Towards An Information Technology Infrastructure Cost Model

By

Ken Huang

Submitted to the System Design & Management Program
In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2007

Signature of Author
Ken Huang
System Design and Management Program
February 2007

Certified By
Dr. Ricardo Valerdi
Thesis Supervisor
Research Associate, Lean Aerospace Initiative

Certified By
Patrick Hale
Director
System Design and Management Program

Towards An Information Technology Infrastructure Cost Model

By

Ken Huang

Submitted to the System Design & Management Program

On Jan 31st, 2007 in Partial Fulfillment of the

Requirements for the Degree of Master of Science in

Engineering and Management

ABSTRACT

Ever since the introduction of the Internet in 1994, one of the defining characteristics of the global economy, particularly in the US, is a dramatic increase in expenditures on Information Technology. While this trend is expected to continue, a major issue for companies of all sizes is the manner in which precise forecasting of future IT cost may be undertaken. The present thesis investigates the possibility that a set of the essential deterministic cost drivers with varying weighted factors may prove capable of estimating total IT infrastructure costs. An online questionnaire was developed for this purpose, and was used to survey senior IT leadership teams. The data collected from this survey was then computed with Analytical Hierarchy Process (AHP) to illustrate the relative importance of different cost drivers.

The study revealed three primary findings. First, that a set of essential deterministic cost drivers with varying weighted factors could be used as a general tool for estimating the total cost of IT infrastructure. Second, these different sectors prioritize cost drivers differently from each other. In the Financial Services sector, for instance, the security of the IT network was reported to be of greater importance than the service call response time. In the Technology sector, however, the opposite was true. Third, numerous correlations were found to exist within each cost driver category defined. The correlated nature of these cost parameters may mean that a more parsimonious model may be more predictive of total IT infrastructure costs.

It is hoped that these findings may be of benefit to a variety of large and small commercial and government entities, which may be able to use the predictive cost drivers to help eliminate problems related to inaccurate IT cost estimates. It is believed that the cost model proposed may be applicable across a variety of economic sectors. In this thesis, its applicability is demonstrated within the

financial services and technology sectors. Future research may be useful in evaluating the model further, by increasing the sample size, and by testing the reliability and validity of the cost model within additional economic sectors.

Thesis Supervisor: Dr. Ricardo Valerdi

Title: Research Associate

Lean Aerospace Initiative

ACKNOWLEDGMENTS

My ten years of experience in the IT workforce has made me highly sensitive to the increased spending on IT infrastructure in recent years. I am fortunate to have had numerous inspirational colleagues who have undergone the pain of projecting annual IT budgets with me and who have provided me with useful insight throughout the years.

In early October, 2005, Dr. Ricardo Valerdi sent out an email seeking an IT Cost Model researcher from within the MIT community. I was so thrilled and applied for it without any hesitation. I was very fortunate that he has accepted me. Ever since then, I have greatly enjoyed researching for Dr. Valerdi. Not only has he provided constant support to developing this thesis, but his dedication, professionalism, and flexibility to accommodate my hectic has deeply impressed and influenced me. I would like to express my highest gratitude to him, for his huge commitment of time and effort, which has gone far beyond the scope of this thesis.

I would also like to thank the following people, who have been highly in my life, and who willingly gave me their time and consideration for the purposes of filling out the cost model survey used within: Mr. ReiJane Huai, Dr. Narayanan Krishnakumar, Mr. Michael Mimo, Mr. Ziad Ghafour, Mr. Thomas Welch, Mr. Brian Garrity and Mr. Goutam Ghosh. My thesis could not be complete without their invaluable input. Indeed, their diverse views, representing various industries, have increased the scope of this thesis, and have provided it with a multi-faceted perspective. I am further grateful to my previous employers, Thomas Welch, Ziad Ghafour and Michael Mimo, who each fought stuck by my side and petitioned for me special time-off so that I could complete my SDM curriculum. Without Director Kevin Murphy's final approval and ongoing support, my study at MIT could not have come to fruition.

Certainly I would also like to mention my most significant other, Joyce Lii, my aunt, Liang-Jun Chen, and my mother, Mei-Yun Chen. Their love and spiritual support are truly unconditional and unlimited. Their encouragement assisted me to conquer any and all obstacles that stood before me.

Lastly, but certainly not least, many thanks go to the SDM Director Pat Hale for his constant help and timely guidance. His views can be found throughout my thesis and will no doubt continue to color my professional views to come.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

THESIS OUTLINE

Chapter 1: *Introduction & Motivation.*

Chapter 2: *Currently Utilized Cost Models.* This section contains a detailed technical description of an existing cost model from which we draw our approach and a leading commercial IT cost model to which we compare our approach.

Chapter 3: *The Present Cost Model.* This section outlines the cost model utilized in the present thesis, and also provides examples of the online survey used to collect the data use to create the cost models.

Chapter 4: *Size and Cost Driver Definitions.* This section describes the logical reasoning behind this survey, and behind the extensive list of cost drivers devised for inclusion in the survey. In addition, a brief illustration of the techniques used to automate the survey data analysis are discussed, along with the manner in which a custom SAS program was used to compute the AHP results for IT cost analysis.

Chapter 5: *A Brief on The Statistical Techniques Employed.* This section describes the statistical methods and tools used to analyze the data. In particular, a brief overview of the SAS programming language will be provided, along with a description of the primary statistical techniques employed within the thesis, primarily the student's t-test and the Pearson correlation. A brief description of the Analytical Hierarchy Process (AHP) method follows, along with a discussion of how this method may prove useful for estimating IT-related costs. A simple AHP example is provided for illustrative purpose.

Chapter 6: *Analysis, Results & Discussion.* This section describes the results of the study, and provides detailed consideration of the implications of these findings. Strengths and weaknesses of the approach utilized within the thesis are discussed, in turn.

Chapter 7: *Conclusions and Future Work.* Building off of the strengths and weaknesses of the present thesis, Chapter seven identifies the primary contributions of the thesis, and identifies valuable avenues for future research.

CHAPTER 1: INTRODUCTION & MOTIVATION

Throughout my ten years as an IT professional, I cannot estimate how often I have witnessed companies underestimate the costs related to their IT infrastructure needs. Indeed, such estimations can be a daunting task. One must take into account the current and future needs of the company, must make decisions regarding how scalable and modular the infrastructure should be, must consider the costs of annual maintenance, and must also factor in costs associated with the IT labor force. Unfortunately, there may be a method, but it is probably very informal and non-repeatable for making these IT-related determinations, or for estimating the associated costs. And thus it is not surprising that companies continue to repeat their past mistakes, and fail to fully consider the costs associated with supporting a valid IT infrastructure. With technological advances and IT costs continuing to rise at an alarming rate, underestimating these costs may lead to dire consequences. This is particular so for businesses in e-commerce intensive fields, where the costs associated with IT infrastructure can be particularly high.

The present thesis has been designed as a means to explore the possibility that a set of essential deterministic cost drivers with varying weighted factors can be arrived at that are capable of predicting the total cost of a company's IT infrastructure needs. In particular, it is believed that such a set of cost drivers may prove a valuable tool for the estimation of IT-related total cost of ownership (TCO). TCO is a financial estimate designed to help consumers and enterprise managers assess all costs associated in the lifecycle of any capital investment, particularly in hardware or software, from acquisition to disposal. In this thesis, the possibility that a limited set of cost drivers can serve to predict relevant IT-related TCO will be undertaken.

Within the following sections, relevant IT market data that triggered the development of this thesis will be outlined, and the potential value of the cost model proposed above will be highlighted. First, IT market size and segmentation will be discussed, followed by data demonstrating some of the common mistakes that lead to the misestimating of IT-related costs.

IT Market Research

According to a report by InfoTech Industry Trends: "The worldwide market for InfoTech products and services was estimated at $2.1 trillion in 2005" (Plunkett Research Ltd., 2006). Indeed, as of 2005, $416 billion was spent in the U.S. on information technology, and $61 billion was spent on related networking equipment. The costs associated with the building and maintaining of effective IT infrastructure is, thus, vast. Table 1, which serves as a reconstructed table from Plunkett's Info Tech Industry Almanac (Plunkett Research, Ltd., 2006), displays the global IT industry overview for 2006. The staggering size of these numbers indicates the magnitude of the costs savings that could be reaped by even a small decrease in required IT costs. Furthermore, as demonstrated by Table 2, (re-depicted from *Segmenting the Business Market – There's More to it Than Size*, J. Jernigan 2005), there appears to be a sharp rising trend related to IT cost expenditures. According to In-Stat (2005), the data concurs with the observation from Figure 1. Based on a sample size of 1007 companies, the total mean increase in IT-related expenditures is 1.8 percent annually. While these statistics varied somewhat depending on the specific sector that a given business was categorized in, these statistics demonstrate the consistently increasing trend of IT-related spending.

Figure 2 illustrates some interesting IT utilization patterns. First, 13% of respondents reported that they consider IT to be a core aspect of their business enterprise. Companies that responded in this fashion tend to be largely IT driven, and thus believe that they needed to keep their IT infrastructure particularly up to date. An additional 34% of respondents regarded IT as an essential investment area, even if not absolutely core to their business practices. These companies

Table 1. A reconstructed table of IT industry overview. (Plunkett Research, Ltd. 2006)

Legend	
PRE = Plunkett Research estimate	**In-Stat** = In-Stat Market Research
DC = Department of Commerce	**IDC** = International Data Corp.

InfoTech Industry Overview				
	Amount	Unit	Date	Source
Global Investment in Information Technology (est.)	$2.10	tril. US$	2005	**IDC**
U.S. Spending on Information Technology	$416	bil. US$	2005	**IDC**
Worldwide PC Shipments	207.7	mil. Units	2005	**IDC**
Worldwide Server Revenues	$53.70	bil. US$	2005	**IDC**
Worldwide Storage Revenues	$27.30	bil. US$	2005	**IDC**
Worldwide Router Sales	$7.20	bil. US$	2005	**In-Stat**
Spending on Network Equipment, U.S. (incl. telecom)	$61.00	bil. US$	2005	**IDC**
Worldwide Software Revenue	$200.00	bil. US$	2005	**PRE**
Worldwide Security Software & Services Spending	$45.00	bil. US$	2005	**IDC**
Worldwide Database Software Market	$12.50	bil. US$	2005	**PRE**
Worldwide Video Game Industry Revenue	$27.00	bil. US$	2005	**PRE**
Worldwide RFID Tag Revenues	$504	mil. US$	2005	**In-Stat**
Number of Cellular Phone Subscribers, U.S.	190	mil.	2005	**PRE**
Total North American Cable Modem Subscribers	25	mil.	2005	**PRE**
Est. Number of VOIP Subscribers, U.S.	5	mil.	2005	**PRE**
Value of Computer Hardware Exports	17	bil. US$	J-05	**DC**
Value of Computer Hardware Imports	37	bil. US$	J-05	**DC**
Number of High Speed Internet Lines in the U.S.	37	mil.	2005	**PRE**

U.S. Industry Revenues:				
Data Processing Services 61.0 bil.	61	bil. US$	2005	**PRE**
Software Publishing 116.0 bil.	116	bil. US$	2005	**PRE**
Wired Telecommunications Carriers 200.0 bil.	200	bil. US$	2005	**PRE**
Wireless Telecom. Carriers (Except Satellite)	135	bil. US$	2005	**PRE**
Semiconductor Industry Revenues	$235	bil. US$	2005	**Gartner**

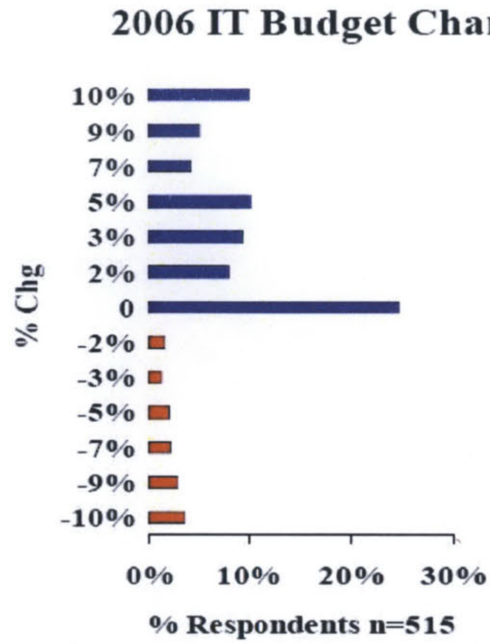


Figure 1 – 2006 IT Budget Change. Abstracted from In-Stat 12/05.

