what variables determine cloud costs and how do they relate to each other?" A literature study is performed to answer the question. Out of an original 498 literature items about the concept of 'cloud cost' 55 papers are selected as useful. Through the systematic gathering of cost variables, a taxonomy of costs is constructed. In total 83 variables where identified from the literature set. An important observation is the fact that the vast majority of the reviewed literature only covers a small portion of the cost variables of the taxonomy. Based on a threshold set, so each variable would at least be described in five articles, there are only eight independent cloud cost variables. However, these eight variables are quite generic, which means that much of the 83 original variables can be wrapped up in these. Eight variables where identified by using a systematic literature review which are described in detail in more than five articles. The cost variables are: cost per virtual machine, network, storage memory, energy, CPU, service level agreements (SLAs) and objectives (SLOs).

Cloud computing should still be considered as an emerging technology (Pettey and Meulen 2012). The lack of maturity in the area creates great complexity in cost calculations. For example, software license costs are still difficult to allocate to the proper customers. In practice a lawyer, accountant, and technical staff is needed to create a cost driver. Because software licenses are expensive, they results in legacy technical solutions and customized contracts. Most of the customized work requires a separated cloud environment. This is troublesome as it directly affect economies of scale in a negative manner. Cloud technology is built on top of these economies of scale. In order to acquire and sustain economies of scale infrastructure operators must standardize their services. It is difficult for infrastructure operators which used to do allot of custom contracting. Most of the customized environments have a large overhead caused by manual intervention. Even for organizations that do not pursue a low-cost strategy it should be done using self-service portals or should be avoided at all.

Finally, in order to let costs of a cloud environment have any relevance at all, organizations should aspire to use an accounting strategy at which price and costs are related, using a calculable profit margin. This is a precondition for enabling the transfer of a cost calculation to a quotable price. Where cost forecasting is a nice thing to have, price forecasting has great potential for cloud operators' customers.

COST FORECASTING TECHNIQUES

This section discusses literature about forecasting techniques, covering literature of the last six decades. An assessment of the applicability for each forecasting technique, including its potential for the prediction of cloud costs is provided. Forecasting examples are provided for two customer groups, being both new customers and existing ones. For some techniques different types of application are provided.

Accuracy, relevance and acceptance of cloud cost forecast are quality characteristics of a prediction. In order to increase accuracy the amount of data recorded about the usage of the service and infrastructure should be as specific as possible. Furthermore, using segmentation, it is possible to distinguish between different customer groups or applications in order to achieve a better statistical fit with historical data.

Accuracy of a prediction can be described using standard metrics, which are *mean error*, *root mean squared error, mean absolute error, mean percentage error, mean absolute percentage error, and mean absolute scaled error.* All three metrics describe the degree to which the forecasted value deviates from the actual value, which are sometimes also called real or perceived value.

Finally, the way in which applications stress infrastructure is identified as an important influencing factor in regard to the choice between suitable forecasting techniques. Also, the homogeneity of customer characteristics determines whether or not statistics of other customers have predictive power on each other. Using these two variables, four application stereotypes of cloud cost forecasting are identified. These stereotypes support the decision-making process, in the selection of a forecasting techniques and their respective application. A two-by-two decision matrix is constructed to aid forecast technique selection.

CLOUD COST FORECASTING TECHNIQUES IN PRACTICE

This section discusses how an automatable technical solution for cloud cost location could be used in practice. The results from the controlled experiment and the expert review indicate that cloud cost forecasting is indeed possible. The research question posed in chapter five is answered using the creation of a technical solution.

The controlled experiment was designed to test three hypotheses. The objective of the experiment is to test existing forecasting techniques. The forecasts from the experiment show an accurate forecast up to two months based on a dataset spanning five months. The automatic forecast model selection algorithm provides the most accurate model. The forecast handles trend very well, however de dataset is too limited to cope with seasonality.

The costs of cloud service are predictable using a bottom up approach that uses metering statistics at the infrastructure level. As cloud is based on a pay per use kind of service model (Rosenthal, Mork, and MH Li 2010), customers are billed based on usage. As demonstrated in a chapter 4, based on literature, most of the costs incurred in a cloud environment can be derived from infrastructure. For example license fees for software can be derived, based on the amount of CPUs used. When there is enough data available about the usage of the infrastructure, the data can be used to create forecast about future usage. To accomplish this, a cost model needs to be created that is able to roll up costs from infrastructure the infrastructure layer up to the other layers of the cloud stack (the platform layer and the software layer). One of the characteristics of such a model is that it does not define the owner of the platform. It could be that the infrastructure is 'leased' from a 3rd party and is offered as a *platform as a service*. It could also be that the model wraps up all costs that incur within a company from infrastructure to platform to software costs for a '*software as a service'* product. The cost model could even be used to calculate facility level costs (i.e. depreciation of a building).

In the technical solutions historical statistics are used to extract past usage. Market leader VMware has an infrastructure management suite called vSphere which performs measurement for monitoring and also billing purposes. After careful consideration these statistics where found useful to create a forecast on. The statistics database includes measurements on utilization of resources consolidated to a daily average.

The technical solution uses extrapolation models to forecast future infrastructure usage. It applies exponential smoothening so that recent data points weight more heavily than older ones. The extrapolation model measure trend and seasonal effects of the dataset and determines the proper extrapolation model out of eighteen different combinations (R.J. Hyndman 2008). Two experiments where performed to validate the technical solution. Real data was used from a cloud production environment. The first experiment is aimed to experiment on different forecasting horizons. Forecasts up to two months can be made with acceptable accuracy. This was validated using an expert review. However, the dataset used in this experiment is too small to create a long term forecast (i.e. a forecast over a year). The second experiment is aimed at validating the automatic selection of the forecast model. The automatic selection proved to select the proper extrapolation model which has the lowest error values. This finding is concurrent with findings from the large-scale M3 challenge (Rob J Hyndman 2006).

It can be expected that when cloud technology matures, forecasts become more accurate, and forecasting horizons become more distant. Also the availability of historical data over a long period of time is vital in performing an accurate forecast. This is even more important when forecast should include seasonal effects. The experiment has showed us that forecasting should be viewed as an ongoing effort to capture practice in a formal manner, which requires continuous improvement.

6.3 FUTURE RESEARCH DIRECTIONS

This section is aimed at providing some future research directions.

Following up on (Keogh and Kasetty 2003), it is vital that the proposed forecasting technique is tested on multiple datasets. Future research should concentrate on improving the external validity of this research. The argument provided by Keogh & Cassity (Keogh and Kasetty 2003), is the lack of testing of data mining techniques in practice.

However, I feel this research could benefit from additional experiments as well. Indications of target cloud environments are a large cloud provider which offers homogeneous software as a service product on a large scale, like Dropbox. In contrast, results should be replicated with different types of software and hardware. Office automation and business information systems cause a different load footprint compared to massive multiplayer online game servers. In addition, the proposed forecasting solution is based on a VMware virtualization software, which could behave differently compared to competitor Oracle VM VirtualBox (Oracle 2011).

The technical solution proposed in this research makes extensive use of the monitoring capabilities of VMware. However, it still only scratched the surface of the potential in regards to forecasting purposes. One of the hurdles that encountered during this research is the consolidation engine of VMware, which condenses statistics logs to a format that is cost effective to store. Most statistics older than one month are therefore consolidating to a daily average or sum. However, this means that the load distribution over the day is lost. As infrastructure loads follow a pattern with peaks and plains this tells a different story than a computed average or sum. Future research should concentrate on a non-intrusive manner to store this distribution information. A graceful solution would be one that causes little or no adjustments to be made to the VMware software, while storing the proper information.

This thesis introduces the concept of elasticity in load which expresses the sensitivity to change of the load footprint measured on the infrastructure and the usage of the application. This relates to forecasting because this elasticity will determine how much extra infrastructure is needed when usage of the applications growths. It can potentially also be used in a different type of forecasting than the one described in this thesis. When measurements of usage on the *front-end* of an application can be performed, as well as on the infrastructure level this data can serve as input for forecasting based on quantitative analogies. The elasticity also provides a metric about how susceptible a piece of software is to change in usage. This could also be used as a non-functional quality characteristic for software.

The technical solution proposed in this research is able to predict future cloud costs. This technique can also be used for other purposes then cost forecasting. One of these purposes is infrastructure re-*sizing*. When these forecasting techniques are able to accurately predict future usage it would also be possible to re-size the infrastructure based on these forecasting by creating a feedback loop. Modern virtualization systems already have an interface for such a feedback loop. Objectives of research would be to search the most effective tactic for this, as well as how to respect service level objectives.

Platform operators as well as their clients have expressed interest for this technology. Future research might be able to focus on the customers of a cloud service, as for this research the primary focus was on the operator of a cloud platform. Customers would likely benefit from the forecast itself, which could be integrated in a reporting environment. Customers could use a self-service portal to view forecasting reports and potentially react on it by committing in advance to cloud resource.

References

Adya, Monica, Fred Collopy, J.Scott Armstrong, and Miles Kennedy. 2001. “Automatic identification of time series features for rule-based forecasting.” *International Journal of Forecasting* 17(2): 143–157. http://linkinghub.elsevier.com/retrieve/pii/S0169207001000796.

Ardagna, Danilo, Sara Casolari, Michele Colajanni, and Barbara Panicucci. 2012. “Dual time-scale distributed capacity allocation and load redirect algorithms for cloud systems.” *Journal of Parallel and Distributed Computing*. http://linkinghub.elsevier.com/retrieve/pii/S0743731512000585 (March 19, 2012).

Armbrust, M, A Fox, R Griffith, and AD Joseph. 2010. “Above the Clouds: A Berkely view of cloud computing.” *Communications of the ACM*. http://dl.acm.org/citation.cfm?id=1721672 (June 22, 2012).

Armstrong, J Scott. 2001. “Combining Forecasts.” In *Principles of Forecasting*, Norwell: Kluwer Academic Publishers, p. 417–439.

Armstrong, J Scott, and Kesten C Green. 2005. “Demand Forecasting : Evidence-based Methods.” In *Strategic Marketing Management: A Business Process Approach*,.

Asoke K. Talukder, Lawrence Zimmerman and Prahalad H. A. 2010. “Cloud Economics: Principles, Costs, and Benefits.” *Cloud Computing*. http://www.springerlink.com/index/H814JK871122253R.pdf (March 28, 2012).

Assunção, Marcos Dias, Alexandre Costanzo, and Rajkumar Buyya. 2010. “A cost-benefit analysis of using cloud computing to extend the capacity of clusters.” *Cluster Computing* 13(3): 335–347. http://www.springerlink.com/index/10.1007/s10586-010-0131-x (March 8, 2012).

Assunção, Marcos Dias De, Alexandre Costanzo, and Rajkumar Buyya. 2009. “Evaluating the Cost-Benefit of Using Cloud Computing to Extend the Capacity of Clusters.” *Proceedings of the 18th ACM international symposium on High performance distributed computing*: 141–150.

Bai, L., T. Li, X. Wu, and Z. Xie. 2011. “Charging Model Research of Infrastructure Layer in Cloud Computing based on Cost-profit Petri net.” *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2011 International Conference on*: 435–441.