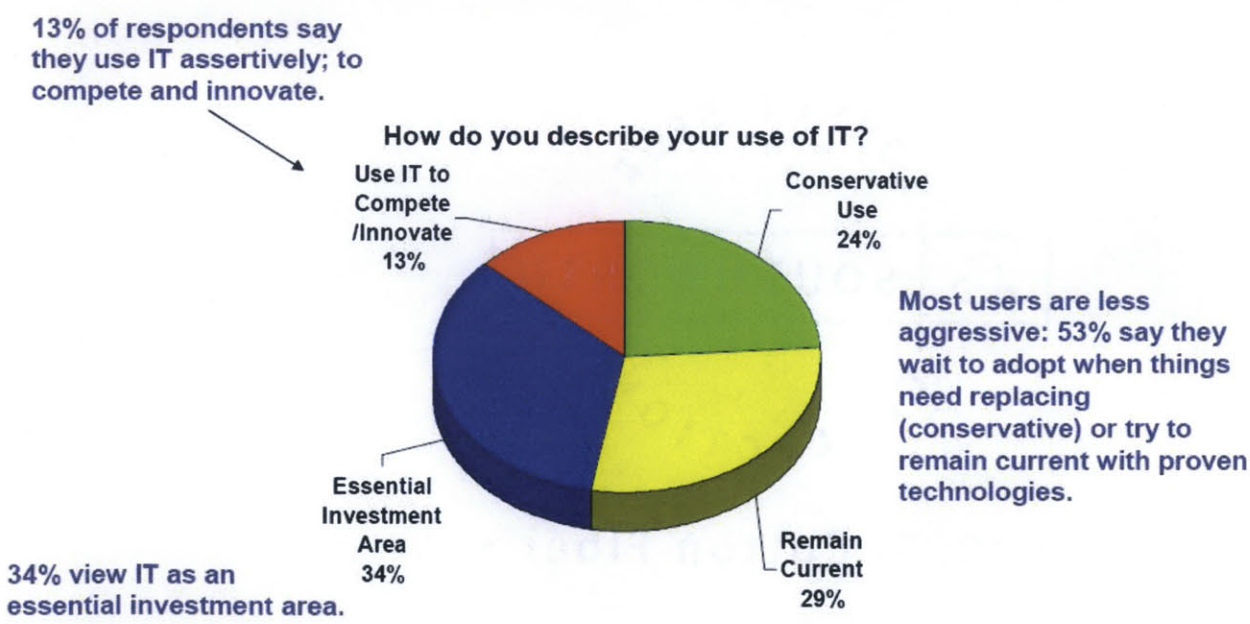
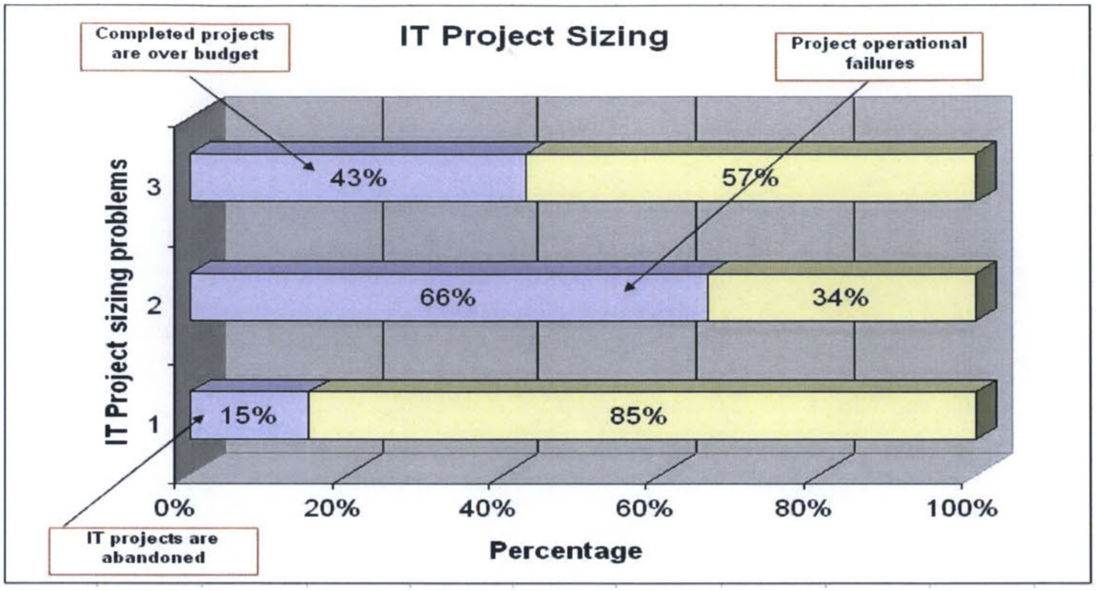


Figure 2 – IT utilization pie chart. Abstracted from In-Stat 12/05.

tend to be small to medium in size, and believed that they required a stable and scalable IT infrastructure to maintain their competitive advantages in the workplace. Surprisingly, 53% of respondents suggested that they were less aggressively investigating their IT needs. Companies that reported this perspective tended to be larger companies, however, that already had a stable and highly complex IT infrastructure. These companies were more concerned with the maintenance of their existing IT infrastructure, rather than upgrading further. Indeed, simply maintaining large IT infrastructures can be a daunting task, let alone undertaking a multi-year upgrading project requiring tremendous time, costs and human resources.

Unfortunately, many companies, both big and small, appear unable to accurately and efficiently estimate their required IT needs. Often this may cause a company to grossly over-estimate their required IT needs, leading to over-spending and higher maintenance costs. Other times a company may under-estimate their needs, leading to higher long-term costs as the inadequacies of the current system become realized and corrected.

Figure 3 depicts data from the Standish Group's Chaos Report, and illustrates some of the IT project sizing problems that exist today. This figure highlights three of the major IT project sizing problems that plague the industry today. Specifically, 43% of the projects reported to be over budget, 66% of them had some type of operational failure, and 15% of them were abandoned. Each of these problems can be seen as leading to a preventable ballooning of IT-related costs. Particularly troublesome is the fact that IT infrastructure may have been purchased in advance for many of these uncompleted projects. This is particularly common within large companies with complex IT projects, as a separation between the IT workforce and upper-management often leads to an unfortunate disconnect regarding needs and resources. Unfortunately, the costs associated with these increase IT expenditures are unlikely to be able to be recouped, and will inevitably effect



Legend			
IT project sizing problems	1	2	3
Problem scenarios	Completed projects are over-budget	Operational failures	IT projects are abandoned
Percentage	43%	66%	15%

Figure 3 - IT Project Sizing problems

the company's bottom line. These issues could have been prevented or mitigated if there would have been a reliable method for estimating the cost of IT projects before resources were committed.

Motivation

My years of experience within the IT workforce have led me to recognize the common pitfalls that those required to estimate IT-related costs can fall into. In particular, three observations have come to serve as motivating forces for the development of this thesis: First, the costs associated with IT spending, while massive, are also inevitable. Indeed, based on Figure 2, 47% of the companies survey – nearly half the marketplace - remain highly aggressive in developing increased IT-related infrastructure today. Thus, rather than trying to reduce IT needs, it makes more sense to try to develop increasingly accurate models for determining actual and realistic requirements. Second, misestimating IT needs remains a difficult task fraught with errors. Even for experienced CIOs, misestimation of IT requirements remains commonplace and global. It is, thus, of considerable urgency to develop precise IT cost models – or to develop methods of developing such models, at least – that can help organizations allocate their annual IT budgets more accurately and intelligently. Given the size of company's IT-related budgets, even small improvements in prediction could save millions of dollars in end of year net profit margins. Third, there currently exists a dearth of methods available to effectively determine IT-related expenditure needs. The leading model, True IT, will be outlined within this thesis, however neither it has yet proven itself effective across a broad array of economic sectors, nor has it amassed the body of evidence required to give CIOs in the business marketplace sufficient confidence to utilize them regularly. It is, indeed, this dearth of appropriate cost models that serves as the primary motivation for the present thesis. As described in the more detail within the following sections, it is the goal of the present thesis to test the proposition that a limited number of cost drivers, with variable weighted derivatives, can be developed that provide maximal predictive ability across all economic sectors. It is believed that the development of such a cost model could be of great value to the business world.

Research Goals

In this spirit, this thesis focuses on five principal research goals to aid in the development of an IT cost model. These goals include:

1. To identify the most common current IT cost estimation approaches.

2. To propose a new but intelligent IT cost estimation approach, in terms of size drivers, cost drivers, and miscellaneous drivers and categories.

3. To collect data from real business units within the technology and financial services sectors, for determination of appropriate weighted derivatives to attach to the relevant cost drivers.

4. To apply complex statistical tools to the developed drivers in order to investigate the relationships between each of these drivers, and their relevance within the context of active business enterprises.

It is understood that the data collected within this thesis can serve only as a first attempt to validate the cost model developed. Indeed, significant work will be required in evaluating the extent to which this model remains relevant within sectors outside the technology and financial services sectors. With this knowledge well in mind, the specific academic contributions from this study are to identify those drivers that appear most relevant, and to provide an initial test of the applicability of these drivers within two fairly diverse sectors. In the process, we will identify any limitations of this approach and note possible threats to the validity of the model (how applicable this approach is to different fields against historical data, time frame, internal validity; do we cover every input variables, etc.). In so doing, we will develop a growing database of knowledge regarding the characteristics necessary to develop a comprehensive cost model capable of estimating IT-related expenditures.

Formal Proposal and Brief Methodology

It is proposed that there is a limited set of cost drivers that are capable of accurately predicting IT-related costs. To this end, the present thesis serves to identify those cost drivers that are maximally predictive of IT-related costs, towards the development of a comprehensive IT cost model. This effort has been undertaken with the expectation that such a cost model may be applicable across a wide range of industries, and may thus be of substantial use for CIOs as they attempt to determine and budget for their future IT needs. It is based on the recognition of similar cost model applications with more limited applicability: uniquely within the software sector, for instance. Given the success of these more limited cost model applications, it seemed the appropriate time to begin investigation of broader, more inclusive, models that could be relevant across a wide range of industry sectors.

In order to develop this list of cost drivers, CIOs within both the financial services and technology sectors were surveyed as to those drivers they believed were most imperative to the future of their business. These two sectors were chosen because of their extensive dependence on IT infrastructure for the nature of their business. The CIOs rankings were then tested with a variety of statistical techniques to identify similarities and differences across the two sectors, and to work towards the development of a complete and comprehensive IT cost model.

CHAPTER 2: CURRENTLY UTILIZED COST MODELS

This section contains a detailed technical description of an existing cost model from which we draw our approach and a leading commercial IT cost model to which we compare our approach.

COSYSMO

COSYSMO, created by Dr. Ricardo Valerdi (Valerdi, 2005) computes precise cost estimates of systems engineering effort for hardware and software systems. The model has proven particularly powerful in the aerospace and defense sectors, and contains a vast dataset of IT-related cost needs, including those for companies such as Raytheon, Northrop Grumman, SAIC, General Dynamics, BAE Systems and Lockheed Martin.

The model employed within the present thesis builds substantially from the COSYSMO's approach, both in terms of the organization of the survey structure and the assignment of ratings to specific size and cost drivers. Specifically, the present IT cost model survey utilizes a similar layout to that of COSYSMO, and shares several of the size and cost driver parameters. The model has expanded the COSYSMO survey in several manners, however, and has utilized a finer scale that is expected to provide increased accuracy of IT-cost determinations. In the following section, the COSYSMO algorithm is considered in greater detail.

The COSYSMO Algorithm

The COSYSMO algorithm is as follows

$$PM_{NS} = A \cdot \left(\sum_k (w_{e,k}\Phi_{e,k} + w_{n,k}\Phi_{n,k} + w_{d,k}\Phi_{d,k}) \right)^E \cdot \prod_{j=1}^{14} EM_j$$

Where:

$\mathbf{PM_{NS}}$ = effort in Person Months (Nominal Schedule)

A = calibration constant derived from historical project data

k = {REQ, IF, ALG, SCN}

$\mathbf{w_x}$ = weight for "easy", "nominal", or "difficult" size driver

$\mathbf{\Phi}_x$ = quantity of "k" size driver

E = represents diseconomies of scale

EM = effort multiplier for the j_{th} cost driver. The geometric product results in an overall effort adjustment factor to the nominal effort.

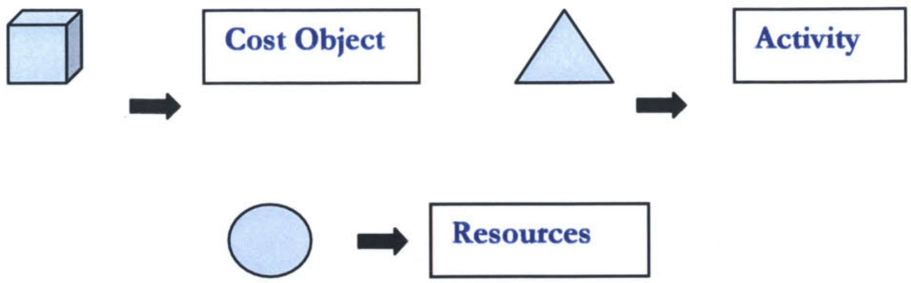
PRICE SYSTEMS TRUE IT

PRICE Systems[1] is one of the leading commercial software companies that specialize in cost research and consulting regarding the affordability of IT- and software-intensive projects. TRUE IT, part of the PRICE TruePlanning software package, exists as an Information Technology Estimating Model application. According to PRICE Systems, by using TRUE IT, companies can expect 30% return on investment (ROI) and yet put 80% less effort into their IT initiatives.

TRUE IT Overview

TRUE IT utilizes PRICE Systems' proprietary cost estimating framework. This framework supplies a common user interface, and common utilities and features for estimating single and multiple models. The taxonomy of TRUE IT and its corresponding icon symbols are as follows:

[1] PRICE Systems official website (http://www.pricesystems.com/index.asp)



The definition of each TRUE IT object is listed below:

Cost Object: Cost objects are fairly straightforward: Any equipment required for the building or maintenance of specific IT-infrastructure may be expected to have a cost associated to it. Indeed, in the TRUE IT software package, cost models are driven by inputs that produce cost/schedule/risk outputs. Examples of cost objects may include servers, workstations, network devices, and application software.

Activity Object: Activities are any action, task or project of interest that may have an associated cost. Examples of IT-related activities relevant to ongoing business concerns may include the installation of new hardware, the upgrading of existing hardware, the design of new software, regular maintenance, beta testing, and ongoing quality control testing. As can be seen from this list, activities can exist as one-time activities, or as longer-term ongoing concerns.

Resources Object: All activities can be expected to require certain resources for their implementation. For instance, the installation of new hardware may require IT-support time, Help Desk Operators, Software or Hardware Specialists and so on. Thus, resources exist as sources for the support and execution of a given Activity. Often-times these resources refer to human labor, however, this is not necessarily so.

All TRUE IT objects are grouped in the logical tree hierarchy shown in Figure 4.

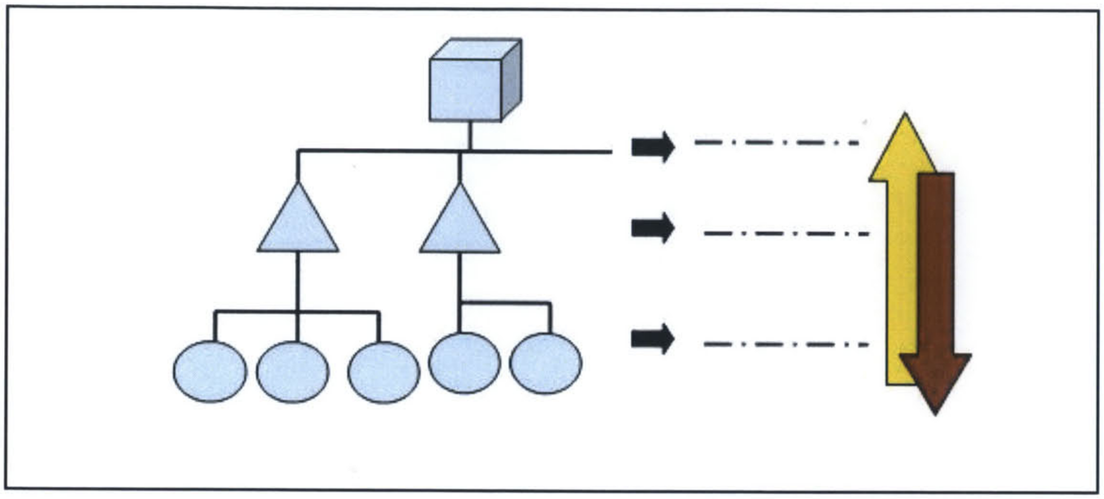


Figure 4 – TRUE IT logical grouping view

As can be seen within this hierarchical model, cost objects and activity objects can be construed as existing as a function of the other two object-classes. Cost objects, for example, exist as a function of the specific activity objects included in the model, and the resource objects that may be required to implement those activities. Activity objects may similar be construed as a function of the resource objects required for implementation of that activity object, and the cost objects that corresponds to those resource objects. Resource objects are construed somewhat differently in the TRUE IT model. Resource objects are not construed as functions of cost objects and activity objects. Rather, resource objects must be determined, based on the expected support requirements of the desired activity objects.

The value of the TRUE IT framework is quite obvious. Indeed, the model is capable of: a) forecasting real workloads and estimating the resources required for the successfully execution of IT projects, b) evaluating IT investment portfolios from a continuous life cycle perspective, and c) rapid modeling and evaluation of complex IT project scenarios. As can be seen, however, decisions must be made at each level of the hierarchy. At the activity object level, the desired changes to the

IT infrastructure must be determined. At the cost object level, the specific hardware/software requirements for implementing these desired changes must be arrived at. Finally, at the resource object level, the manpower required to bring these changes to fruition must be estimated. Thus, while helpful in breaking down the steps required to implement infrastructure changes, the model leaves open the opportunity for error at all three levels within the hierarchy. And because of the interdependency of the levels within the hierarchy, errors or miscalculations at any one level may multiply the estimation discrepancy of the final model

CHAPTER 3: THE PRESENT COST MODEL

The following chapter outlines the cost model utilized in the present thesis, and also provides examples of the online survey used to collect the data use to create the cost models.

The model employed within the present thesis builds substantially from the approach employed within the COSYSMO model. First, the organization of the survey structure is similar. While this may appear a superficial similarity, the manner in which the data are collected may have important influences on the quality and type of data obtained. In this case, the present model and the COSYSMO model share the strategy of asking practitioners to complete an online survey ranking the importance of various cost drivers. Thus, the data can be guaranteed to come from relevant and trustworthy sources, and from those individuals most likely to actually make the IT decisions for their company.

In addition, the current cost model shares several cost drivers with the COSYSMO model, and also uses a method of assigning ratings to specific size and cost drivers that are similar to that employed by COSYSMO. The overlapping cost drivers are considered a strength of the current model. The drivers utilized within the COSYSMO model have already undergone substantial validation, and are well accepted to provide accurate estimation of IT-related costs. The current model thus builds off of the strengths of the COSYSMO model, and adds additional cost drivers in an attempt to add additional predictive validity to the model.

Several differences exist as well, however. First, as already stated, the list of cost drivers in the current model is larger than that in the COSYSMO model. It is the goal of the present thesis to determine whether any of these additional drivers add incrementally to the validity of the total cost model. In addition, the current model has utilized a finer scale that is expected to provide increased accuracy of IT-cost determinations.

Comparison of the Current Model and the TRUE IT Model

A primary difference between the cost model utilized in the present study and the TRUE IT application is the size of the list of cost drivers associated with both models. As will be seen below, the currently proposed model includes a broad list of cost drivers, while the TRUE IT system employs a significantly less extensive list. This does not necessarily indicate the superiority of one model over the other, as the sheer number of cost drivers is not the most relevant criteria for success. Rather, the ability of the model to predict IT infrastructure needs should be considered the most appropriate mode of evaluation. The present thesis did not perform a direct comparison of the two models, and so claims of superiority cannot be made. It can be noted, however, that one of the primary reasons for the larger driver size in the current model is the early stage of testing that this model is in. It was the purpose of this thesis to develop a broad list of drivers, that could be whittled down to those drivers that offered the maximal prediction of IT infrastructure needs. As will be seen in the results and analysis section, several cost drivers were rated as highly important, while other drivers were rated less so. Future iterations of the current cost model may, then, become increasingly narrow as only those drivers deemed most relevant will remain included.

It should also be noted that the source of the data used to support each model also differs. The current survey data was gathered from a group of senior IT leadership members in large reputable organizations. The TRUE IT system's algorithm was, instead, based on the industrial market data provided by other commercial marketing research companies. It may be argued that an advantage of the former technique is the directness of the approach, wherein the data could be collected directly from the CIO of the company. A necessary downfall of this approach, however, is that the sample size will be necessarily smaller. Indeed, while the TRUE IT algorithm is based on data from hundreds of companies, the current model was designed based on the input from only seven CIOs.

The final difference between the two models relates to the nature of the cost drivers included in the model. Whereas the TRUE IT system utilizes a standard weighting on their cost drivers, the current model explores the possibility that a variable weighted factor may serve to make the model increasingly flexible across different industry sectors. Indeed, as will be seen, the weightings identified within the financial services and technology sectors varied greatly, indicating that a standard weighting would have been less likely to have provided as good a fit. It must, at this point, be noted once again, however, that the two models were not directly compared, and thus it cannot be claimed with complete confidence that the variable weightings improved predictive ability.

The Online MIT IT Cost Model Survey

The MIT IT Cost Model Survey is a very convenient, web-based survey created especially for this thesis. The survey can be found at: http://oursdm.com/mit/. Figure 5 shows a screenshot of the survey webpage. The purpose of the survey was to gather all the data provided by those senior IT leadership teams (from financial and technology sectors) who were primarily responsible of the IT cost for their companies. Statistical analyses were then conducted on the data and a cost estimating relationships relevant to IT infrastructure cost was developed.

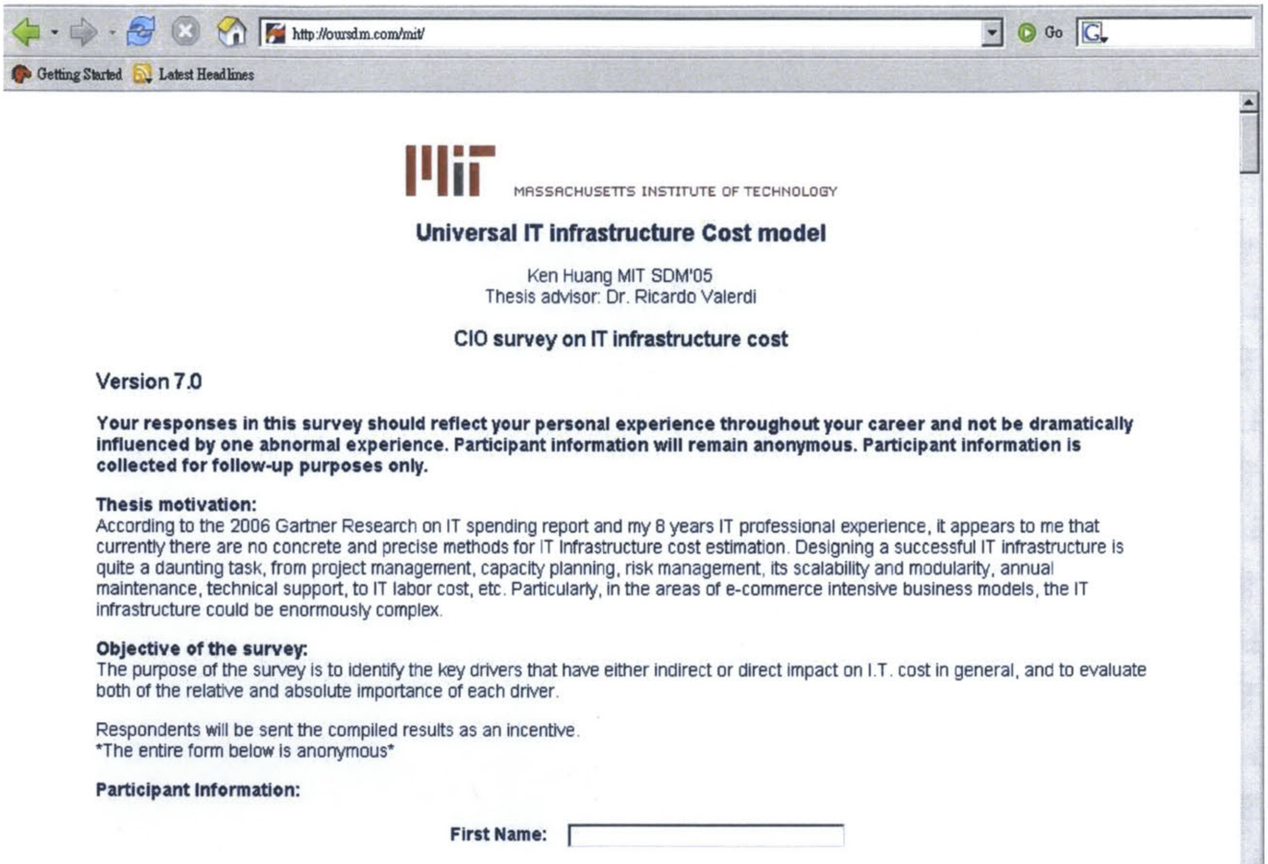


Figure 5 – A screenshot of the online MIT CIO survey

Collecting the survey results and computing the AHP results

Filling out the survey should take approximately 30 minutes. Clicking on the "Submit Form" button completes the survey submission electronically. See Figures 5 and 6 for complete directions on the survey submission process, as well as the successful submission confirmation screen.

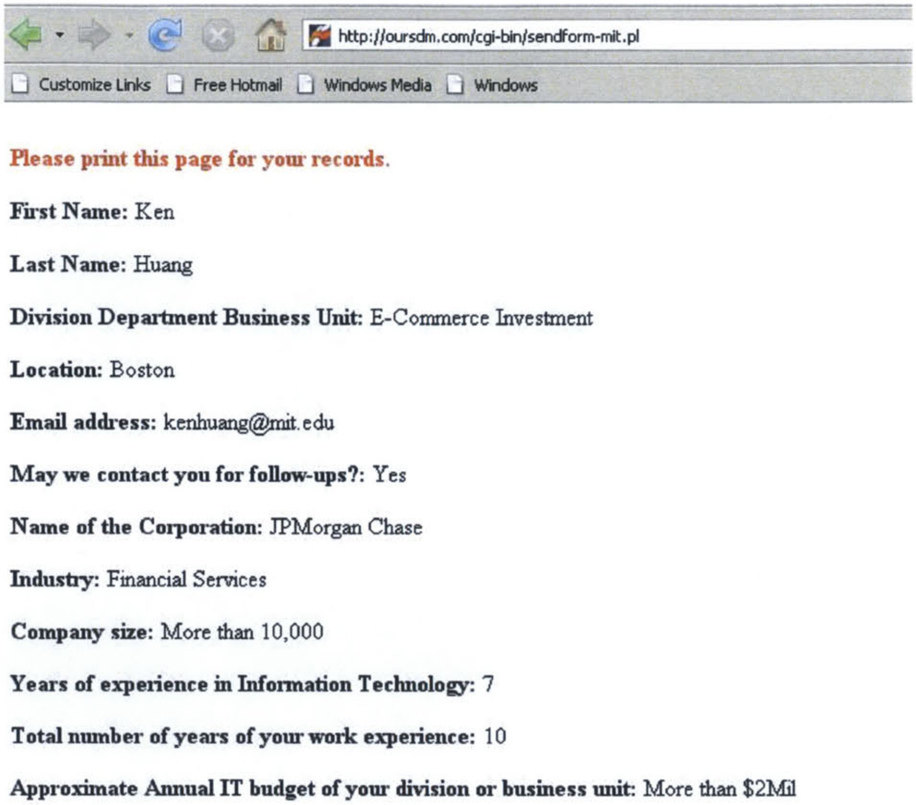


Figure 6 – A screenshot of successful survey submission.

The survey data is stored electronically on a hosting Linux server, in a flat file format that can be imported directly into SAS. The data dictionary in Appendix A defines all the survey input variable names required for referencing the data in SAS. Appendix B shows the actual SAS program used to import the survey data, and to compute the AHP results. These AHP results can, in turn, be exported into Word, Excel or pdf format for convenient transfer and modifiability.