Baskerville, Richard, and J Pries-Heje. 2009. “Soft design science methodology.” In *Proceedings of the 4th international conference on design science research in information systems and technology*, http://dl.acm.org/citation.cfm?id=1555631 (June 22, 2012).

Bossche, R Van den. 2010. “Cost-optimal scheduling in hybrid iaas clouds for deadline constrained workloads.” *2010 IEEE 3rd International Conference on Cloud Computing*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5557990 (March 28, 2012).

Boutaba, Raouf, Lu Cheng, and Qi Zhang. 2011. “On Cloud computational models and the heterogeneity challenge.” *Journal of Internet Services and Applications*. http://www.springerlink.com/index/10.1007/s13174-011-0054-7 (April 4, 2012).

Breskovic, Ivan, Michael Maurer, Vincent C Emeakaroha, Ivona Brandic, and Schahram Dustdar. 2011. “Cost-Efficient Utilization of Public SLA Templates in Autonomic Cloud Markets.” *Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on*: 229–236.

Brillinger, D.R. 2001. “Time series: data analysis and theory.” *Society for Industrial Mathematics* 36.

Bryant, A. 2002. “Grounding systems research: Re-establishing grounded theory.” *System Sciences, 2002. HICSS. Proceedings of the* 4(1): 52–42. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=994383 (June 22, 2012).

Byun, Eun-Kyu, Yang-Suk Kee, Jin-Soo Kim, and Seungryoul Maeng. 2011. “Cost optimized provisioning of elastic resources for application workflows.” *Future Generation Computer Systems* 27(8): 1011–1026. http://linkinghub.elsevier.com/retrieve/pii/S0167739X11000744 (March 12, 2012).

Caron, Eddy, Frederic Desprez, and Adrian Muresan. 2010. “Forecasting for Grid and Cloud Computing On-Demand Resources Based on Pattern Matching.” *2010 IEEE Second International Conference on Cloud Computing Technology and Science*: 456–463. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5708485 (March 9, 2012).

Caron, Eddy, Frédéric Desprez, and Adrian Muresan. 2011. “Pattern Matching Based Forecast of Non-periodic Repetitive Behavior for Cloud Clients.” *Journal of Grid Computing* 9(1): 49–64. http://www.springerlink.com/index/10.1007/s10723-010-9178-4 (March 26, 2012).

Chaisiri, S. 2009. “Optimal virtual machine placement across multiple cloud providers.” *… Computing Conference, 2009. …*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5394134 (March 28, 2012).

Chaisiri, Sivadon, Bu-sung Lee, and Dusit Niyato. 2011. “Optimization of Resource Provisioning Cost in Cloud Computing.” *Services Computing, IEEE Transactions on*: 1–32.

Chard, Kyle, Michael Russell, Yves A Lussier, and Jonathan C Silverstein. 2011. “Scalability and Cost of a Cloud-based Approach to Medical NLP.” *Computer-Based Medical Systems (CBMS), 2011 24th International Symposium on*: 1–6.

Chen, Yao, and Radu Sion. 2011. “To Cloud Or Not To Cloud ? Musings On Costs and Viability.” *The ACM Symposium on Cloud Computing SOCC 2011*.

Corporation, Alinean. 2012. “TCO tools.” http://www.alinean.com/value_selling/tco_tools (May 10, 2012).

Csorba, M., H. Meling, and P Heegaard. 2011. “A Bio-inspired Method for Distributed Deployment of Services.” *New Generation Computing* 29: 185–222.

Deelman, Ewa, Gurmeet Singh, Miron Livny, Bruce Berriman, John Good, U S C Information, Marina Rey, and Wisconsin Madison. 2008. “The Cost of Doing Science on the Cloud: The Montage Example.” *Proceedings of the 2008 ACM/IEEE conference on Supercomputing (SC’08)*: 1–12.

Dougherty, Brian, Jules White, and Douglas C. Schmidt. 2012. “Model-driven auto-scaling of green cloud computing infrastructure.” *Future Generation Computer Systems* 28(2): 371–378. http://linkinghub.elsevier.com/retrieve/pii/S0167739X11000902 (March 27, 2012).

Fayyad, U.M., G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. 1996. *Advances in knowledge discovery and data mining*.

Ferrer, Ana Juan, Francisco Hernández, Johan Tordsson, Erik Elmroth, Ahmed Ali-Eldin, Csilla Zsigri, Raül Sirvent, Jordi Guitart, Rosa M. Badia, Karim Djemame, Wolfgang Ziegler, Theo Dimitrakos, Srijith K. Nair, George Kousiouris, Kleopatra Konstanteli, Theodora Varvarigou, Benoit Hudzia, Alexander Kipp, Stefan Wesner, Marcelo Corrales, Nikolaus Forgó, Tabassum Sharif, and Craig Sheridan. 2012. “OPTIMIS: A holistic approach to cloud service provisioning.” *Future Generation Computer Systems* 28(1): 66–77. http://linkinghub.elsevier.com/retrieve/pii/S0167739X1100104X (March 16, 2012).

Foundation, R. 2012. “R foundation members and supporters.” http://www.r-project.org/foundation/memberlist.html (October 9, 2012).

Garfinkel, Simson L. 2007. Applied Sciences *Technical Report TR-08-07 : An Evaluation of Amazon ’ s Grid Computing Services : EC2 , S3 and SQS*.

Garg, Saurabh Kumar, Chee Shin Yeo, Arun Anandasivam, and Rajkumar Buyya. 2011. “Environment-conscious scheduling of HPC applications on distributed Cloud-oriented data centers.” *Journal of Parallel and Distributed Computing* 71(6): 732–749. http://linkinghub.elsevier.com/retrieve/pii/S0743731510000936 (March 17, 2012).

Gentry, Lance, R. Calantone, and S Cui. 2006. “The forecasting classification grid: A typology for method selection.” *Journal of Global Business Management* 2: 48–60.

Ghanbari, Hamoun, Bradley Simmons, Marin Litoiu, and Gabriel Iszlai. 2012. “Feedback-based optimization of a private cloud.” *Future Generation Computer Systems* 28(1): 104–111. http://linkinghub.elsevier.com/retrieve/pii/S0167739X11001014 (March 12, 2012).

Gigerenzer, Gerd. 1990. “HOW TO INTEGRATE MANAGEMENT JUDGMENT WITH STATISTICAL.”

Glaser, B.G., and A.L. Strauss. 1967. *The discovery of grounded theory: Strategies for qualitative research*. Chicago: Aldine de Gruyter.

Gmach, Daniel, and S Krompass. 2005. “Dynamic load balancing of virtualized database services using hints and load forecasting.” *Conference on Advanced Information Systems Engineering*. http://webkemper1.informatik.tu-muenchen.de/research/publications/conferences/ASMEA05_AutoGlobe.pdf (June 14, 2012).

Gmach, Daniel, Jerry Rolia, and L. Cherkasova. 2010. “Resource and Virtualization Costs up in the Cloud : Models and Design Choices.” *Dependable Systems \& Networks (DSN), 2011 IEEE/IFIP 41st International Conference on*: 395–402.

Goldkuhl, G. 2004. “Design theories in information systems-a need for multi-grounding.” *Journal of Information Technology* 6(2): 59–72.

Greenberg, Albert, James Hamilton, David A. Maltz, and Parveen Patel. 2008. “The cost of a cloud: research problems in data center networks.” *ACM SIGCOMM Computer Communication Review*. http://dl.acm.org/citation.cfm?id=1496091.1496103 (March 28, 2012).

Gregory, Robert, and Goethe Wayne. 2010. “Design science research and the grounded theory method: characteristics, differences, and complementary uses.” In *European Conference on Information Systems*,.

Haak, Steffen, and Michael Menzel. 2011. “Autonomic Benchmarking for Cloud Infrastructures An economic optimization model.” *Proceedings of the 1st ACM/IEEE workshop on Autonomic computing in economics*: 27–32.

Hamilton, J. 2010. “Cloud computing economies of scale.” *MIX’10*. http://perspectives.mvdirona.com/2009/03/29/CloudComputingEconomiesOfScale.aspx (March 28, 2012).

Hevner, Alan R. 2007. “A Three Cycle View of Design Science Research.” *Scandinavian Journal of Information Systems* 19(2): 87–92.

Hevner, Alan R, and Salvatore T March. 2003. “The information systems research cycle.” *Computer* 36(11): 111–113.

Hevner, Alan R, Salvatore T March, Jinsoo Park, and Sudha Ram. 2004. “Design Science in Information Systems Research.” *Management Information Systems* 28(1): 75–105.

Hong-Linh Truonga,∗, Schahram Dustdar. 2010. “Composable cost estimation and monitoring for computational applications in cloud computing environments.” *Procedia Computer Science*. http://www.sciencedirect.com/science/article/pii/S1877050910002449 (March 28, 2012).

Hsiao, Jen-Hao, and Yu-Zheng Wang. 2007. Proceedings of the 2007 conference on Digital libraries - JCDL ’07 *Virtual Center database sizing guidelines (MSSQL Server)*. New York, New York, USA: ACM Press.

Hyndman, R.J. 2008. *Forecasting with exponential smoothing: the state space approach*. Springer Verlag.

Hyndman, Rob J. 2006. “Another look at forecast accuracy.” *Foresight* (4): 43–46.

Hyndman, Rob J., Anne B. Koehler, Ralph D. Snyder, and Simone Grose. 2002. “A state space framework for automatic forecasting using exponential smoothing methods.” *International Journal of Forecasting* 18: 439–454.

Hyndman, Rob, Slava Razbash, and Drew Schmidt. 2012. *Forecasting functions for time series and linear models*. http://cran.r-project.org/package=forecast.

IBM corporation. 2012. “Estimate and compare your cost.”

IDC. 2009. “IDC Says Cloud Computing Is More Than Just Hype; Worldwide IT Spending On Cloud Services Expected To Reach US$42 Billion By 2012.” *IDC Press*. http://www.idc.com/AP/pressrelease.jsp?containerId=prSG21724009 (March 9, 2012).

Ibrahim, Shadi, Bingsheng He, and Hai Jin. 2011. “Towards Pay-As-You-Consume Cloud Computing.” *2011 IEEE International Conference on Services Computing*: 370–377. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6009283 (March 9, 2012).

Jarvinen, P. 2007. “Action research is similar to design science.” *Quality & Quantity* 41(1): 37–54.

Jedlitschka, Andreas, Marcus Ciolkowski, and Dietmar Pfahl. 2008a. “Guide to Advanced Empirical Software Engineering.” In *Advanced Topics in Empirical Software Engineering*, London: Springer Verlag, p. 201–228.

———. 2008b. “Reporting Experiments in Software Engineering.” In *Guide to Advanced Empirical Software Engineering*,.

Kalekar, Prajakta S. 2004. 1–13 *Time series Forecasting using Holt-Winters Exponential Smoothing Under the guidance of*.

Kantere, V., D. Dash, G. Gratsias, and A. Ailamaki. 2011. "Predicting Cost Amortization for Query Services." *Proceedings of the 2011 international conference on Management of data*: 325–336.