CHAPTER 4: SIZE AND COST DRIVER DEFINITIONS

This section describes the logical reasoning behind this survey, and behind the extensive list of cost drivers devised for inclusion in the survey. In addition, a brief illustration of the techniques used to automate the survey data analysis are discussed, along with the manner in which a custom SAS program was used to compute the AHP results for IT cost analysis.

Designing the CIO IT Cost Model Survey

The overall layout of the CIO IT Cost Model survey was based on the COSYSMO cost model survey. The survey consists of three main sections:

1. **Background information of a survey participant** -> this section gathers demographic and work-related data concerning the participant, including, for instance, their work experience, the size of their company, and their annual IT cost budget.

2. **Size drivers** -> size drivers are size parameters that are tangible and quantifiable; it can be expressed in discrete numeric format. Below is a list of all the size drivers utilized within the cost model analysis utilized within this thesis (see Table 2).

3. **Cost drivers** -> cost drivers are cost parameters that in most cases are intangible but have qualitative impact on constructing the IT cost model. Below are the descriptions of four categories of cost drivers, with explanations of the inclusion of each driver category (see Tables 3 through 6, and relevant discussions).

Table 2. Types of size drivers

Types of size drivers
Number of servers or racks
Number of software licenses
Number of KVMs[2] (terminal servers)
Number of distinct sites(data centers)
Number of users / PC equipments
Number of feet of FA-CL or CAT5 cables.
Number of software applications that need to be supported (COTS and new)
Number of databases (i.e., human resources, sales, etc)
Number of data & phone jacks

[2] KVM is short for keyboard, video and mouse. It is a hardware switch device that enables a single keyboard, video monitor and mouse to control multiple computers at a time.

Table 3. Systems Cost Drivers

Service call response time	The time required or agreed to respond a technical support ticket opened by the customer.
Reimplementation / re-design	To re-architecture / to enhance some/entire functionalities of the systems in question.
Client/server compatibility	The handshakes or cohesiveness of communications between its clients and the server. Would be there any hiccups at the network communication level.
Security	The company compliance level of the "systems" in question.
Server Redundancy	A hot-standby (disaster recovery) server for the primary server of the same functionality.
Business Continuity	High Availability of the "systems" infrastructure overall.
MTTR (mean time to recovery	The average amount of time required to resolve most hardware or software problems with a given device.
TCO (total cost of ownership)	Cost to purchase and maintain software over time.
SLA (Service level agreement)	Formal agreement between a Service Provider and customers to provide a certain level of service. Penalty clauses might apply if the SLA is not met.

Each of these cost drivers were devised based on their relevance to the determination of IT-system needs. The relevance of certain drivers are quite obvious: the TCO of a given system, for instance, is likely to be a great interest to most purchasers of IT systems. Other drivers may be less obvious, but not necessarily less important. A CIO may, for instance, neglect to take into account the fact that the service response time of the company associated with a given system may be particularly slow, and may further neglect the impact on his company's bottom-line that delays in service calls may cause. As a second example, some companies may feel that server redundancy is just an unnecessary cost, while other companies may feel that having all of their data doubly, triply, and even quadruply backed up is integral for their ongoing business concerns.

Table 4. Hardware Cost Drivers

Seamless integration	The smoothness of the coordination between two or more hardware components.
Component volatility	The rate of stability of the component.
Component application complexity	The level of complexity of a component's functionality and operations.
Interface Complexity	The level of effort to interact with another hardware component.
Product Support	The hardware warranty provided by the hardware vendor.
Experience with Component	The overall technical experience of the engineers handling the hardware.
Learning rate	A measure of the technical personnel mastering the installation/replacement of the hardware in relation to some specification of time
Reliability	The probability of performing a specified function without failure under given conditions for a specified period of time.
Confidence level	The level of comfort of having this hardware lives within the current system infrastructure.

As can be seen, the cost drivers within the Hardware category are considerably different than in the system's category. Indeed, different concerns may be relevant when purchasing new hardware, compared to the purchase of new system resources. Obvious considerations include the ease with which the new systems will integrate with existing architecture, and the reliability of the hardware. Less obvious drivers, on the other hand, may include the speed with which employees are going to be able to learn the new systems. In total, however, each of these drivers were devised due to their direct relevance to the purchase of new hardware equipment. Many of these driver were utilized from the COSYSMO model.

Table 5. Software Cost Drivers

Confidence level	The level of comfort of running this application live within the current system infrastructure.
Lines of Codes	The total number of lines of codes required to run this application.
Redesign required	The necessity of re-organizing the layout.
Retest required	The necessity of examining the software for quality assurance purpose.
Reimplementation required	The necessity of enhancing the functionalities of the application.
Time constraints	The total time allowed performing any tasks relevant to this application.

The considerations that need to be taken into account when purchasing new software are numerous, and quite different from considerations regarding hardware and system architecture. These considerations may include the ease with which the software can be implemented on the existing architecture, the ease with which users will be able to make use of the software, and the speed with which the software is able to undertake its core duties. Less obvious, but equally relevant, may be the total lines of code that are required to run the program. Longer code may lead directly to longer application time, and also may require increased IT costs if the software ever requires modification.

Table 6. Support Cost Drivers

Learning Rate	A measure of the technical personnel mastering the maintenance in relation to some specification of time.
Professional Experience	The technical expertise from the staff or the vendor technical support team to escalate all the issues that might arise.
Cost	The annual monetary spending for maintaining the current server infrastructure.
Repairs	The frequency rate of fixing any hardware component or software.
Call center	The 24/7 surveillance center for monitoring any server failure and coordinating the failure to the appropriate teams.
Upgrades	The rate of upgrading the current server infrastructure design or functionalities.

The last category of drivers is drivers related to support. This category can be considered relatively analogous to the 'Resources' category in the TRUE IT cost model. In short, the drivers in this category relate to the specific needs of those individuals who may be required to provide ongoing maintenance and support of the system/hardware/software. Drivers in this category thus include the learning rate of the IT personnel, the frequency with which upgrades are required on the new hardware/software, and the cost of repairs to the system, if repairs become necessary.

CHAPTER 5: A BRIEF ON THE STATISTICAL TECHNIQUES EMPLOYED

This section describes the statistical methods and tools used to analyze the data. In particular, a brief overview of the SAS programming language will be provided, along with a description of the primary statistical techniques employed within the thesis, primarily the student's t-test and the Pearson correlation. A brief description of the Analytical Hierarchy Process (AHP) method follows, along with a discussion of how this method may prove useful for estimating IT-related costs. A simple AHP example is provided for illustrative purpose.

SAS

The primary data analyses relevant to the present thesis were performed within SAS. SAS exists as one of the leading professional scientific statistical analysis applications on the market today, and provides a complete, comprehensive and integrated platform for data analysis. Experienced SAS users may be interested in Appendix C, which provides all of the source code developed for analyzing the present data.

Student's T-test

The student's t-test is a standard statistical equation used to evaluate the magnitude of differences between two groups on a variable. Specifically, the t-statistic serves as a measure of effect size, and is calculated as follows:

$$t = \frac{\overline{X}_1 - \overline{X}_2}{s}$$

Where *s* is the grand standard deviation, 1 = group one, 2 = group two.

The strategy employed throughout this thesis was to separate the survey participants into two broad groups: those whose companies existed with the technology sector, and those whose businesses exist within the financial services sector. A primary goal of the study was to test the generalizability of the cost model developed. Although complete generalizability cannot be evaluated within a single study, and initial test of this proposition would consider the extent to which the cost model holds accurate within the two sectors chosen. Thus, a t-test was used to evaluate the magnitude of any differences that existed between companies within each sector. The results of these analyses are discussed in Chapter 6: "Analysis, Results and Discussion".

Pearson Correlation

The Pearson product-moment correlation coefficient (PMCC) was created by Karl Pearson to measure the tendency for the value of two variables to vary in correlated fashion. If this were true, then as one variable increased, so would the other variable.

According to (Schmuller, 2005), the Pearson coefficient can be expressed as:

$$r=\frac{\left[\frac{1}{N-1}\right]\sum(x-\bar{x})(y-\bar{y})}{s_x s_y}$$

where s_x is standard deviation of variable x and s_x is the standard deviation of variable *y*. Because the numerator can be broken down to equal the covariance between variables x and y, the Pearson correlation can also be written in the following manner:

$$r=\frac{cov(x,y)}{s_x s_y}$$

As the pearson coefficient changes from zero, the magnitude of the association between the two variables increases. A positive Pearson coefficient indicates that the relationship between the two variables is such that as one variable increases, the other variable also increases. A negative Pearson coefficient indicates, in turn, that the relationship between the two variables is reciprocal – that is that as one variable increases, the other variable decreases systematically.

Analytical Hierarchy Process (AHP)

Analytical Hierarchy Process (AHP), developed in the 1970s by Dr. Thomas Saaty exists as an extremely robust and flexible decision making process (Saaty, 1979). The primary function of AHP is to help people set priorities and make adaptive decisions in complex situations. It is able to handle both qualitative and quantitative decision-making scenarios, and has been shown to improve on a number of human decision-making errors that can occur from a lack of focus and a lack of participation, as well as from conflicts regarding ownership and planning. Most particularly, AHP provides a useful mechanism for checking the consistency of both objective and subjective human evaluations. A primary benefit of AHP is that it helps set priorities and increases the likelihood that the most optima decision will be arrived at, even when complex qualitative and quantitative considerations must be taken into account.

The computations employed by AHP can be decomposed into the following six steps:

1. Form a pair wise comparison matrix with all the variables of interest.

$$\underline{A} = \begin{bmatrix} w_1/w_1 & w_1/w_2 & w_1/w_3 & \cdots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & w_2/w_3 & \cdots & w_2/w_n \\ w_3/w_1 & w_3/w_2 & w_3/w_3 & \cdots & w_3/w_n \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ w_n/w_1 & w_n/w_2 & w_n/w_3 & \cdots & w_n/w_n \end{bmatrix}$$

Where each *w* is the corresponding weight (or importance level) of each individual variable, and each entry within the matrix represents the relative weight ratio between two variables.

2. Gather all the data, convert the fraction ratio into decimal format, and calculate the average for each variable in the matrix.

3. Calculate the matrix product of the matrix. The matrix product is formed by multiplying, element by element, each row of the first factor, A, by corresponding elements of the second factor, w, and adding. Thus, the first element of the product would be:

   $(w_1/w_1)*w_1+ (w_1/w_2)*w_2+ \ldots..+ (w_1/w_n)*w_n= nw_1$. Similarly, the second element would be $(w_2/w_1)*w_2+ (w_2/w_2)*w_2+ \ldots..+ (w_2/w_n)* w_n= nw_n$. The n^{th} element would be nw_n. Thus, the resulting vector would be nw.

4. Compute the eigenvector of the matrix A.

5. Sum the rows and row totals, and standardize the result (This is the first approximation of the eigenvector).

$$\begin{bmatrix} w_1/w_1 & w_1/w_2 & w_1/w_3 & \dots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & w_2/w_3 & \dots & w_2/w_n \\ w_3/w_1 & w_3/w_2 & w_3/w_3 & \dots & w_3/w_n \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ w_n/w_1 & w_n/w_2 & w_n/w_3 & \dots & w_n/w_n \end{bmatrix} * \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \dots \\ \dots \\ \dots \\ w_n \end{bmatrix} = \begin{bmatrix} nw_1 \\ nw_2 \\ nw_3 \\ \dots \\ \dots \\ \dots \\ nw_n \end{bmatrix}$$

$$\underline{A} \quad * \underline{w} = n\,\underline{w}$$

6. Finally, normally by iterate steps three and four additional times in order to arrive a consistent matrix, with the following features, can be arrived at:

$$a_{ik}a_{kj} = a_{ij} \text{ for all } i,j.$$

To increase clarity regarding the AHP process, an example of computing AHP Importance Level Scaling is provided below:

1 equal **3 moderate** **5 strong** **7 very strong** **9 extreme**

(1) Forming the pair-wise comparison matrix

	Apple	**Orange**	**Banana**
Apple	1/1	1/2	3/1
Orange	2/1	1/1	4/1
Banana	1/3	1/4	1/1

(2) Gather all the data, convert the fraction ratio into decimal format, and calculate the average for each variable in the matrix.

	Apple	Orange	Banana
Apple	1.0000	0.5000	3.0000
Orange	2.0000	1.0000	4.0000
Banana	0.3333	0.2500	1.0000

(3) Calculate the matrix produce of the matrix:

That is, a_{11} = (1.0000 * 1.0000) + (0.5000 * 2.0000) + (3.0000 * 0.3333) = 3.0000, and following the computation for a_{12} … to a_{33}.

As a result, the first matrix product is:

	Apple	Orange	Banana
Apple	3.0000	1.7500	8.0000
Orange	5.3332	3.0000	14.0000
Banana	1.1666	0.6667	3.0000

(4) Compute the eigenvector of the matrix A.

$$\begin{bmatrix} 3.0000 & + & 1.7500 & + & 8.0000 \\ 5.3332 & + & 3.0000 & + & 14.0000 \\ 1.1666 & + & 0.6667 & + & 3.0000 \end{bmatrix} \begin{matrix} = \\ = \\ = \end{matrix} \begin{matrix} 12.7500 & 0.3194 \\ 22.3332 & 0.5595 \\ 4.8333 & 0.1211 \end{matrix}$$

39.9165	**1.0000**

(5) Sum the row totals and standardize the summed totals. The standardized data becomes the eigenvector:

$$\textbf{Eigenvector} = \begin{bmatrix} 0.3194 \\ 0.5595 \\ 0.1211 \end{bmatrix}$$

Finally, iterate steps three and four, three additional times to obtain the consistent matrix, which will give you the following:

	Apple	**2nd most important criterion**
The consistent eigenvector =	**Orange**	**1st most important criterion**
	Banana	**3rd most important criterion**

This eigenvector is known as the weighing factor for each variable in the matrix.

Examples of Real-world AHP applications

According to (Khosrow-Pour, 2006), there are numerous real-world examples of successful AHP applications. Some of these applications may include strategic planning, microcomputer selection, software productivity measures and budget allocation. In addition, other less obvious applications may exist, including consideration of oil pipeline routes, the flexibility of specific manufacturing systems, and the planning of energy policy (Finnie, Wittig, & Petkov, 1993; Hamalainen & Seppalainen, 1986; Lee, 1993; Ramanathan & Ganesh, 1995; Saaty, 1994).

CHAPTER 6: ANALYSIS, RESULTS & DISCUSSION

Survey Results

The following demographic data describe the characteristics of the total sample surveyed in the present thesis:

Forty-three percent of individuals surveyed (3 out of 7) were from the Technology sector, while 57% of those surveyed (4 out of 7) were from Financial Services sector. Nearly half of those surveyed (43%, or 3 out of 7) worked at companies with a workforce greater than 10,000 employees, 14% of those surveyed (1 out of 7) worked for companies with a workforce between 500 and 1000 employees, and 2 out of 7, or 29% of those surveyed, worked for companies with a workforce under 100 employees. Despite the varying company sizes, all of the employees surveyed worked for companies that reported an annual IT-budget of over 2 million dollars. Thus, each of these companies had significant IT-related needs, and thus served as appropriate models for use within the present thesis. The average of total work experience was 20.1 years, while the average length of IT-specific experience was 17.7 years

Table 7 displays additional descriptive statistics related to each of the companies surveyed in the present thesis. As can be seen from this data, the average number of users across each site was 2,765 and the average number of sites was 6.5. Furthermore, the average number of distinct applications was found to be 281, and the average number of software licenses was 887. Thus, the companies surveyed appeared to utilize IT infrastructure that incorporated a substantial degree of complexity.

Table 7 – Descriptive statistics pertaining to the companies surveyed.

Variable	Label	N	Mean	Std Dev	Minimum	Maximum
ExperYear	Total number of years of your work experience	7	20.14	3.80	15	24
ITExper	Years of experience in IT	7	17.714	4.89	8	22
BUD_SYS	System	7	28.57	17.49	10	65
BUD_SW	Software	7	25.71	9.32	10	40
BUD_HW	Hardware	7	22.86	8.59	5	30
BUD_SUP	Support	7	22.86	11.13	10	45
SUP_U	Users	4	2765.00	4827.48	110	10000
SUP_SL	Software licenses	4	887.50	606.05	50	1500
SUP_S	Sites	4	6.50	5.92	2	15
SUP_DA	Distinct applications	4	281.00	481.17	4	1000
Rank_Rack	Number of servers or racks	7	2.43	1.51	1	5
Rank_Lice	Number of software licenses	7	2.71	2.87	1	9
Rank_KVM	Number of KVMS(terminal servers)	5	7.40	2.19	5	9
Rank_Sites	Number of distinct sites(data centers)	6	4.67	2.73	1	7
Rank_Users	Number of users/PC equipments	7	3.14	2.12	1	6
Rank_Feet	Number of feet of FA-CL or CAT5 cables	5	7.80	1.09	7	9
Rank_Appli	Number of software applications to be supported	7	2.86	2.34	1	7
Rank_DBS	Number of databases	7	3.00	2.16	1	7
Rank_Data	Number of data & phone jacks	6	5.00	3.52	1	9

Cost Drivers

This section shows the AHP results of the four categories of cost drivers listed in section II of the survey.

Systems

The following four tables represent each of the four iterations of the matrix product. These matrices are listed in sequential order as part of the AHP pair-wise method described in Chapter 2.

Table 8 – The result of first iteration of computing the matrix product

Matrix 1 of Table 1 (Systems)								
47.808232	30.138119	40.981666	28.345904	44.659838	34.521396	69.897612	68.666418	85.331335
61.561623	41.225938	55.030153	41.207227	62.29698	49.280739	100.77255	91.224933	120.66813
42.806033	26.081831	39.917676	26.809977	48.165431	35.339327	58.096232	60.363457	64.930183
53.364853	32.736281	51.690681	37.532249	55.940736	41.509577	79.972901	78.99319	104.13134
42.042636	28.390652	37.702821	28.692107	36.783305	21.893638	48.177279	54.507882	80.031138
40.459122	24.154013	37.540361	24.484619	35.462819	23.591442	42.822056	51.897269	74.728642
28.494882	19.324993	21.971382	19.212532	26.449251	14.272892	25.833965	25.035469	40.471478
19.118493	15.437179	16.49588	15.077887	20.908721	11.045665	19.812167	17.66651	24.004382
23.382357	15.145883	22.48859	18.859695	31.436505	21.823762	34.692135	30.765768	34.829528

Table 9 – The results of second iteration of computing the matrix product

Matrix 2 of Table 1 (Systems)								
15982.009	10485.096	14286.067	10873.885	16406.16	11171.061	20782.917	20559.275	26932.014
22085.803	14481.477	19749.665	15047.175	22678.559	15417.96	28658.514	28365.124	37231.281
14574.073	9543.1366	13043.811	9862.6967	14823.77	10087.789	18866.646	18823.94	24740.872
19037.376	12467.64	17061.255	12968.964	19582.05	13330.69	24749.771	24540.577	32107.094
13621.313	8917.8857	12239.637	9297.6097	14126.563	9731.9805	18095.292	17791.731	23074.772
12739.99	8332.9456	11451.604	8677.3006	13197.438	9078.291	16846.81	16627.148	21536.853
8368.3092	5451.8275	7525.1379	5659.6331	8567.5681	5954.2781	11204.98	11080.098	14430.11
6164.6487	4016.0123	5540.4075	4156.3105	6262.335	4349.715	8231.077	8167.4085	10696.082
8615.1501	5625.5636	7721.8173	5821.2862	8712.036	5929.3917	11138.573	11164.287	14749.357

Table 10 - The results of third iteration of computing the matrix product.

Matrix 3 of Table 1 (Systems)								
1.80069E9	1.17809E9	1.61426E9	1.22281E9	1.84533E9	1.26321E9	2.35845E9	2.33827E9	3.05957E9
2.48788E9	1.62769E9	2.2303E9	1.68946E9	2.54955E9	1.74527E9	3.25846E9	3.23059E9	4.22716E9
1.63906E9	1.07235E9	1.46936E9	1.11303E9	1.67963E9	1.14976E9	2.14668E9	2.12837E9	2.78501E9
2.14673E9	1.40449E9	1.92448E9	1.4578E9	2.19995E9	1.50597E9	2.81169E9	2.78763E9	3.64755E9
1.54635E9	1.01168E9	1.38627E9	1.05008E9	1.5847E9	1.08486E9	2.0255E9	2.00814E9	2.62753E9
1.44419E9	944842873	1.29469E9	980705981	1.48E9	1.01318E9	1.89167E9	1.87546E9	2.45393E9
950515025	621852921	852122823	645443521	974023399	666806453	1.24503E9	1.23442E9	1.61522E9
699186179	457426014	626807263	474774363	716456771	490471517	915801280	908008919	1.18815E9
968605455	633702386	868319036	657737913	992546199	679425366	1.26856E9	1.25776E9	1.64584E9

Table 11 - The results of third iteration of computing the matrix product.

Matrix 4 of Table 1 (Systems)								
2.2962E19	1.5023E19	2.0585E19	1.5593E19	2.3531E19	1.6108E19	3.0075E19	2.9818E19	3.9017E19
3.1725E19	2.0756E19	2.8441E19	2.1544E19	3.2511E19	2.2256E19	4.1553E19	4.1198E19	5.3907E19
2.0901E19	1.3674E19	1.8737E19	1.4193E19	2.1419E19	1.4662E19	2.7375E19	2.7142E19	3.5514E19
2.7375E19	1.791E19	2.4541E19	1.8589E19	2.8053E19	1.9204E19	3.5855E19	3.5549E19	4.6515E19
1.9719E19	1.2901E19	1.7678E19	1.3391E19	2.0208E19	1.3833E19	2.5828E19	2.5607E19	3.3507E19
1.8417E19	1.2049E19	1.651E19	1.2506E19	1.8873E19	1.2919E19	2.4122E19	2.3915E19	3.1293E19
1.2121E19	7.9301E18	1.0866E19	8.2311E18	1.2421E19	8.5031E18	1.5876E19	1.574E19	2.0596E19
8.916E18	5.8332E18	7.993E18	6.0546E18	9.1369E18	6.2547E18	1.1678E19	1.1578E19	1.515E19
1.2351E19	8.0807E18	1.1073E19	8.3874E18	1.2657E19	8.6646E18	1.6177E19	1.6039E19	2.0987E19

The following summary table shows the final normalized eigenvector in percentage format (from Matrix 4 of Table 1 (Systems)):

Table 12 – The summary table of the final normalized eigenvectors (pertaining to Table 3).

	Total	
	Sum	PctSum
ID		
1	2.127131E20	13.16
2	2.9388922E20	18.18
3	1.936177E20	11.98
4	2.5359117E20	15.69
5	1.8267251E20	11.30
6	1.7060389E20	10.55
7	1.1228509E20	6.95
8	8.2594434E19	5.11
9	1.1441757E20	7.08
All	1.6163847E21	100.00

Figure 7 ranks the relative importance, in descending order, of cost drivers listed in Table 3.

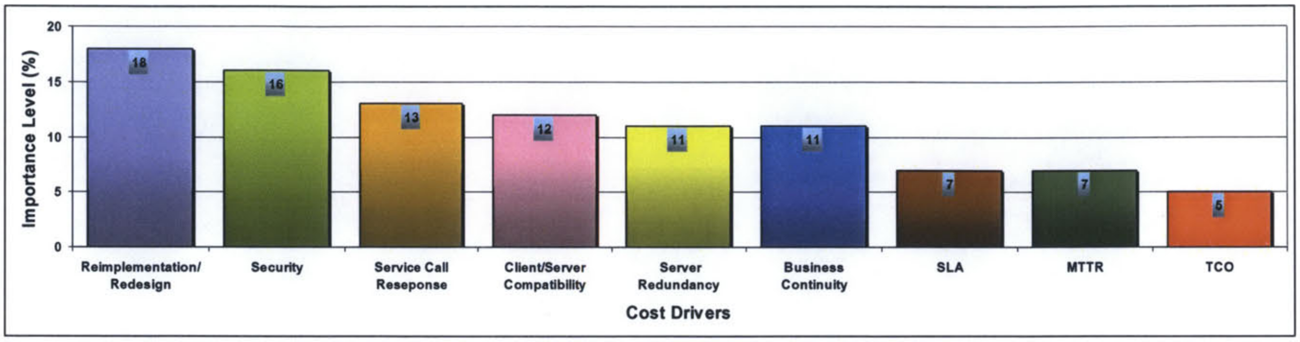


Figure 7 – The importance level of the cost drivers in Table 3.

As can be seen, the three most important cost drivers within the service section were:

1. **Reimplementation / Redesign (18%)**

2. **Security (16%)**

3. **Service call response (13%)**

The majority of the cost drivers in this table are relatively similar. Nevertheless, it is clear that both 'reimplementation/redesign' and 'security' are characteristics of the IT system that are

Hardware

Figure 8 ranks the relative importance, in descending order, of all cost drivers in Table 4

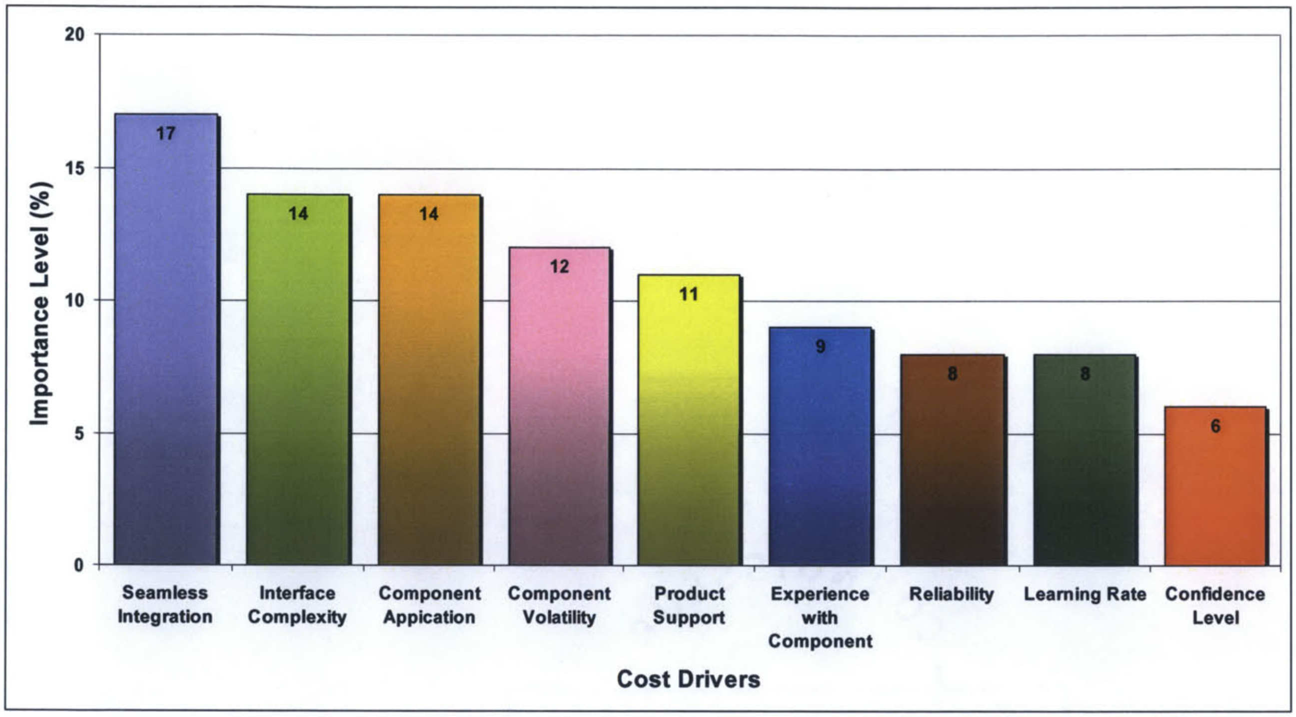


Figure 8 – The importance level of the cost drivers in Table 4

. As can be seen, the top three most important cost drivers in terms of "Hardware" are

1. **Seamless integration (17%)**

2. **Interface Complexity & Component application (14%)**

3. **Component Volatility (12%)**

Similar to the AHP result from Table 3, the importance attributed to each cost driver in Table 4 is relatively evenly distributed. Only 'seamless integration' appears to be of greater important to IT professionals than the other cost drivers. Thus, we can assume that the smooth introduction of new hardware into the existing IT infrastructure is a primary concern.