Keogh, Eamonn, and S. Kasetty. 2003. "On the need for time series data mining benchmarks: a survey and empirical demonstration." *Data Mining and Knowledge Discovery* 7(4): 349–371. http://www.springerlink.com/index/G7535342U0781722.pdf (June 11, 2012).

Kitchenham, B.A. 2007. Technical Report S.o.C.S.a.M. *Guidelines for performing Systematic Literature Reviews in Software Engineering*.

Kuechler, Bill, and Vijay Vaishnavi. 2008. "Theory Development in Design Science Research: Anatomy of a Research Project." *European Journal of Information Systems* 17(5): 489–504.

Le, Kien, Ricardo Bianchini, R Zhang, J. Jaluria, Y Meng, and T.D. Nguyen. 2011. "Reducing Electricity Cost Through Virtual Machine Placement in High Performance Computing Clouds." *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*: 22–34.

Lefèvre, Laurent, and Anne-Cécile Orgerie. 2010. "Designing and evaluating an energy efficient Cloud." *The Journal of Supercomputing* 51(3): 352–373. http://www.springerlink.com/index/10.1007/s11227-010-0414-2 (April 3, 2012).

Li, Ang, Xiaowei Yang, Srikanth Kandula, and Ming Zhang. 2010. "CloudCmp : Comparing Public Cloud Providers." *Proceedings of the 10th annual conference on Internet measurement*: 1–14.

Li, X, Y Li, and T Liu. 2009. "The method and tool of cost analysis for cloud computing." *Cloud Computing, 2009. CLOUD '09. IEEE International Conference on*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5284157 (March 28, 2012).

Lindner, Maik, F Galán, and Clovis Chapman. 2010. "The cloud supply chain: A framework for information, monitoring, accounting and billing." *2nd International ICST Conference on Cloud Computing (CloudComp 2010)*. https://www.ee.ucl.ac.uk/~sclayman/docs/CloudComp2010.pdf (October 1, 2012).

Liu, Chengshui, and Hongmei Yi. 2010. "Research on Power System Load Forecasting Model Based on Data Mining Technology." *2010 International Conference on Intelligent System Design and Engineering Application*: 240–243. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5743416 (June 7, 2012).

Lucas-Simarro, Jose Luis, Rafael Moreno-Vozmediano, Ruben S. Montero, and Ignacio M. Llorente. 2012. "Scheduling strategies for optimal service deployment across multiple clouds." *Future Generation Computer Systems*: 1–11. http://linkinghub.elsevier.com/retrieve/pii/S0167739X12000192 (March 19, 2012).

Mach, Werner, and Erich Schikuta. 2011. "A Consumer-Provider Cloud Cost Model Considering Variable Cost." *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*: 628–635.

Maciel, Paulo Ditarso, Francisco Brasileiro, Ricardo Araújo Santos, David Candeia, Raquel Lopes, Marcus Carvalho, Renato Miceli, Nazareno Andrade, and Miranda Mowbray. 2012. "Business-driven short-term management of a hybrid IT infrastructure." *Journal of Parallel and Distributed Computing* 72(2): 106–119. http://linkinghub.elsevier.com/retrieve/pii/S0743731511002176 (April 4, 2012).

Makridakis, S., S. Wheelwright, and R.J. Hyndman. 1998. *Forecasting - methods and applications*. 3rd ed. New York: John Wiley & Sons.

Mani, Sudha, and Shrisha Rao. 2011. “Operating Cost Aware Scheduling Model for Distributed Servers Based on Global Power Pricing Policies.” *Computer*.

March, S.T., and V.C. Storey. 2008. “Design science in the information systems discipline: an introduction to the special issue on design science research.” *MIS Quarterly* 32(4): 725–730.

Martens, Benedikt, Marc Walterbusch, and Frank Teuteberg. 2012. “Costing of Cloud Computing Services : A Total Cost of Ownership Approach.” *2012 45th Hawaii International Conference on System Sciences*: 1563–1572.

Mazhelis, O., P. Tyrväinen, and Author Mazhelis. 2011. “Role of Data Communications in Hybrid Cloud Costs.” *Proc. of 37th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA2011)*: 138–145. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6068337 (March 28, 2012).

Mazzucco, M. 2010. “Maximizing Cloud Providers’ Revenues via Energy Aware Allocation Policies.” *Cloud Computing (CLOUD) …*. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5558002 (March 28, 2012).

Mell, Peter, and Tim Grance. 2009. “The NIST Definition of Cloud Computing.” *National Institute of Standards and Technology* 53(6): 50. http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc.

Mian, Rizwan, Patrick Martin, and Jose Luis Vazquez-Poletti. 2012. “Provisioning data analytic workloads in a cloud.” *Future Generation Computer Systems*. http://linkinghub.elsevier.com/retrieve/pii/S0167739X12000209 (April 4, 2012).

Mihoob, Ahmed, Carlos Molina-Jimenez, and Santosh Shrivastava. 2010. “A Case for Consumer–centric Resource Accounting Models.” *2010 IEEE 3rd International Conference on Cloud Computing*: 506–512. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5558015 (March 14, 2012).

Moschakis, Ioannis a., and Helen D. Karatza. 2010. “Evaluation of gang scheduling performance and cost in a cloud computing system.” *The Journal of Supercomputing* 59(2): 975–992. http://www.springerlink.com/index/10.1007/s11227-010-0481-4 (March 17, 2012).

Napper, Jeffrey, and P. Bientinesi. 2009. “Can Cloud Computing Reach The TOP500 ?” *Proceedings of the combined workshops on UnConventional high performance computing workshop plus memory access workshop*: 17–20.

Nunamaker, J.F., M. Chen, and T.D.M. Purdin. 1991. “Systems Development in Information Systems Research.” *Journal of Management Information Systems (1991)* 7(3): 89–106.

Oracle. 2011. *ORACLE VM 3: data sheet*.

Owonibi, Michael, and Peter Baumann. 2010. “A Cost Model for Distributed Coverage Processing Services.” *Communication*: 19–26.

Pandey, Gaurav. 2011. “Current Cloud Scenario Review and Cost Optimization by Efficient Resource Provisioning.” *Design*.

Pearl, Judea. 2003. 19 Journal of Econometric Theory 675–685 *Causality: models, reasoning, and inference*. Cambridge: Cambridge Univercity Press.

Peffers, K., T. Tuunanen, C.E. Gengler, M. Rossi, W. Hui, V. Virtanen, and J. Bragge. 2006. “The design science research process: a model for producing and presenting information systems research.” *Proceedings of the* *first international conference on design science research in information systems and technology (DESRIST 2006)*: 83–106.

Pettey, Christy, and Rob van der Meulen. 2012. *Gartner's 2012 Hype Cycle for Emerging Technologies*. Stamford. http://www.gartner.com/it/page.jsp?id=2124315.

Pfleeger, S.L. 1999. "Albert Einstein and empirical software engineering." *Computer* 32(10): 32–38. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=796106.

R Foundation. 2012. "A Language and Environment for Statistical Computing."

Ramakrishnan, Lavanya, and Daniel a. Reed. 2009. "Predictable quality of service atop degradable distributed systems." *Cluster Computing* (December 2008). http://www.springerlink.com/index/10.1007/s10586-009-0078-y (April 4, 2012).

Rosenthal, A, P Mork, and MH Li. 2010. "Cloud computing: A new business paradigm for biomedical information sharing." *Journal of biomedical …*. http://www.sciencedirect.com/science/article/pii/S1532046409001154 (March 28, 2012).

Rossi, M., and M.K. Sein. 2003. "Design research workshop: a proactive research approach." *Action Research* 1(2): 1–20.

Ryan, Jeffrey, and Josh M Ulrich. 2012. *eXtensible Time Series*. http://cran.r-project.org/package=xts.

Sekar, Vyas, and P. Mantis. 2011. "Verifiable Resource Accounting for Cloud Computing Services." *Proceedings of the 3rd ACM workshop on Cloud computing security workshop*: 21–26.

Shakimov, Amre, Landon P Cox, and Alexander Varshavsky. 2009. "Privacy , Cost , and Availability Tradeoffs in Decentralized OSNs." *Proceedings of the 2nd ACM workshop on Online social networks*: 13–18.

Shannak, RO. 2009. "Grounded Theory as a Methodology for Theory Generation in Information Systems Research." *European Journal of Economics* 15(15). http://www.mendeley.com/research/grounded-theory-methodology-theory-generation-information-systems-research/ (June 7, 2012).

Sharma, Upendra, Prashant Shenoy, S. Sahu, and A. Shaikh. 2011. "A Cost-aware Elasticity Provisioning System for the Cloud." *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*: 559–570.

Smith, J.E., and R. Nair. 2005. "The Architecture of Virtual Machines." *Computer* 38(5): 32–38.

Takeda, Hideaki, and Paul Veerkamp. 1990. "Modeling Design Processes." *AI Magazine* 11(4): 37–48.

Tian, Yuan, Biao Song, and Eui-nam Huh. 2011. "Dynamic Content-based Cloud Data Integration System with Privacy and Cost Concern." *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*: 193–199.

Tran, V, and J Keung. 2011. "Application migration to cloud: a taxonomy of critical factors." *… on Software engineering for cloud …*. http://dl.acm.org/citation.cfm?id=1985505 (March 28, 2012).

Truong, Hong-Linh, and Schahram Dustdar. 2010. "Composable cost estimation and monitoring for computational applications in cloud computing environments." *Procedia Computer Science* 1(1): 2175–2184. http://linkinghub.elsevier.com/retrieve/pii/S1877050910002449 (March 9, 2012).

Truong Huu, Tram, Guilherme Koslovski, Fabienne Anhalt, Johan Montagnat, and Pascale Vicat-Blanc Primet. 2010. "Joint Elastic Cloud and Virtual Network Framework for Application Performance-cost Optimization." *Journal of Grid Computing* 9(1): 27–47. http://www.springerlink.com/index/10.1007/s10723-010-9168-6 (March 18, 2012).

VMware. 2007. *VirtualCenter Monitoring and Performance Statistics*.

Verma, Akshat, Gautam Kumar, and Ricardo Koller. 2010. "The Cost of Reconfiguration in a Cloud." *Proceedings of the 11th International Middleware Conference Industrial track*: 11–16.

Wang, Hongyi, Qingfeng Jing, Rishan Chen, Bingsheng He, Zhengping Qian, and Lidong Zhou. 2010. "Distributed Systems Meet Economics: Pricing in the Cloud." *Proceedings of the 2nd USENIX Workshop on Hot Topics in Cloud computing (HotCloud'10)*.

Wang, Lei, and Jianfeng Zhan. 2009. "In Cloud , Do MTC or HTC Service Providers Benefit from the Economies of Scale ?" *Environment*.

Wang, Zhe, Tao Li, Naixue Xiong, and Yi Pan. 2011. "A novel dynamic network data replication scheme based on historical access record and proactive deletion." *The Journal of Supercomputing*. http://www.springerlink.com/index/10.1007/s11227-011-0708-z (April 4, 2012).