Software

Figure 9 ranks the relative importance, in descending order, of all cost drivers in Table 5 in descending order.

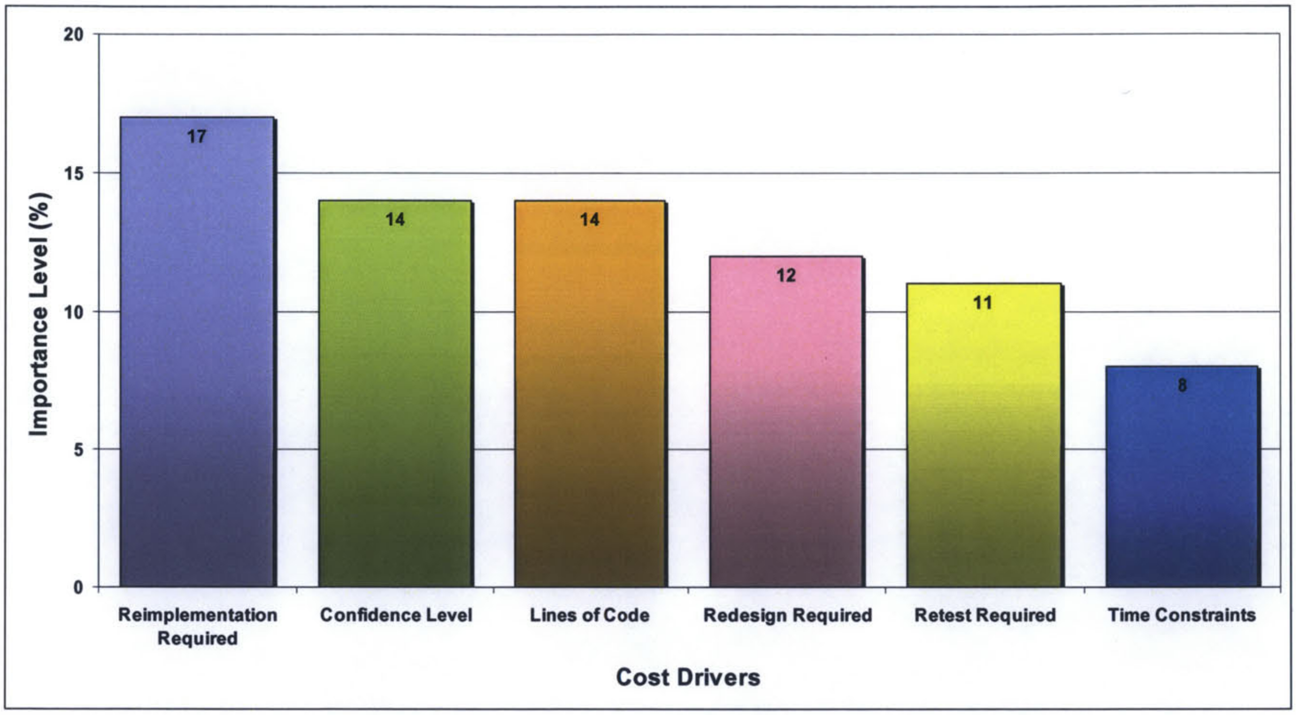


Figure 9 – The importance level of the cost drivers in Table 5

As can be seen, the three most important cost drivers in terms of "Software" are:

1. **Reimplementation required (20%)**

2. **Confidence Level (19%)**

3. **Lines of code & Redesign required (17%)**

While the AHP results suggest an equal distribution of importance ratings, both "Reimplementation" and "Confidence level" received slightly higher weightings than the other possible features of IT software. Thus, it appears that companies rely primarily on their software engineers' capability to improve their software and to keep existing software available and running stability.

Support

Figure 10 ranks the relative importance, in descending order, of all cost drivers in Table 6 in descending order.

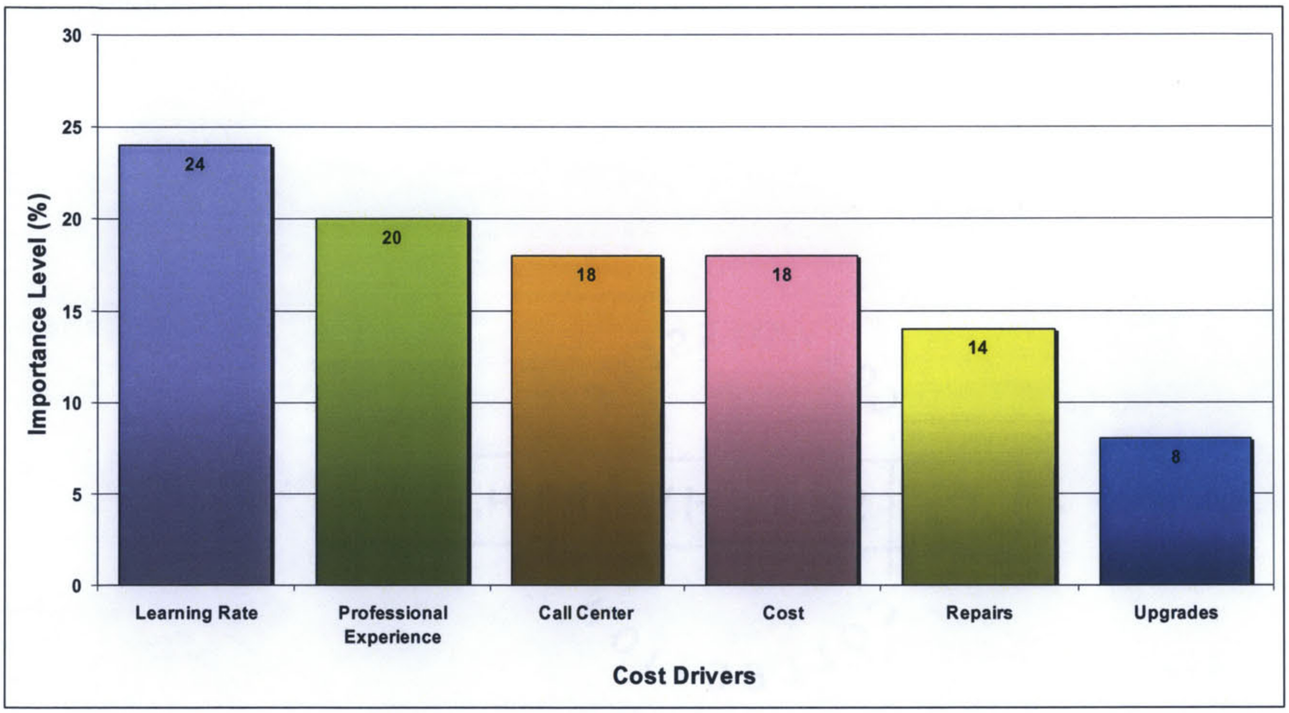


Figure 10 – The importance level of the cost drivers in Table 6 - Support

As can be seen, the three most important cost drivers in terms of "Support" are:

1. **Learning rate (24%)**

2. **Professional Experience (20%)**

3. **Call center (19%)**

With regard to IT support, those surveyed suggested that learning rate was the most important characteristic. In addition, the extent of professional experience, and the call-center cost parameters received high weightings as well. It may be that companies focus primarily on their staff's professional expertise, and concern themselves with ensuring that the technical ability of their staff is sufficient to support relevant knowledge expansion, and to accommodate necessary IT infrastructure upgrades in the future.

Sector-Specific Similarities and Differences

In the following two subsections, the cost driver rankings associated with each of the technology and financial services industries are compared and contrasted, in order to evaluate similarities and differences across the two sectors. It is hoped that such an analysis will aid consideration of the extent to which the cost model put forward within may be readily adapted for addition sectors, not directly tested at present.

Systems

Table 13 compares on contrasts the rankings of System cost drivers across each of the two sectors

Table 13. Comparison of System Cost Drivers Across Financial and Technology Sectors

Driver	Importance level for Financial Sector	Importance level for Technology Sector
Security	19	10
Reimplementation	15	20
Business continuity	13	9
Server redundancy	12	10
Client/server	10	12
SLA	9	5
MTTR	9	5
Service call Response	7	24
TCO	6	5

As can be seen, there were significant differences across the two sectors. In the financial services sector, for instance, security of the system infrastructure was considered the most important driver, followed by the amount of reimplementation that may be required, and the extent to which business continuity may be kept. In the technology sector, in contrast, the speed of service calls was the most important driver, followed by reimplementation considerations, and the client/server compatibility. The fact that the financial sector values security, whereas technology companies value service call response is interesting, and may provide insight into the nature of the companies in each sector. Financial service companies may, for instance, require particularly tight security in order to ensure that hackers cannot access public records. Indeed, keeping hackers out is necessary, not just to retain people's money, but also to keep consumer confidence high. Without appropriate security, the financial services company cannot survive. This is not similarly the case for many high tech companies, however. It is not, for instance, going to lower consumer confidence substantially if a telecom company has its computer network's hacked. That service call response was most important may have more to do with the specific high tech companies surveyed – companies in the service call industry would, no doubt, consider this a high priority.

Hardware

Table 14 compares on contrasts the rankings of Hardware cost drivers across each of the two sectors

Table 14. Comparison of Hardware Cost Drivers Across Financial and Technology Sectors

Driver	Importance level for Financial Sector	Importance level for Technology Sector
Interface Complexity	15	10
Seamless Integration	13	24
Component Application	12	20
Reliability	12	5
Component Volatility	11	12
Experience with Component	10	9
Product	9	10
Learning Rate	9	5
Confidence Level	8	5

While there are again differences across the two sectors, with regard to hardware it is more interesting to note the similarities. Indeed, the top three drivers: Interface Complexity, Seamless

Integration and Component Application, were the same for companies in both industries (although the ranking within the top three differed). Thus, companies in both sectors are extremely concerned with the integration and complexity of any new hardware brought into the company.

Software

Table 15 compares on contrasts the rankings of Software cost drivers across each of the two sectors

Table 15. Comparison of Software Cost Drivers Across Financial and Technology Sectors

Driver	Importance level for Financial Sector	Importance level for Technology Sector
Reimplementation	26	9
Time Constraints	21	5
Redesign	16	17
Retest Required	14	14
Lines of Code	13	26
Confidence Level	10	29

Note the staggering disagreement across the two sectors. Indeed, they are almost reciprocal opposites, with the financial sector being particularly interested in the amount of time that reimplementation and redesign of the software will require, while technology companies are instead much more concerned with their confidence in the software, the efficiency of the code, and the number of retests that may be needed before complete confidence is established. The reasons for this drastic difference are unclear, but should be evaluated further in future research.

Support

Table 16 compares on contrasts the rankings of Support cost drivers across each of the two sectors

Table 16. Comparison of Support Cost Drivers Across Financial and Technology Sectors

Driver	Importance level for Financial Sector	Importance level for Technology Sector
Call Center Service	26	6
Learning Rate	18	39
Professionalism	17	24
Repairs	16	8
Cost	13	18
Upgrades	11	4

Again, substantial differences resulted across the two sectors. Whereas financial service companies were particularly concerned with the call center service and professionalism, technology companies were unanimously concerned with the learning rate of their support staff. Indeed, this may again have to do with differences in the business models of companies in both sectors. Support staff for technology companies is likely to be required to perform cutting edge updates and maintenance, requiring highly skilled individuals with the capacity for fast learning. The financial services support staff may, instead, be more likely to be part of the financial service's call center, and may, thus, not require the same level of skilled expertise.

T-Test Results: How Exactly Did the Two Sectors Differ?

A series of Student's t-tests were performed, in order to identify those cost drivers were significantly more important within one sector than the other. Thus, these t-test provide empirical support for the descriptive findings narrated above. Table 17 displays those cost drivers that did, in fact, reach significance (p = .05). As can be seen, nine pairs of variable comparison between the two reached significance. Thus, the first row in Table 17 indicates that the two sectors differed significantly with regard to their relative importance of Service call response time and reimplementation, a finding that mirrors that described above.

Table 17 – Nine pairs of cost drivers that have contrasting importance level between two sectors

Two Variables Comparison	Financial Sector (n=4) Mean of importance	Technology Sector (n=3) Mean of importance	P value
Service call response time vs. Reimplementation/Re-design	0.2048	4	0.0035
Service call response time vs. Security	0.3635	7	0.0063
Re-implementation /Re-design vs. Security	0.4	6.3333	0.0106
Client/Server compatibility vs. Security	0.4524	6.3333	0.0110
Component application complexity vs. Interface complexity	0.418	5	<0.0001
Product support vs. Confidence level.	1.4444	6.3333	0.0094
Lines of code vs. Re-implementation required	0.8492	5	0.0232
Professional experience vs. Repairs	1.6333	5.6667	0.0393
Cost vs. Upgrades	0.7333	5	0.0228

Pearson Correlations: Were there Linear Patterns Within the Driver Rankings?