Weerd, Inge Van De, and Sjaak Brinkkemper. 2008. "Meta-Modeling for Situational Analysis and Design Methods." In *Handbook of research on modern systems analysis and design technologies and applications*, , p. 38–58.

Weida, Major William J. 1977. DEPT OF ECONOMICS, GEOGRAPHYA AND MANAGEMENT *A general technique for R&D cost forecasting*. Colorado.

Witten, I.H., E. Frank, and M.A. Hall. 2011. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufman.

Wu, Linlin, Saurabh Kumar Garg, and Rajkumar Buyya. 2011. "SLA-based admission control for a Software-as-a-Service provider in Cloud computing environments." *Journal of Computer and System Sciences* 1: 1–20. http://linkinghub.elsevier.com/retrieve/pii/S0022000011001590 (April 4, 2012).

Wu, Zhangjun, Xiao Liu, Zhiwei Ni, Dong Yuan, and Yun Yang. 2011. "A market-oriented hierarchical scheduling strategy in cloud workflow systems." *The Journal of Supercomputing*. http://www.springerlink.com/index/10.1007/s11227-011-0578-4 (March 12, 2012).

Yi, Sangho. 2011. "Monetary Cost-Aware Checkpointing and Migration on Amazon Cloud Spot Instances." *IEEE Transactions on Services Computing*: 1–14.

Yi, Sangho, Derrick Kondo, and Artur Andrzejak. 2010. "Reducing Costs of Spot Instances via Checkpointing in the Amazon Elastic Compute Cloud." *2010 IEEE 3rd International Conference on Cloud Computing* (January): 236–243. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5557987 (March 14, 2012).

Young, Warren L. 1976. "Forecasting types and forecasting techniques - a taxonomic approach." *Journal of Quality and Quantity* 10(2): 165–170.

Zhai, Yan, Mingliang Liu, and Jidong Zhai. 2011. "Cloud Versus In-house Cluster : Evaluating Amazon Cluster Compute Instances for Running MPI Applications." *Proceeding SC '11 State of the Practice Reports*.

Zhang, Guoqiang, and B Eddy Patuwo. 1998. "Forecasting with artificial neural networks: The state of the art." *International journal of forecasting* 14: 35–62. http://www.sciencedirect.com/science/article/pii/S0169207097000447 (June 8, 2012).

Zhang, Qi, Lu Cheng, and Raouf Boutaba. 2010. "Cloud computing: state-of-the-art and research challenges." *Journal of Internet Services and Applications* 1(1): 7–18. http://www.springerlink.com/index/10.1007/s13174-010-0007-6.

Zhang, Xinwen, Anugeetha Kunjithapatham, Sangoh Jeong, and Simon Gibbs. 2011. "Towards an Elastic Application Model for Augmenting the Computing Capabilities of Mobile Devices with Cloud Computing." *Mobile Networks and Applications* 16(3): 270–284. http://www.springerlink.com/index/10.1007/s11036-011-0305-7 (March 10, 2012).

Zhao, Laiping, Yizhi Ren, Mingchu Li, and Kouichi Sakurai. 2012. "Flexible service selection with user-specific QoS support in service-oriented architecture." *Journal of Network and Computer Applications* 35(3): 962–973. http://linkinghub.elsevier.com/retrieve/pii/S1084804511000671 (April 4, 2012).

Zheng, Qin, and Bharadwaj Veeravalli. 2012. "Utilization-based pricing for power management and profit optimization in data centers." *Journal of Parallel and Distributed Computing* 72(1): 27–34. http://linkinghub.elsevier.com/retrieve/pii/S0743731511001730 (April 4, 2012).

Zheng, Xinying, and Yu Cai. 2011. "Energy-aware load dispatching in geographically located Internet data centers." *Sustainable Computing: Informatics and Systems* 1(4): 275–285. http://linkinghub.elsevier.com/retrieve/pii/S221053791100045X (April 4, 2012).

Appendices

List of appendices:

- Appendix A: Structured literature research cost variables
- Appendix B: Structured literature research core concepts mapping
- Appendix C: Literature research classification matrix
- Appendix D: Forecasting data retrieval script in SQL
- Appendix E: Forecasting scripts using R
- Appendix F: VMware database design
- Appendix G: Results from the Forecasting experiment
- Appendix H: Expert review protocol

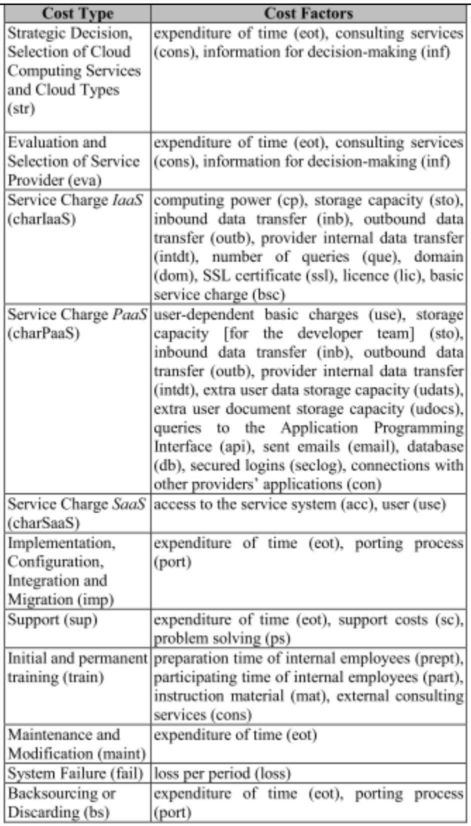
Appendix A: Structured literature research cost variables

Paper	Cost variables and measures	Cost driver and relationship	Remark
(Ardagna et al. 2012)	Average response time VM (hourly, flat fee or on demand)	SLA + Workload Higher SLA requirements increase costs. VM's can are cheaper using flat fee than on demand.	N/A
(M. D. Assunção, Costanzo, and Buyya 2010)	Virtual machines Storage	N/A	N/A
(M. D. D. Assunção, Costanzo, and Buyya 2009)	VM	Batch processing 1VM per x batches	Scheduling
(Bai et al. 2011)	Storage in GB Network measured in bandwidth and throughput Computing capacity Geographical location Value added services (monitoring, remote administration, alerts)	Capacity and storage time Network = capacity / time Time or amount, time of day Country Cost per service	N/A
(Boutaba, Cheng, and Q. Zhang 2011)	Job heterogeneity	Variety Costs are positively related to variety in heterogeneity.	N/A
(Breskovic et al. 2011)	SLO/SLA SLA contract maintenance SLA contract set-up	Degree of service Times of change Degree of customization High service increases costs	N/A
(Byun et al. 2011)	Instance VM	Hours	N/A
(Caron, Frédéric Desprez, and Muresan 2011)	Instance VM Instance type	per hour bases. Capacity proportional to expense	On Amazon EC2 m1 instance 0,095.
(Sivadon Chaisiri, Lee, and Niyato 2011)	Timeliness of resource planning	Planning or on demand utilization Planning is cheaper than on demand.	
(Chard et al. 2011)	Instance VM Instance type Capacity (cores, memory, architecture, storage)	Hours Fixed instances have x capacity Instance type determines capacity	A test indicates that it is more cost effective to run several small instances than less to one big instance resulting in equal capacity.
(Y. Chen and Sion 2011)	number of servers,switches administrator : server ratio watt per sq ft server switch price personel, floor cost/sec electricity price/watt (sec) CPU utalisation	Floor surface sqft price can be brought down much lower, often amortized to zero over time. Floor surface is directly related to power consumption and cooling with designs supporting anywhere from 40 to 250 watt/sqft. Overall power requirements (driven by CPUs) impact directly the required floor space	Server hardware Energy Service Network hardware Floor space

	CPU frequency Server/switch lifespan Server power at peak, idle		
(Csorba, Meling, and Heegaard 2011)	Cost of service deployment Instance cost Communication cost QoS	Workload / instance / hour	N/A
(Deelman et al. 2008)	Storage GB/month Bandwith in Bandwith out CPU hours	Persistent storage Communications Communications Capacity	CCR communication to computation ration determines the degree of
(Dougherty, White, and D. C. Schmidt 2012)	N/A	N/A	N/A
(Ferrer et al. 2012)	Maintenance Infrastructure characteristics (redundance, etc.) Security Customer support Legal aspects	Non-functions aspects in SLA's.	N/A
(Garg et al. 2011)	Energy, carbon emission rate, workload, CPU efficiency	N/A	N/A
(Ghanbari et al. 2012)	CPU capacity, SLA	Server load, response time	N/A
(Gmach, Rolia, and Cherkasova 2010)	Acquisition costs for facilities, physical IT equipment and software, power costs for operating the physical machines and facilities, and administration costs. Acquisition costs are often considered with respect to a three year time horizon and reclaimed according to an assumed rate for each costing interval. Without loss of generality, this paper focuses on server and virtualization software licensing costs only. Below, we define three categories of resource usage that can be tracked separately for each server resource, e.g., CPU, memory, for each costing interval. To simplify the notation, the equations that we present consider only one Server resource at a time, e.g., CPU or memory for one costing interval	We define CPU capacity and CPU demand in units of CPU shares. A CPU share denotes one percentage of utilization of a processor with a clock rate of 1 GHz. Direct resource notation ds,w represents consumption by a workload: the average physical server utilization of a server s by a workload w. Burstiness for a workload and for a server: the notation bs,w represents the difference between peak utilization of a server s by workload w and its average utilization represented by ds,w. Unallocated resource for a server: the notation as represents unallocated (unused) server capacity; it is defined as the difference between 100 and the peak utilization of server s.	N/A
(Greenberg et al. 2008)	Infrastructure • Servers (CPU, memory, storage)	45% servers 25% infrastructure	N/A

	• Infra (power distribution, cooling) • Electricity • Network (links, transit, equipment)	15% power draw 15% network	
(Haak and Menzel 2011)	CPU, Ram performance, network latency, Benchmark costs: Staff, licences, infrastructure usage, network traffic QoS costs	QoS costs: response time, throughput, availability	N/A
(Kantere et al. 2011)	Query execution	Time of execution Proportional to time.	"Query performance is measured in terms of execution time. The faster the execution, the more expensive the service"
(Le et al. 2011)	Total electricity cost ($) of the service The set of data centers Cost for energy consumed at data center d Cost for peak power at data center d Avg. power demand (KW) at data center d in epoch t Avg. dynamic power demand of one loaded server Energy price ($/KWh) at data center d in epoch t Peak power price ($/KW) at data center d, o is on-peak or off-peak	Set of active servers at data center d in epoch t Avg. utilization (%) of server s in epoch t Run time of job j The number of VMs comprising job j Avg. dynamic power demand at data center d in epoch t Avg. cooling power demand at data center d in epoch t Base power demand of one idle server Avg. dynamic power demand of one loaded server	N/A
(Lefèvre and Orgerie 2010)	Energy costs	Computational work	
(A. Li et al. 2010)	CPU resource CPU speed, Memory, Disk I/O, scaling latency, storage service response time, time to reach consistency, network latency, bandwidth.	CPU cycles	Public cloud providers all offer: - Elastic compute cluster - Persistent storage - Intra-cloud network - Wide-area network
(X. Li, Y. Li, and T. Liu 2009)	Server, software, network, support and maintenance, power, cooling, facilities, real-estate.	Power is driven by workload Servers is driven by amount of customers and risk of provisioning Cooling is proportional to energy consumption Real-estate is costs are calculated through amount of racks Facility costs are incurred for every rack	N/A
(Lucas-Simarro et al. 2012)	Brokering services (interface selection, instance type selection, pricing model selection, etc.) Cores, ram, disk storage,	N/A	N/A