As a final analysis, Pearson Correlations were computed to investigate the possibility that there may be specific linear relationships between those cost drivers that were rated as most important, and those cost drivers that were rated as less important. Such analyses are not of crucial importance, in and of themselves, but rather aid two issues: a) understanding of sector specific needs, and b) identification of redundancy in the current list of cost drivers. The results of two selected Pearson correlations, one that depicts each type of information, are reported in Figures 11 and 12 respectively. Interested readers will find details regarding all correlations, including details of the computations underlying these correlations, in Appendix D.

Systems: An Example of Sector Specificity

Figure 11 displays a scatterplot of the relationship between rankings of Service Call Response Time and Security, across both sectors. As can be seen, a significantly negative linear trend exists, whereby companies that ranked Service Response Time high tended to rank Security very low, and vice versa. Indeed, this relationship was $r = -0.85$, $p < .05$, indicating that the linear trend was statistically significant, even with this small a sample. It may be easy to understand the nature of this relationship by considering the interests of companies who are about to undertake new IT-infrastructure purchases. In the real world, for instance, companies that value smoothly running call centers may fall under a business model that does not require high levels of system security. In opposite fashion, those with a need for security likely do not undertake significant call center business.

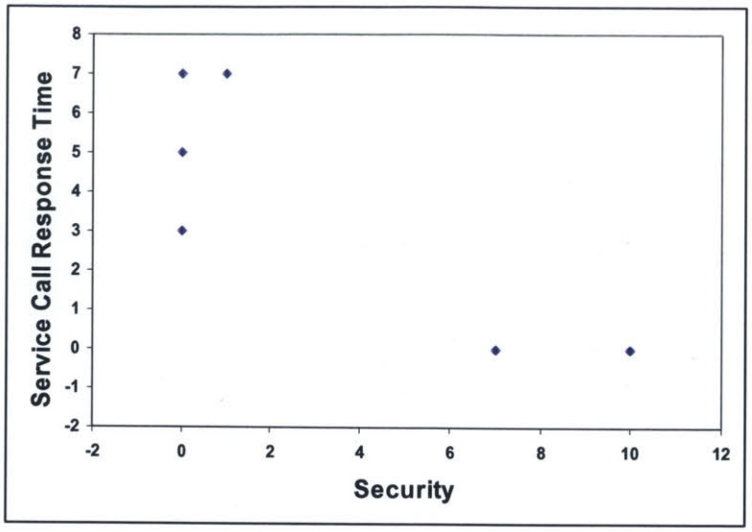


Figure 11. A symmetric scatter plot contrasting Service call response time and Security from Table 13.

Hardware: An example of Redundancy

Figure 12 displays a scatterplot of the relationship between rankings of Experience with Components and Learning Rate, across both sectors. As can be seen, a significantly positive linear trend exists, whereby companies that ranked Experience high also ranked Learning Rate high, and vice versa. This, of course, makes complete sense, as Experience with the Hardware should correlate highly with one's Learning Rate in the real world. Thus, an individual with greater experience should require less time to learn the hardware. This positive relationship thus indicates that one of these two components may be redundant within the model, and that a more parsimonious model, with only one of these two variables, may remain equally predictive of overall IT expenditures.

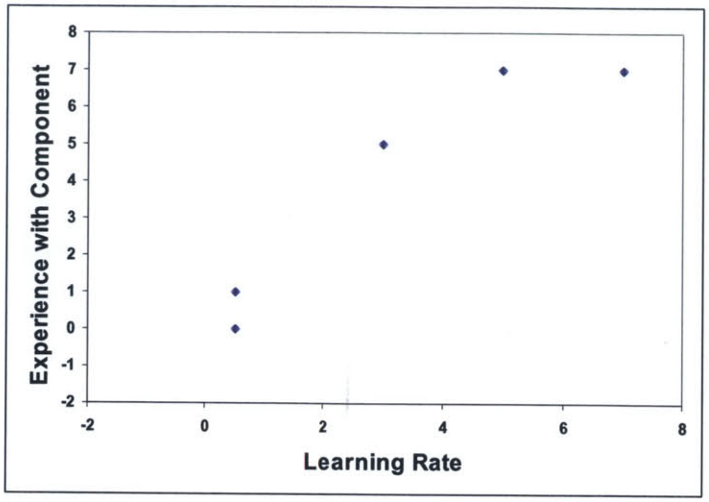


Figure 12. A symmetric scatter plot contrasting Experience with component and Learning rate from Table 14.

These two scatterplots thus identify two important aspects of the collected data. First: there are, indeed, considerable sector-specific differences in the IT-related needs, and any global cost model developed must be able to account for these differences. Indeed, the current model, while not explicitly tested, appears likely to be capable of adapting to the unique needs of both the financial services and technology sectors by utilizing variable weightings in line with the relative rankings of each cost driver surveyed. Second: the current cost model remains in need of several iterations, in order to make it as efficient and parsimonious as possible. Indeed, as can be seen in Appendix D, there were a large number of highly correlated variables, indicating that many of those variables may be redundant within the model. Future research comparing the complete set of cost drivers against a smaller subset would enhance the accuracy and preciseness of constructing an IT cost model.

Summary of Findings

According to the research results of META Group Inc. (2003) and additional industrial market research reports that were described in chapter 2, the cost to estimate IT project costs now exceeds $20K per $1M project. Because IT spending has consistently increased by 1%-6% a year, it would be of great worth to develop a set of common cost drivers that can estimate the cost of IT projects across a variety of sectors. The present thesis specifically surveyed companies within the financial services and technology sectors. Indeed, within these sectors, there does appear to be a set of essential deterministic cost drivers that can accurately project the total cost of IT infrastructure when appropriate weightings are included in the model. Future research will be required to further this research, and investigate the applicability of the currently devised cost drivers within additional sectors.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

This chapter encapsulates the research results and also discusses some of the factors that may limit the scope of the present work. This chapter also describes the cornerstone tools that can be used for further research.

Known Problems and Limitations

Several challenges were encountered during the course of the project. First, because of the need to find senior VPs in the IT industry to agree to participate in the research, the total sample size of the project remains unfortunately small. Indeed, with seven participants, it is difficult to know how well the results obtained will generalize more widely. Second, most company budgets are confidential, and the difficulty obtaining detailed information on the specific budgets of the surveyed companies limited the ability to perform a systematic investigation of the relationship between the cost driver weightings and the actual dollars spent on IT. It may be that the rankings obtained do not correlate exactly with actual dollars spent, and future research will be required to determine the true relationships between these constructs. Finally, the model was not tested in the real world, and future research will need to do this to determine the accuracy of the model.

Future Research

The findings from the present thesis can serve as a good starting point both for further academic research and commercial applications. By using the set of tools developed, an online CIO survey with the capability of exporting the data in SAS readable format and a SAS program that can dynamically generate the data set from the survey data file and compute the AHP eigenvectors, and can help calibrate the cost driver rankings for any sector. An increased sample size will make this research more relevant outside of the two domains studied. Revisiting the list of cost drivers would also be necessary to verify its relevance in an industrial setting.

Additional future work could include expanding the scope of this study to another level that could potentially resolve other IT cost related issues. A very popular and current topic is the issue of strategic IT outsourcing. It would be interesting to see how we can perform intelligent trade-off analysis between outsourcing and in-house IT operations, thus organizations can make the best cost-effective decisions.

Conclusion

In summary, based on a complete analysis of the survey data, the findings appear to validate the proposition that there is a finite set of essential deterministic cost drivers that, with varying weighing factors, may prove capable of estimating the total cost of ownership for IT infrastructure. While the present data suggest its applicability within two varying sectors, additional work is required to test its global applicability. Additional work is also required to identify cost drivers that are good candidates for elimination within the model. Nonetheless, the present work provides encouraging results, and suggests that a global cost model is within reach. The significance of such a model for the bottom line of companies who depend on IT infrastructure for their success makes continued investigation paramount.

REFERENCES

Analytic Hierarchy Process – AHP definition in Wikipedia < http://en.wikipedia.org/wiki/Analytic_Hierarchy_Process>

Finnie, G. R., Wittig, G. E., & Petkov, D. I. (1993). Prioritizing software development productivity factors using the analytic hierarchy process. *Journal of Systems & Software, 22*(2), 129-139.

Forman E. H. (2002) Decision by Objectives, World Scientific Publishing Company <http://mdm.gwu.edu/Forman/DBO.pdf>

Hamalainen, R. P., & Seppalainen, T. O. (1986). The analytic network process in energy policy planning. *Socio-Economic Planning Sciences, 20*(6), 399-405.

Jernigan J. (2005) Segmenting the Business Market – There's More to it than Size, In-Stat

Khosrow-Pour M. (2006) Advanced Topics in Information Resources Management, Volume 5, Idea Group Publishing

Lee, H. (1993). A structured methodology for software development effort prediction using the analytic hierarchy process. *Journal of Systems & Software, 21*(2), 179-186.

Plunkett Research, Ltd. (2006) Plunkett's Info Tech Industry Almanac 2006

Plunkett Research, Ltd. (2006) InfoTech Industry Trends.

Ramanathan, R., & Ganesh, L. S. (1995). Using AHP for resource allocation problems. *European Journal of Operational Research, 80*, 410-417.

Saaty, T. L. (1994). Highlights and critical points in the theory and application of the analytic hierarchy process. *European Journal of Operational Research, 74*, 426-447.

SAS Official website of SAS INSTITUTE INC. <http://www.sas.com/index.html?sgc=u>

Schmuller J. (2005) Statistical Analysis with Excel For Dummies, John Wiley & Sons

SER Serlio Software <http://www.serlio.com/casecomplete/UseCaseBenefits.aspx>

Student's T test Merriam Webster's definition < http://www.m-w.com/cgi-bin/dictionary>

TRUE IT Overview (2005) Price Systems

UHCL UML overview slide at University of Houston Clear Lake < http://sce.uhcl.edu/helm/SWTOOLS/uml_overview/UMLOverview1/sld015.htm>

Valerdi R. (2005) The Constructive Systems Engineering Cost Model (COSYSMO), PhD Dissertation, University of Southern California

APPENDIX A: TWO USE CASES OF THE CURRENT COST MODEL

Below is a detailed description of two IT usage scenarios. These scenarios are intended to illustrate the complexity of IT-needs determinations, and to provide running examples of the manner in which costs associated with a given cost model may be arrived at with the currently proposed cost driver model.

Use Case Fundamentals

Use case modeling has been a well-known method for software development. Use case modeling can be used to conduct unexpected event elimination and to identify operational exception errors early, and before serious complications ensue. In addition use case modeling can serve as a good source for technical documentation, for easy user validation, and for better project traceability throughout the entire project cycle.

The following two use cases were developed based on the most common scenarios of building/modifying an IT infrastructure.

CASE 1: COST OF A NEW IT SYSTEM/INFRASTRUCTURE

Name: Cost of a new IT system/infrastructure

Identifier: IT01

Description: The purpose of estimating the cost of a new IT system/infrastructure is to eliminate the common issues/obstacles that companies may experience due to over- or under-budgeting, or to the inaccurate allocation of human labor to a particular IT project. Implementation of the universal

IT cost model (the ultimate goal of this study), can lead to increased IT cost accuracy, and to more effective and more timely project management.

Primary Actors:

- Chief Information Officer (CIO)s/IT executive teams
- IT managers

Secondary Actors:

- IT systems engineers
- IT systems/storage administrators

Goal:

To gain a comprehensive understanding of the monetary and time expenditures that will go into developing a new IT system/infrastructure. In particular, costs associated with the implementation and maintenance of the IT infrastructure must be taken into account, as well as the labor costs associated with these processes.

Frequency:

At the commencement of a new IT infrastructure-building project.

Preconditions:

- A new project folder must be created.
- Define the project name, project start date and desired completion date.

- User can distinguish software development cost from IT infrastructure cost.

- issuer has a comprehensive understanding of the nature of each of the cost drivers (SLAs) and size drivers (servers) that will go into the calculation of the IT-infrastructure needs..

Post conditions:

- Project data has been saved.
- Inputs have been checked for manual entry errors.
- All required blank fields have been filled and confirmed with user as accurate.
- User will be able to modify / update the inputs in this project at any time.
- A risk assessment calculation and a graph can be compiled.
- The total estimated IT costs, as well as the corresponding estimated project completion date.
- A critical path / DSM can be graphically presented.
- User could customize output of the model.
- The model can display cost estimate in a format similar to other cost models

Assumptions:

- An IT project has a clear start date and target completion date.
- Required date inputs would be entered.

- The user has a working understanding of the manner in which pre-defined input parameters and drivers can be customized.

Basic Course:

- The user launches the application.
- The main screen appears, and the user creates a new project.
- The user enters the project start date and target completion date.
- The user drags the desired functionalities out of the various IT categories onto the project in a tree hierarchy fashion.
- The user enters all of the required input parameters for each IT functionality (eg. cost drivers, size drivers and misc. drivers).
- The user can customize any pre-defined parameter values.
- The user clicks on the "Calculate" button to generate the cost of the IT needs.
- The user can highlight the parameters for risk assessment.
- The user saves the project.
- The user exits out of the application.

Alternate Courses of Action:

- The application experiences a syntax error, and informs the user of such.

- The user can import the input parameters from an Excel spreadsheet.

Issues:

- Will the user be able to define additional customized input fields?

- Should the user be able to import their data from files?

- Should the outsourcing IT cost be included as a relevant variable in the model?

Decisions:

- User determines the level of complexity for each driver.

Change history:

Created on Feb 23rd , 2006 by Ken Huang.

Created on Mar 2nd, 2006 by Ken Huang.