(Mach and Schikuta 2011)	Rack costs, server costs, power cost, network costs **Depreciation** of servers, hard disks, occupancy cost, administration cost, power consumption cost during idle time and the cost of network gears like router and switches are fixed cost. **Administration** of hard disc and server	N/A	Additional metrics: Server occupancy cost **Production factors:** Storage devices, servers, network devices **Produced goods** Storage capacity, performance, network bandwidth
(Maciel et al. 2012)	Number of machines and cycles	N/A	N/A
(Mani and Rao 2011)	Power SLA Geographic location	N/A	N/A
(Martens, Walterbusch, and Teuteberg 2012)	IaaS: instance/hour, ram, transferred GB, storage, BIT platform, price per query, hourly support, licence, SSL certificates. PaaS: price per user, storage, data transfer, computing power, emails, storage, computing power per hour, size of database, pay per feature or API. SaaS: api-calls, monthly charge Project costs: descision making, evaluation service providers, service charge, implementation, configuration, integration, migrionat, support, training, maintenance, modification, system failures, backsourcing or discarding.	Unfortunately no direct relation between cost driver and variables.	Also TCO

			
(O. Mazhelis, Tyrväinen, and A. Mazhelis 2011)	Operating costs Resource costs Data communications cost Computing capacity Hardware Software Storage	Load Portion on premises or cloud	N/A
(Mian, Martin, and Vazquez-Poletti 2012)	SLA Capacity VM's Datastorage Storing VM images Data communications	Workload Service	N/A

Cost Type	Cost Factors
Strategic Decision, Selection of Cloud Computing Services and Cloud Types (str)	expenditure of time (eot), consulting services (cons), information for decision-making (inf)
Evaluation and Selection of Service Provider (eva)	expenditure of time (eot), consulting services (cons), information for decision-making (inf)
Service Charge *IaaS* (charIaaS)	computing power (cp), storage capacity (sto), inbound data transfer (inb), outbound data transfer (outb), provider internal data transfer (intdt), number of queries (que), domain (dom), SSL certificate (ssl), licence (lic), basic service charge (bsc)
Service Charge *PaaS* (charPaaS)	user-dependent basic charges (use), storage capacity [for the developer team] (sto), inbound data transfer (inb), outbound data transfer (outb), provider internal data transfer (intdt), extra user data storage capacity (udats), extra user document storage capacity (udocs), queries to the Application Programming Interface (api), sent emails (email), database (db), secured logins (seclog), connections with other providers' applications (con)
Service Charge *SaaS* (charSaaS)	access to the service system (acc), user (use)
Implementation, Configuration, Integration and Migration (imp)	expenditure of time (eot), porting process (port)
Support (sup)	expenditure of time (eot), support costs (sc), problem solving (ps)
Initial and permanent training (train)	preparation time of internal employees (prept), participating time of internal employees (part), instruction material (mat), external consulting services (cons)
Maintenance and Modification (maint)	expenditure of time (eot)
System Failure (fail)	loss per period (loss)
Backsourcing or Discarding (bs)	expenditure of time (eot), porting process (port)

	Cost of missing SLO		
(Moschakis and Karatza 2010)	Virtualisation software VM’s leased	Lease time	
(Napper and Bientinesi 2009)	Cores Instances Ram Network capacity	Computation time (per hour)	N/A
(Owonibi and Baumann 2010)	Costs of data transfer Data i/o Data decoding and encoding Calibration cost	N/A	
(Pandey 2011)	Instance size - VMs Machine instance Disk storage Memory consumption Data I/O Input/output bandwidth	N/A	
(Ramakrishnan and Reed 2009)	QoS Instance capacity	N/A	
(Sekar and Mantis 2011)	I.O time Internal network bandwith Memory Bandwidth	N/A	
(Shakimov, Cox, and Varshavsky 2009)	Data transfer costs Instance QoS availability costs	N/A	
(Sharma et al. 2011)	Instance Amount of cores Ram Disk Set-up time (transition costs) Non-functional: location	Instance per hour cost	Server size is a consumption package
(Tian, Song, and Huh 2011)	Data uploading Query execution	N/A	
(Tran and Keung 2011)	Migration costs, redevelopment, administrative costs SaaS; software development costs, integration costs, customization costs, subscription costs. IaaS: middleware, hardware costs, resource costs, storage cost, data IO cost,	N/A	Dominant is effort costs.

	Direct costs (facility, energy, cables, servers)		
(Hong-Linh Truonga,∗ 2010)	Computational capacity Storage Bandwidth	Charge per hour, week, month	Values are sealed
(Truong Huu et al. 2010)	CPU Storage Network transfer Non-functional: Security	For storage (size, storage time) Machine (hour) Data (in/out MB)	
(Verma, Kumar, and Koller 2010)	VM instances CPU clock frequency Reconfiguration cost Power Software licenses	Reconfiguration decreases actual capacity and takes time	VM migration or resizing
(L. Wang and Zhan 2009)	Instance hours Overhead (claiming, releasing VMs) CPU Memory RAM Storage	N/A	N/A
(Z. Wang et al. 2011)	Network traffic Data storage duration Server	N/A	N/A
(L. Wu, Kumar Garg, and Buyya 2011)	QoS - Service initiation time - Service time (end user) Adminstration Maintenance cost Processing power Bandwith Power Storage SLA violation cost	VM – price per hour Data transfer – quantity Initiation – through server load	N/A
(Z. Wu et al. 2011)	VM cost SLA cost	N/A	N/A
(S. Yi, Kondo, and Andrzejak 2010)	Instance Total datacenter workload (spot prices) Location Computation power	N/A	Spot price
(Y. Zhai, M. Liu, and J. Zhai 2011)	Amount of cores Memory Inter-server communication Disk storage	N/A	N/A

	NIC costs + switch costs Rack costs		
(X. Zhang et al. 2011)	CPU cycles Storage Communication traffic (in . out) Intra-cloud communication	N/A	N/A
(Zhao et al. 2012)	Instance (core, speed) Ram Storage Platform	N/A	N/A
(Q. Zheng and Veeravalli 2012)	SLA Capital cost Power cost Instance cost	N/A	N/A
(X. Zheng and Cai 2011)	Electricity + UPS Cooling Network cost	Workload Server density Active servers Network equipment power consumption UPS power consumption Power distribution consumption	N/A

Appendix B: Structured literature research core concepts mapping

Paper	Core concepts	Opposition Towards cloud	Opposition rationale
(Ardagna et al. 2012)	Scaling, private, public clouds, cloud economics, more economically then internal data centers, Cloud systems Performance modeling Resource management Capacity allocation Load balancing, SLA	Possitive	Low cost
(M. D. Assunção, Costanzo, and Buyya 2010)	Organization and design, distributed systems, computer communicatin networks, design, management, performance	Neutral	
(M. D. D. Assunção, Costanzo, and Buyya 2009)	Hybrid deployment, performance, cloud design, infrastructure as a service, scheduling, SLA	Possitive	Low cost
(Bai et al. 2011)	Cloud computing, charge-model, petri-net, infrastructure layer, infrastructure as a service	Possitive	Low cost
(Boutaba, Cheng, and Q. Zhang 2011)	cloud computing; heterogeneity; MapReduce; resource allocation; scheduling	Possitive	Low cost
(Breskovic et al. 2011)	Service level agreement; cloud economics; electronic markets; mapping; SLA; SLA management	Possitive	Low cost
(Byun et al. 2011)	Resource capacity estimation Resource allocation, Resource allocation, Application workflow, Cloud computing economy	Neutral	
(Caron, Frédéric Desprez, and Muresan 2011)	Workload prediction; auto-scaling; cloud computing	Possitive	Low cost
(Sivadon Chaisiri, Lee, and Niyato 2011)	Cloud computing, resource provisioning, virtualization, virtual machine placement, stochastic programming	Possitive	Low cost
(Chard et al. 2011)	Cloud, Scaling, workload distribution, public cloud	Possitive	Low cost
(Y. Chen and Sion 2011)	Cloud, performance, economics	Possitive	Low cost
(Csorba, Meling, and Heegaard 2011)	Service Deployment, Biologically-inspired Systems, Decentralized Optimization	Possitive	Low cost
(Deelman et al. 2008)	Cloud, costs	Possitive	Low cost
(Dougherty, White, and D. C. Schmidt 2012)	Cloud computing, Auto-scaling, Power optimization, Model-driven engineering	Possitive	Low cost
(Ferrer et al. 2012)	Cloud service, cloud optimization, SLA, cloud architecture, legislation	Possitive	Low cost
(Garg et al. 2011)	Cloud computing, High Performance Computing (HPC), Energy-efficient scheduling, Dynamic Voltage Scaling (DVS), Green IT	Possitive	Low cost
(Ghanbari et al. 2012)	Optimization, Modeling, State estimation, Private cloud, PaaS, IaaS	Possitive	Low cost
(Gmach, Rolia, and Cherkasova 2010)	component; as opposed to pricing; bid; burstiness; cost models; customers are willing to; or what; pay for resources; recovery or chargeback; resource sharing; virtualization; workload placement	Possitive	Low cost
(Greenberg et al. 2008)	Cloud-service data centers, costs, network challenges	Possitive	Low cost
(Kantere et al. 2011)	Database services, cost estimation, public cloud, database as a service (DaaS),	Possitive	Low cost
(Haak and Menzel 2011)	Cloud Computing, Benchmarking, Theory of Optimal Stop- ping, Self-Optimizing, Self-Configuring, Economic Model	Possitive	Low cost
(Le et al. 2011)	Multi-data-center, computing cloud, cooling, energy	Possitive	Low cost

(Lefèvre and Orgerie 2010)	Energy efficiency · Clouds · Virtualization · Migration · Energy awareness · Power management	Possitive	Low cost
(A. Li et al. 2010)	Cloud computing, comparison, performance, cost	Possitive	Low cost
(X. Li, Y. Li, and T. Liu 2009)	Cloud, cost analysis, method and tools	Possitive	Low cost
(Lucas-Simarro et al. 2012)	Cloud brokering, multi-cloud, scheduling algorithm, resource allocation, infrastructure as a service	Possitive	Low cost
(Mach and Schikuta 2011)	cloud computing; cost model; business strategies; cloud consumer; cloud provider	Possitive	Low cost
(Maciel et al. 2012)	Cloud computing, grid computing, peer-to-peer, business-driven IT management, short-term management, capacity planning	Possitive	Low cost
(Mani and Rao 2011)	Distributed servers, scheduling, operating cost, energy cost, economics, algorithms, performance, management	Possitive	Low cost
(Martens, Walterbusch, and Teuteberg 2012)	Cloud costing, TCO, computing services, systematic literature review	Possitive	Low cost
(O. Mazhelis, Tyrväinen, and A. Mazhelis 2011)	Hybrid cloud, cost model, data communication costs	Possitive	High cost
(Mian, Martin, and Vazquez-Poletti 2012)	Cloud computing, data analytics, workload management, resource provisioning	Possitive	Low cost
(Moschakis and Karatza 2010)	Cloud computing · Gang scheduling · HPC · Virtual machines	Possitive	Low cost
(Napper and Bientinesi 2009)	Cloud computing, amazon, HPL, measurement, performance	Possitive	Low cost
(Owonibi and Baumann 2010)	CIPS, Distributed processing, cost model, geo-processing, design, experimentation, performance	Possitive	Low cost
(Pandey 2011)	Cloud computing, pricing, application design, resource provisioning	Possitive	Low cost
(Ramakrishnan and Reed 2009)	Performability, reliability, workflow scheduling, fault tolerance, gird and cloud resource management	Neutral	
(Shakimov, Cox, and Varshavsky 2009)	Cloud computing, online social networks, privacy, replication, utility computing, virtual machines, design, performance, reliability, security	Neutral	
(Sharma et al. 2011)	Cloud, infrastructure, scalability, elasticity, scheduling, workflow management	Possitive	Low cost
(Tian, Song, and Huh 2011)	Cloud, costs affectivity, economics, privacy	Possitive	Low cost
(Tran and Keung 2011)	Cloud computing, utility computing, cost-benefit analysis, software metrics, cost factors and overheads, deployment strategy, taxonomy, design, economics, deployment	Possitive	Low cost
(Hong-Linh Truonga,∗ 2010)	Cloud computing, resources allocation, IaaS, workflows, network virtualization, description language	Possitive	Low cost
(Truong Huu et al. 2010)	Cloud computing, cost monitoring, cost analysis, cost estimation, scientific applications	Possitive	Low cost
(Verma, Kumar, and Koller 2010)	Cloud computing, reconfiguration cost, virtualization	Possitive	Low cost
(L. Wang and Zhan 2009)	Many task computing, high throughput computing, cloud computing, service providers, economies of scale, design, management, performance	Possitive	Low cost
(Z. Wang et al. 2011)	Data replication, read overhead, update overhead, historical access record, proactive deletion	Possitive	Low cost
(L. Wu, Kumar Garg, and	Cloud computing, service level agreement, admission control, software as a service, scalability of application services	Possitive	Low cost

Buyya 2011)			
(Z. Wu et al. 2011)	Cloud workflow system, cloud computing, workflow scheduling, hierarchical scheduling, meta-heuristics	Possitive	Low cost
(S. Yi, Kondo, and Andrzejak 2010)	Check pointing, Reliability, fault-tolerance, cloud computing, volatile resources	Possitive	Low cost
(Y. Zhai, M. Liu, and J. Zhai 2011)	Cloud computing, grid computing, MPI applications, HPC applications	Neutral	
(X. Zhang et al. 2011)	Elastic application, cloud computing, mobile device, weblet, dynamic execution, configuration	Possitive	Low cost
(Zhao et al. 2012)	Quality of service, web service, scheduling, service-oriented architecture	Neutral	
(Q. Zheng and Veeravalli 2012)	Utilization, pricing, load balancing, revenue, power, cost	Neutral	Neutral
(X. Zheng and Cai 2011)	Internet data centers, cooling management, electricity market, load dispatching, energy proportional	Possitive	Low cost

Appendix C: Literature research classification matrix

Cost variable	Amount	Literature
VM	31	(Ardagna et al. 2012; M. D. Assunção, Costanzo, and Buyya 2010; M. D. D. Assunção, Costanzo, and Buyya 2009; Bai et al. 2011; Byun et al. 2011; Caron, Frédéric Desprez, and Muresan 2011; Chard et al. 2011; Y. Chen and Sion 2011; Csorba, Meling, and Heegaard 2011; Deelman et al. 2008; Garg et al. 2011; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; Hong-Linh Truonga,∗ 2010; X. Li, Y. Li, and T. Liu 2009; Lucas-Simarro et al. 2012; Martens, Walterbusch, and Teuteberg 2012; O. Mazhelis, Tyrväinen, and A. Mazhelis 2011; Mian, Martin, and Vazquez-Poletti 2012; Moschakis and Karatza 2010; Napper and Bientinesi 2009; Pandey 2011; Ramakrishnan and Reed 2009; Shakimov, Cox, and Varshavsky 2009; Sharma et al. 2011; Tran and Keung 2011; Verma, Kumar, and Koller 2010; L. Wang and Zhan 2009; Z. Wang et al. 2011; Z. Wu et al. 2011; S. Yi, Kondo, and Andrzejak 2010; Q. Zheng and Veeravalli 2012)
Network	24	(Bai et al. 2011; Csorba, Meling, and Heegaard 2011; Deelman et al. 2008; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; Haak and Menzel 2011; X. Li, Y. Li, and T. Liu 2009; Lucas-Simarro et al. 2012; Martens, Walterbusch, and Teuteberg 2012; O. Mazhelis, Tyrväinen, and A. Mazhelis 2011; Mian, Martin, and Vazquez-Poletti 2012; Napper and Bientinesi 2009; Owonibi and Baumann 2010; Pandey 2011; Sekar and Mantis 2011; Shakimov, Cox, and Varshavsky 2009; Tian, Song, and Huh 2011; Tran and Keung 2011; Truong Huu et al. 2010; Z. Wang et al. 2011; L. Wu, Kumar Garg, and Buyya 2011; Y. Zhai, M. Liu, and J. Zhai 2011; X. Zhang et al. 2011; X. Zheng and Cai 2011)
Storage	23	(M. D. Assunção, Costanzo, and Buyya 2010; Bai et al. 2011; Chard et al. 2011; Deelman et al. 2008; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; Hong-Linh Truonga,∗ 2010; A. Li et al. 2010; Lucas-Simarro et al. 2012; Martens, Walterbusch, and Teuteberg 2012; O. Mazhelis, Tyrväinen, and A. Mazhelis 2011; Mian, Martin, and Vazquez-Poletti 2012; Pandey 2011; Sekar and Mantis 2011; Sharma et al. 2011; Tran and Keung 2011; Verma, Kumar, and Koller 2010; L. Wang and Zhan 2009; Z. Wang et al. 2011; L. Wu, Kumar Garg, and Buyya 2011; Y. Zhai, M. Liu, and J. Zhai 2011; X. Zhang et al. 2011; Zhao et al. 2012)
Memory	15	(Chard et al. 2011; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; A. Li et al. 2010; X. Li, Y. Li, and T. Liu 2009; Lucas-Simarro et al. 2012; Martens, Walterbusch, and Teuteberg 2012; Napper and Bientinesi 2009; Pandey 2011; Sekar and Mantis 2011; Sharma et al. 2011; Verma, Kumar, and Koller 2010; L. Wang and Zhan 2009; Y. Zhai, M. Liu, and J. Zhai 2011; Zhao et al. 2012)
Energy	15	(Y. Chen and Sion 2011; Garg et al. 2011; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; Haak and Menzel 2011; Le et al. 2011; Lefèvre and Orgerie 2010; Mach and Schikuta 2011; Mani and Rao 2011; Tran and Keung 2011; Verma, Kumar, and Koller 2010; L. Wu, Kumar Garg, and Buyya 2011; X. Zhang et al. 2011; Q. Zheng and Veeravalli 2012; X. Zheng and Cai 2011)
CPU	15	(Y. Chen and Sion 2011; Garg et al. 2011; Ghanbari et al. 2012; Gmach, Rolia, and Cherkasova 2010; Greenberg et al. 2008; Haak and Menzel 2011; A. Li et al. 2010; Martens, Walterbusch, and Teuteberg 2012; O. Mazhelis, Tyrväinen, and A. Mazhelis 2011; Sharma et al. 2011; Truong Huu et al. 2010; Verma, Kumar, and Koller 2010; L. Wu, Kumar Garg, and Buyya 2011; S. Yi, Kondo, and Andrzejak 2010; Y. Zhai, M. Liu, and J. Zhai 2011; Zhao et al. 2012)
SLA/SLO	13	(Breskovic et al. 2011; Sivadon Chaisiri, Lee, and Niyato 2011; Csorba, Meling, and Heegaard 2011; Ghanbari et al.

		2012; Haak and Menzel 2011; Mani and Rao 2011; Martens, Walterbusch, and Teuteberg 2012; Mian, Martin, and Vazquez-Poletti 2012; Ramakrishnan and Reed 2009; Shakimov, Cox, and Varshavsky 2009; L. Wu, Kumar Garg, and Buyya 2011; Z. Wu et al. 2011; Q. Zheng and Veeravalli 2012)
Licence en software costs	4	(Haak and Menzel 2011; Lucas-Simarro et al. 2012; Moschakis and Karatza 2010; Verma, Kumar, and Koller 2010)
Geographical location	4	(Bai et al. 2011; Mani and Rao 2011; Sharma et al. 2011; S. Yi, Kondo, and Andrzejak 2010)
Maintenance	3	(Ferrer et al. 2012; X. Li, Y. Li, and T. Liu 2009; L. Wu, Kumar Garg, and Buyya 2011)
Personnel	3	(Y. Chen and Sion 2011; Haak and Menzel 2011; Martens, Walterbusch, and Teuteberg 2012)
Cooling	2	(Gmach, Rolia, and Cherkasova 2010; X. Zheng and Cai 2011)
Security	2	(Ferrer et al. 2012; Truong Huu et al. 2010)
Migration	2	(Martens, Walterbusch, and Teuteberg 2012; Tran and Keung 2011)
Job heterogeneity	1	(Boutaba, Cheng, and Q. Zhang 2011)
Response Time	1	(Ardagna et al. 2012)
Deployment cost	1	(Csorba, Meling, and Heegaard 2011)
Customer support	1	(Ferrer et al. 2012)
Legal aspects	1	(Ferrer et al. 2012)
Value added services	1	(Bai et al. 2011)
Query execution	1	(Kantere et al. 2011)
Real estate	1	(X. Li, Y. Li, and T. Liu 2009)
Rack costs	1	(Mach and Schikuta 2011)

Appendix D: Forecasting data retrieval script in SQL

```
/****** Script for Memory statistics ******/
SELECT CONVERT(VARCHAR(10), SAMPLE_TIME, 103) AS [DATE], (SUM(STAT_VALUE)/1024) as
STAT_VALUE
 FROM [VIM_VCDB5].[dbo].[VPXV_HIST_STAT_YEARLY]
 where stat_id = 98
 and ENTITY like'vm%'
 and SAMPLE_TIME >=  '2012-03-01'
 group by (SAMPLE_TIME)
 order by SAMPLE_TIME
/****** Script for Network statistics  ******/
SELECT CONVERT(VARCHAR(10), SAMPLE_TIME, 103) AS [DATE], (SUM(STAT_VALUE)) as STAT_VALUE
 FROM [VIM_VCDB5].[dbo].[VPXV_HIST_STAT_YEARLY]
 where stat_id = 143
 and ENTITY like'vm%'
 and SAMPLE_TIME >=  '2012-03-01'
 group by (SAMPLE_TIME)
 order by SAMPLE_TIME
/****** Script for storage statistics  ******/
SELECT CONVERT(VARCHAR(10), SAMPLE_TIME, 103) AS [DATE], (SUM(STAT_VALUE)) as STAT_VALUE
 FROM [VIM_VCDB5].[dbo].[VPXV_HIST_STAT_YEARLY]
 where stat_id = 125
 and ENTITY like'vm%'
 and SAMPLE_TIME >=  '2012-03-01'
 group by (SAMPLE_TIME)
 order by SAMPLE_TIME
/****** Script for cpu statistics  ******/
SELECT CONVERT(VARCHAR(10), SAMPLE_TIME, 103) AS [DATE], (SUM(STAT_VALUE)) as STAT_VALUE
 FROM [VIM_VCDB5].[dbo].[VPXV_HIST_STAT_YEARLY]
 where stat_id = 6
 and ENTITY like'vm%'
 and SAMPLE_TIME >=  '2012-03-01'
 group by (SAMPLE_TIME)
 order by SAMPLE_TIME

/****** Script for exploration of VM's.
Script returns a list of vm's for which an a single explored by using a single
statisitic.
The script returns descriptive statistics grouped by unique entity.
 ******/
  SELECT  ENTITY as [VM ID]
              , min(SAMPLE_TIME) AS [STARTDATE]
              , count(*) as [DAYS MEASURED]
              , SUM([STAT_VALUE]) as [SUM VALUE]
              , AVG([STAT_VALUE]) as [AVG VALUE]
              , STDEV([STAT_VALUE]) as [STDEV VALUE]
              , MIN([STAT_VALUE]) as [MIN VALUE]
              , MAX([STAT_VALUE]) as [MAX VALUE]
  FROM [VIM_VCDB].[dbo].[VPXV_HIST_STAT_YEARLY]
    where STAT_ID  = 6 /* Manually enter an id of a statistic */
  and ENTITY like 'host%' /* Use this to limit to host-% to inspect hosts only or vm-%
for vm's. Use % to inspect everything.*/
  group by ENTITY
  order by [DAYS MEASURED] desc
```

Appendix E: Forecasting scripts using R

SCRIPTS FOR THE FORECASTING GRAPHS

```
// Set-up script
library("astsa")
library("xts")
library("forecast")

// Forecast script - CPU
vcstats <- read.csv("~/vcenterstatvalue-cpu.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
plot(forecast(x.ets,31),ylab="CPU usage",xlab="Day in 2012",xaxt="n",
yaxt="n")
axis(1,at=seq(60,290,5),seq(60,290,5))
axis(2,at=seq(0,40000,1000), seq(0,40000,1000))

// Forecast script - Memory
vcstats <- read.csv("~/vcenterstatvalue-memory.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
plot(forecast(x.ets,31),ylab="Memory usage",xlab="Day in
2012",xaxt="n", yaxt="n")
axis(1,at=seq(60,290,5),seq(60,290,5))
axis(2,at=seq(170000,300000,10000), seq(170000,300000,10000))

// Forecast script - storage
vcstats <- read.csv("~/vcenterstatvalue-storage.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
plot(forecast(x.ets,31),ylab="Storage usage",xlab="Day in
2012",xaxt="n", yaxt="n")
axis(1,at=seq(60,290,5),seq(60,290,5))
axis(2,at=seq(170000,300000,10000), seq(170000,300000,10000))

// Forecast script - network
vcstats <- read.csv("~/vcenterstatvalue-network.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
plot(forecast(x.ets,31),ylab="Network usage",xlab="Day in
2012",xaxt="n", yaxt="n")
axis(1,at=seq(60,290,5),seq(60,290,5))
axis(2,at=seq(0,50000,1000), seq(0,50000,1000))
```

Scripts for the error comparison

```
// Set-up script
library("astsa")
library("xts")
library("forecast")

// Forecast script - CPU
vcstats <- read.csv("~/vcenterstatvalue-cpu.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
//auto detect
summary(ets(x.ts))
//different models
summary(ets(x.ts,model="MNN"))
summary(ets(x.ts,model="MNA"))
summary(ets(x.ts,model="MNN"))
summary(ets(x.ts,model="MAN"))
summary(ets(x.ts,model="MAA"))
summary(ets(x.ts,model="MAM"))
summary(ets(x.ts,model="MMM"))
summary(ets(x.ts,model="MMA"))
summary(ets(x.ts,model="MNN"))
vcstats <- read.csv("~/vcenterstatvalue-cpu-short.csv")
> vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
> x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
> x.ts = ts(x, freq=1, start=60)
> x.ets <- ets(x.ts)
>
> summary(forecast(x.ets))
```

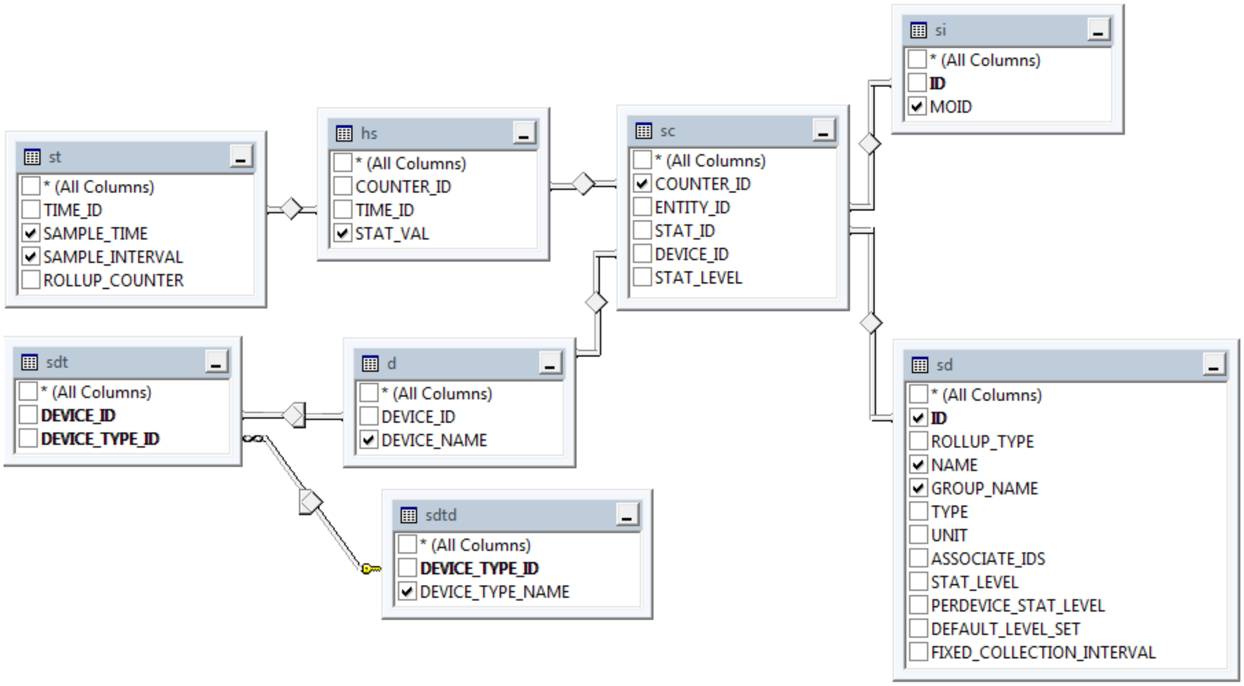
Scripts for forecasting horizon experiment

```
// Set-up script
library("astsa")
library("xts")
library("forecast")

// Forecast script - CPU
//For this experiment a file is used containing data to 8st of August
vcstats <- read.csv("~/vcenterstatvalue-cpu-short.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
summary(forecast(x.ets,7))
summary(forecast(x.ets,14))
summary(forecast(x.ets,21))
summary(forecast(x.ets,28))
summary(forecast(x.ets,31))
```

Scripts for second forecasting horizon experiment

```
// Set-up script
library("astsa")
library("xts")
library("forecast")

// Forecast script - CPU
//For this experiment a file is used containing data to 1st of Juli
vcstats <- read.csv("~/vcenterstatvalue-cpu-short.csv")
vcstats$ï..DATE = as.Date(vcstats$ï..,format="%d/%m/%Y")
x = xts(x=vcstats$STAT_VALUE, order.by=vcstats$ï..DATE)
x.ts = ts(x, freq=1, start=60)
x.ets <- ets(x.ts)
summary(forecast(x.ets,50))
summary(forecast(x.ets,62))

//real values
vcstatsresult <- read.csv("~/vcenterstatvalue-cpu - results.csv")
plot(forecast(x.ets,62), ylab="CPU usage",xlab="Day in 2012",xaxt="n",
yaxt="n",ylim=c(0,42000))
lines(seq(from=182,to=242,by=1),vcstatsresult$X12390,col="green")
axis(1,at=seq(60,245,5),seq(60,245,5))
axis(2,at=seq(0,42000,1000), seq(0,42000,1000))
```

Appendix F: VMware database design



st
- ☐ * (All Columns)
- ☐ TIME_ID
- ☑ SAMPLE_TIME
- ☑ SAMPLE_INTERVAL
- ☐ ROLLUP_COUNTER

hs
- ☐ * (All Columns)
- ☐ COUNTER_ID
- ☐ TIME_ID
- ☑ STAT_VAL

sc
- ☐ * (All Columns)
- ☑ COUNTER_ID
- ☐ ENTITY_ID
- ☐ STAT_ID
- ☐ DEVICE_ID
- ☐ STAT_LEVEL

si
- ☐ * (All Columns)
- ☐ **ID**
- ☑ MOID

sdt
- ☐ * (All Columns)
- ☐ **DEVICE_ID**
- ☐ **DEVICE_TYPE_ID**

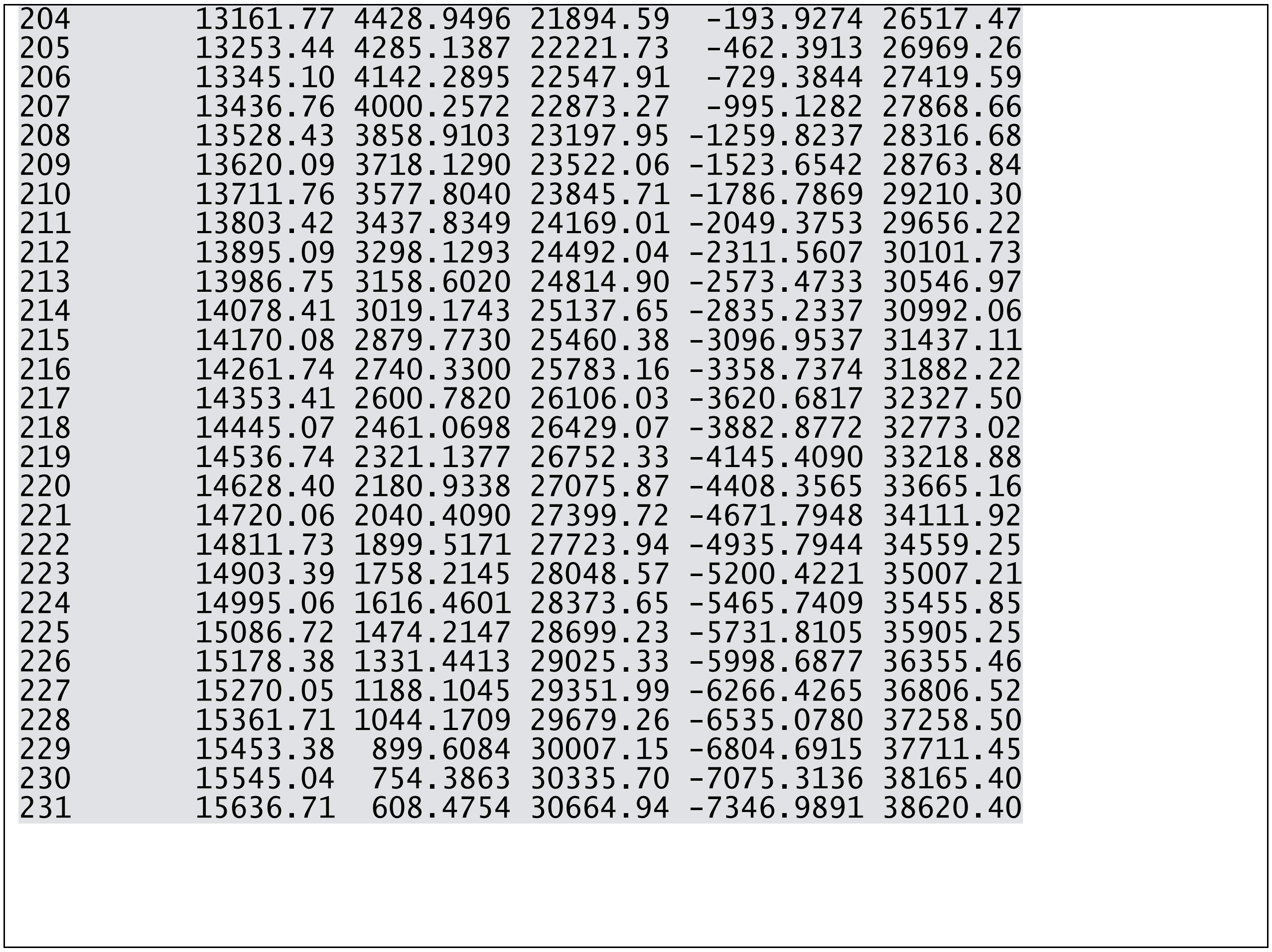
d
- ☐ * (All Columns)
- ☐ DEVICE_ID
- ☑ DEVICE_NAME

sdtd
- ☐ * (All Columns)
- ☐ **DEVICE_TYPE_ID**
- ☑ DEVICE_TYPE_NAME

sd
- ☐ * (All Columns)
- ☑ **ID**
- ☐ ROLLUP_TYPE
- ☑ NAME
- ☑ GROUP_NAME
- ☐ TYPE
- ☐ UNIT
- ☐ ASSOCIATE_IDS
- ☐ STAT_LEVEL
- ☐ PERDEVICE_STAT_LEVEL
- ☐ DEFAULT_LEVEL_SET
- ☐ FIXED_COLLECTION_INTERVAL

Appendix G: Results from the Forecasting experiment

```
> summary(forecast(x.ets,50))

Forecast method: ETS(M,A,N)

Model Information:
ETS(M,A,N)

Call:
 ets(y = x.ts)

  Smoothing parameters:
    alpha = 0.6421
    beta  = 1e-04

  Initial states:
    l = 1435.7091
    b = 91.9158

  sigma:  0.1647

     AIC     AICc      BIC
2297.389 2297.731 2308.605

In-sample error measures:
          ME          RMSE           MAE           MPE          MAPE
MASE
 -20.6035957 1357.4350391  961.7949186   -2.2600638   12.5060832
0.9579158

Forecasts:
    Point Forecast     Lo 80    Hi 80      Lo 95    Hi 95
182         11145.16 8792.4586 13497.86  7547.0138 14743.31
183         11236.82 8413.2637 14060.38  6918.5612 15555.09
184         11328.49 8091.7814 14565.20  6378.3723 16278.60
185         11420.15 7808.0046 15032.30  5895.8491 16944.46
186         11511.82 7551.0590 15472.57  5454.3606 17569.27
187         11603.48 7314.2692 15892.69  5043.6977 18163.26
188         11695.15 7093.1964 16297.09  4657.0720 18733.22
189         11786.81 6884.7171 16688.90  4289.7063 19283.91
190         11878.47 6686.5371 17070.41  3938.0921 19818.85
191         11970.14 6496.9149 17443.36  3599.5658 20340.71
192         12061.80 6314.4926 17809.11  3272.0508 20851.55
193         12153.47 6138.1880 18168.74  2953.8922 21353.04
194         12245.13 5967.1231 18523.14  2643.7468 21846.51
195         12336.79 5800.5737 18873.02  2340.5074 22333.08
196         12428.46 5637.9351 19218.98  2043.2491 22813.67
197         12520.12 5478.6961 19561.55  1751.1900 23289.06
198         12611.79 5322.4203 19901.15  1463.6626 23759.91
199         12703.45 5168.7313 20238.17  1180.0915 24226.81
200         12795.12 5017.3025 20572.93   899.9771 24690.25
201         12886.78 4867.8479 20905.71   622.8819 25150.68
202         12978.44 4720.1157 21236.77   348.4209 25608.47
203         13070.11 4573.8826 21566.33    76.2525 26063.96
```



```
204        13161.77 4428.9496 21894.59  -193.9274 26517.47
205        13253.44 4285.1387 22221.73  -462.3913 26969.26
206        13345.10 4142.2895 22547.91  -729.3844 27419.59
207        13436.76 4000.2572 22873.27  -995.1282 27868.66
208        13528.43 3858.9103 23197.95 -1259.8237 28316.68
209        13620.09 3718.1290 23522.06 -1523.6542 28763.84
210        13711.76 3577.8040 23845.71 -1786.7869 29210.30
211        13803.42 3437.8349 24169.01 -2049.3753 29656.22
212        13895.09 3298.1293 24492.04 -2311.5607 30101.73
213        13986.75 3158.6020 24814.90 -2573.4733 30546.97
214        14078.41 3019.1743 25137.65 -2835.2337 30992.06
215        14170.08 2879.7730 25460.38 -3096.9537 31437.11
216        14261.74 2740.3300 25783.16 -3358.7374 31882.22
217        14353.41 2600.7820 26106.03 -3620.6817 32327.50
218        14445.07 2461.0698 26429.07 -3882.8772 32773.02
219        14536.74 2321.1377 26752.33 -4145.4090 33218.88
220        14628.40 2180.9338 27075.87 -4408.3565 33665.16
221        14720.06 2040.4090 27399.72 -4671.7948 34111.92
222        14811.73 1899.5171 27723.94 -4935.7944 34559.25
223        14903.39 1758.2145 28048.57 -5200.4221 35007.21
224        14995.06 1616.4601 28373.65 -5465.7409 35455.85
225        15086.72 1474.2147 28699.23 -5731.8105 35905.25
226        15178.38 1331.4413 29025.33 -5998.6877 36355.46
227        15270.05 1188.1045 29351.99 -6266.4265 36806.52
228        15361.71 1044.1709 29679.26 -6535.0780 37258.50
229        15453.38  899.6084 30007.15 -6804.6915 37711.45
230        15545.04  754.3863 30335.70 -7075.3136 38165.40
231        15636.71  608.4754 30664.94 -7346.9891 38620.40
```

Appendix H: Expert review protocol

Interviewer	Kevin van Ingen
Interviewee	
Occupancy	
Date	28-09-2012
Place	Centric Gouda, the Netherlands

PREPARATIONS (IN ADVANCE)

- Notify the interviewee that the interview will be recorded
- Two copies of the thesis, together with a summary with main findings (graphs, tables)
- Supporting materials (laptop, paper, pen, whiteboard/flip over, marker)
- Provide the interviewee with an interview outline in advance

INSTRUCTIONS DURING THE INTERVIEW

Welcome the interviewee and prepare the outline of the interview. The transcribed text and questions can be found below.

1 Introduction (2-5 minutes)

2 Introduction of the research (rationale, progress until now) (10 minutes)

3 The expert review (20-40 minutes)

  a) Discussion of the input of the forecasting solutions

  b) Discussion of the numeric output of the forecasting solution

  c) Discussion of the practical application of the forecasting solution

4 Summarize findings (3 minutes)

5 Thank the interviewee and after-care

INTRODUCTION

"My name is Kevin van Ingen, I am a student at Utrecht University, and currently doing my Master's Thesis at Centric IT Solutions on the subject of *cloud cost forecasting.* I have been doing so in the past six months. In this final stage of the research project, you are elected to contribute to this research as a means of an expert review which is used as a validation on the research results."

INTRODUCTION TO THE RESEARCH

"Cloud technology is being adopted by both international industry and Centric as well. The transition to pay-per-use costing entails uncertainty about future costs. The objective of this project is to research the possibilities of a forecasting solution, that would counter that uncertainty by providing a *cost estimation,* and create a technical solution to do so. In this research the costs incurred in the operation of cloud technology is researched. This led to a classification of cost variables, for which pay-per-use applies. Different forecasting techniques are reviewed and compared. An extrapolation forecasting technique that allows for trend and seasonal effects is selected because of its broad applicability. The data where the forecast is based on is derived from an actual production environment on which business applications are run. Composite statistics about four cost variables are created that describe the usage of 52 virtual machines. Experiments are created to validate the selection of the proper forecasting model. Furthermore experiments are performed with varying forecasting horizons and varying datasets.”

The expert review

The following questions should be asked by the interviewer. Answers should be written down during the interview.

a) Discussion of the input of the forecasting solutions
   1. Do you think that the eight cost variables cover the majority of the operational costs?
   2. What do you think of application of VMWare statistics as a source for forecasting?
   3. What do you think of the decision to forecast on (cpu, memory, network, storage)?

b) Discussion of the numeric output of the forecasting solution
   1) Are the results of the different forecasts (with varying horizon) accurate enough?
   2) Does the output of the forecast described in this thesis provide enough information?

c) Discussion of the practical application of the forecasting solution
   1) Do you think the proposed forecasting solution has potential for practical application?
   2) Do you think that a forecast based on quantitative analysis has the potential to achieve a cost forecast in the near future? And what would make this happen?

Summarize findings

Summarize the findings of the expert and make sure there is consensus about them.

Thank the interviewee and after-care

Thank the interviewee. Also mention that after the interview, transcriptions will be e-mailed to the interviewee as a means of validation, and for privacy purposes. The interviewee should be offered the opportunity to correct any mistakes in the transcriptions.