* * *

CASE 2: COST OF MODIFYING AN EXISTING IT SYSTEM/ INFRASTRUCTURE

Name: Cost of modifying an existing IT system/infrastructure (with SEER-SEM handling the SW development & maintenance)

Identifier: IT02

Description: The purpose of estimating the cost of modifying an existing IT system/infrastructure is to eliminate the common issues/obstacles that companies may experience due to over- or under-budgeting, to inaccurate re-budgeting, or to the mis-appropriation of human labor needs on a given project modification. Implementation of the universal IT cost model, can lead to increased IT cost accuracy, and to more effective and more timely project management.

Primary Actors:

- CIOs/IT executive teams
- IT managers

Secondary Actors:

- IT systems engineers
- IT systems/storage administrators

Goal:

To gain a comprehensive understanding of the costs associated with modifying an existing IT system/infrastructure. These costs may be anticipated to vary according to size and the complexity of the modification of IT infrastructure.

Frequency:

Every time a comparable IT infrastructure building project occurs. Approximately once a year.

Preconditions:

- A new project / folder must be created.
- The project name must be defined, and a project start date and desired completion date must be known.
- The user can distinguish software development cost from IT infrastructure cost.
- The user has a comprehensive understanding of the nature of each of the cost drivers (SLAs) and size drivers (servers) that exist within the IT cost model.

Post conditions:

- The project data has been saved.
- The inputs have been checked for manual entry errors.
- All required blank fields have been filled and confirmed with user as accurate.
- The user is able to modify / update the inputs in this project at any time.

- A risk assessment calculation and a graph can be compiled.
- The total estimated IT costs have an associated project completion date.
- A critical path / DSM can be graphically presented.
- The user could customize output of the model.
- The model can display cost estimate in a format similar to other cost models.
- The model can visually illustrate the impacted segments of the project.

Assumptions:

- An IT project that has a clear start date and target completion date.
- Required date inputs are entered appropriately.
- The user is able to customize all pre-defined input parameters and drivers.
- The user is able to save their project.
- The user is familiar with the existing IT infrastructure.
- The user is aware of their company's existing IT costs.

Basic Course:

- The user launches the application.
- The main screen is presented, and the user creates a new project.

- The user enters the project start date and target completion date.
- The user drags the desired functionalities out of the various IT categories onto the project in a tree hierarchy fashion.
- The user defines the existing IT infrastructure cost by entering all the required input parameters.
- The user selects "Upgrade/Modify the defined IT environment.", thus creating a template.
- The user enters all of the required parameters within this newly created template or its subnets (the categories he/she would like to modify or upgrade on).
- The user defines the impact relationship among each category.
- The user can customize any pre-defined parameter values.
- The user clicks on the "Calculate" button to generate the cost.
- The user can highlight the parameters for risk assessment.
- The user saves the project.
- The user exits out of the application.

<u>Alternate Courses of Action:</u>

- The application experiences a syntax error, and informs the user of such.
- The user can import the input parameters from an Excel spreadsheet.

Issues:

- Will the user be able to define additional customized input fields?
- Should the user be able to import their data from files?
- Should the outsourcing IT cost be included as a relevant variable in the model?

Decisions:

- User determines the level of complexity for each driver.

Change history:

Created on Mar 2^{nd}, 2006 by Ken Huang.

APPENDIX B: MIT SURVEY

Universal IT infrastructure Cost model **Ken Huang MIT SDM'05**

Thesis advisor: Dr. Ricardo Valerdi

CIO survey on IT infrastructure cost

Version 7.0

Your responses in this survey should reflect of your personal experience throughout your career and not be dramatically influenced by one abnormal experience. Participant information will remain anonymous. Participant information is collected for follow-up purposes only.

Participant Information:

Name: __________ _____________ ID
Last First
Division/Department/Business Unit:____________________ **DDBU**
Location:____________________ ADDRESS
Email address:_______________________ EMAIL
May we contact you for follow-ups? YES (1) or NO (0) CONTACT

Name of the Corporation: ___________________________ CORP

Industry:____________________ (drop down menu) INDUSTRY

Company size:_________________ (< 50, > 100, 500, 1000-5000, 10,000 +) Size

1 2 3 4 5

Years of experience in Information Technology: _________ ITEXPER

Total number of years of your work experience: _________ EXPERYEAR

Approximate Annual IT budget of your division or business unit:_____________ $US ITBUDGET

☐ < $50,000 ☐ $50K - $500K ☐ $500K - $2million

☐ > $2million

What is your typical budget distribution across these four categories? BUD_

System	_________%	SYS	Software	_________%	SW
Hardware	_________%	HW	Support	_________%	SUP

Total: 100%

How many of the following does your IT organization support? SUP_

Users __________ U

Software Licenses __________ SL

Sites __________ S

Distinct applications __________ DA

For questions please feel free to contact:

Ken Huang

Email: kenhuang@mit.edu

Instructions: The survey is divided into two sections: size drivers and cost drivers. Each section is designed to calibrate specific parameters that will be calculated based on the survey results we've collected.

Section I:

Size drivers

Which are the most representative of the size of an IT system/infrastructure?

(Ranking: 1 – (9-12); 1 is being the most representative, and (9-12) is being the least representative; duplicate ranking is acceptable. You may write your own types of size drivers into the three blank rows and incorporate them into the ranking.)

Types of size drivers	Ranking	Not Relevant
Number of servers or racks	RACK	
Number of software licenses	LICE	
Number of KVMs (terminal servers)	KVMS	
Number of distinct sites(data centers)	SITES	
Number of users / PC equipments	USERS	
Number of feet of FA-CL or CAT5 cables.	FEET	
Number of software applications that need to be supported (COTS and new)	APPLI	
Number of databases (i.e., human resources, sales, etc)	DBS	
Number of data & phone jacks	DATA	

Section II:

Cost Drivers

An AHP (Analytic Hierarchy Process) example

An AHP is a mathematical process involving matrices that produces a Ranking through Pair-Wise Comparison voting of competing alternatives and different Criteria. In other words, it is a decision making process for organizing and assessing alternatives against a hierarchy of multifaceted objectives.

Pair-Wise Comparison:

1 equal 3 moderate 5 strong 7 very strong 9 extreme

	Orange	Apple	Banana
Orange		1/3	3/1
Apple	3		5/1
Banana	1/3	1/5	

You would simply need to fill in the upper half of the table. The lower half of the table reflects the inverse relative importance of the upper half of the table, and it would be filled in. In other words, **Numbers in green is what you should enter by comparing its relative importance level ranging from 1, 3,5,7,or 9. The numbers in red will then later be computed based on the numbers in green. Using the table above as an example, 1/3 = (Orange / Apple), it means that having one apple is equivalent to having 3 oranges; in other words, 1 apple weighs 3 times more than 1 orange. The pair-wise comparison goes from rows against columns. On Row 1 / Column 3, 3/1 = (Orange / Banana), it means that having one orange is equivalent to having 3 banana; in other words, 1 orange weighs 3 times more than 1 banana.**

Cost drivers

Cost drivers (determine the complexity or operational environment of the system)

Please enter (1 ,3,5,7,and 9) with 1 being the lowest importance level and 9 being the highest importance level as compared to each individual parameter.

Table 1:Systems -> An independent entity or a group of entities that serves a specific function or set of functions. It ranges from a single standalone server, to the entire server infrastructure, e.g. clustered application servers.

Table 2:Hardware -> A physical, tangible electronic, electrical or mechanical component of a computer.

Table 3:Software -> A commercial, COTS (commercial off-the-shelf), GNU (GPU's Not Unix) licensed or in-house application that comprises a complete set of line of codes for a specific functionality or set of functionalities.

Table 4:Support -> A dedicated source for assistance in maintaining the systems operations. It contains the information about technical issues and troubleshooting strategies which are provided in the format of engineering manual via (online/pdf/books/voice phone line).

***definition of each cost driver can be found at Definitions Page 9 ***

Instructions: SCALE: 1 equal 3 moderate 5 strong 7 very strong 9 extreme

Please fill in the upper half of the table.

Cost driver Importance level using AHP approach

Systems	service call response time	Reimplementation / Re-design	Client/server compatibility	Security	Server Redundancy	Business Continuity	MTTR(mean time to recovery)	TCO(total cost of ownership)	SLA(service level agreement)
service call response time	SCRT	RSCRT	CS	SECS	SRS	BS	MS	TS	SLAS
Reimplementation / Re-design		RNR	CR	SECR	SRR	BR	MR	TR	SLAR
Client/server compatibility			CSC	SECC	SRC	BC	MC	TC	SLAC
Security				SEC	SRSEC	BSEC	MSEC	TSEC	SLASEC
Server Redundancy					SR	BRED	MRED	TRED	SLARED
Business Continuity						BC	MBC	TBC	SLABC
MTTR(mean time to recovery)							MTTR	TMMTR	SLAMTTR
TCO(total cost of ownership)								TCO	SLATCO
SLA(service level agreement)									SLA

For future reference: do you have any cost drivers that are not covered under Systems? If so, enter any of the blanks below

__SYS_OTHER1_____, __SYS_OTHER2____, __SYS_OTHER3__, __SYS_OTHER4__

Instructions:

SCALE: 1 equal 3 moderate 5 strong 7 very strong 9 extreme

Please fill in the upper half of the table.

Cost driver Importance level using AHP approach

Hardware	Seamless integration	Component Volatility	Component Application Complexity	Interface Complexity	Product Support	Experience with Component	Learning Rate	Reliability	Confidence Level
Seamless integration	SI	VS	CACS	IS	PS	ES	LS	RS	CLS
Component Volatility		CV	CACC	IC	PC	EC	LC	RC	CLC
Component Application Complexity			CAC	ICAC	PCAC	ECAC	LCAC	RCAC	CLCAC
Interface Complexity				IC	PIC	EIC	LIC	RIC	CLIC
Product Support					PS	EPS	LPS	RPS	CLPS
Experience with Component						EC	LEC	REC	CLEC
Learning Rate							LR	RLR	CLLR
Reliability								RELI	CLR
Confidence Level									CL

For future reference: do you have any cost drivers not covered under Hardware? If so, please enter in the blanks below:

____HARD_OTHER1____, ____HARD_OTHER2____, ____HARD_OTHER3____, ____HARD_OTHER4____

Instructions:

SCALE: 1 equal 3 moderate 5 strong 7 very strong 9 extreme

Please fill in the upper half of the table.

Cost driver Importance level using AHP approach						
Software	**Confidence Level**	**Lines of Code**	**Redesign Required**	**Retest Required**	**Reimplementation Required**	**Time Constraints**
Confidence Level	CL	LCL	RCL	RRCL	REQCL	TCL
Lines of Code		LC	RLC	RRLC	REQLC	TLC
Redesign Required			RR	RRRED	REQRED	TREDREQ
Retest Required				RETEST	REQRET	TRET
Reimplementation Required					REIM	TREIM
Time Constraints						TC

For future reference: do you have any cost drivers that are not covered under Software? If so, please enter any of the blanks below:

__SW_OTHER1____, __SW_OTHER2____, __SW_OTHER3____, __SW_OTHER4__

Instructions:

SCALE: 1 equal 3 moderate 5 strong 7 very strong 9 extreme

Please fill in the upper half of the table.

Cost driver Importance level using AHP approach

Support	Learning Rate	Professional Experience	Cost	Repairs	Call center	Upgrades
Learning Rate	LR	PLR	CLRATE	RLRATE	CCLRATE	ULRATE
Professional Experience		PE	CPE	RPE	CCPE	UPE
Cost			COST	RCOST	CCCOST	UCOST
Repairs				REPAIR	CCR	UR
Call center					CC	UCALL
Upgrades						UP

For future reference: do you have any cost drivers not covered under Software? If so, please enter in the blanks below:

____________, ________________, ______________, ________________.

We greatly appreciate your time and input for completing this survey!!

Definitions:

Systems

Service call response time - > The time required or agreed to respond a technical support ticket opened by the customer.

Reimplementation / re-design -> To re-architecture / to enhance some/entire functionalities of the systems in question.

Client/server compatibility -> The handshakes or cohesiveness of communications between its clients and the server. Would be there any hiccups at the network communication level.

Security -> The company compliance level of the "systems" in question.

Server Redundancy -> A hot-standby (disaster recovery) server for the primary server of the same functionality.

Business Continuity -> High Availability of the "systems" infrastructure overall.

MTTR (mean time to recovery) -> the average amount of time required to resolve most hardware or software problems with a given device.

TCO (total cost of ownership) -> Cost to purchase and maintain software over time.

SLA (Service level agreement) -> Formal agreement between a Service Provider and customers to provide a certain level of service. Penalty clauses might apply if the SLA is not met.

Hardware

Seamless integration -> The smoothness of the coordination between two or more hardware components.

Component volatility -> The rate of stability of the component.

Component application complexity -> The level of complexity of a component's functionality and operations.

Interface Complexity -> The level of effort to interact with another hardware component.

Product Support -> The hardware warranty provided by the hardware vendor.

Experience with Component -> The overall technical experience of the engineers handling the hardware.

Learning rate -> A measure of the technical personnel mastering the installation/replacement of the hardware in relation to some specification of time.

Reliability -> The probability of performing a specified function without failure under given conditions for a specified period of time.

Confidence level -> The level of comfort of having this hardware lives within the current system infrastructure.

Software

Confidence level -> The level of comfort of running this application lives within the current system infrastructure.

Lines of Codes -> The total number of lines of codes required to run this application.

Redesign required -> The necessity of re-organizing the layout

Retest required -> The necessity of examining the software for quality assurance purpose.

Reimplementation required -> the necessity of enhancing the functionalities of the application.

Time constraints -> The total time allowed to perform any tasks relevant to this application.

Support

Learning Rate -> A measure of the technical personnel mastering the maintenance in relation to some specification of time.

Professional Experience -> The technical expertise from the staff or the vendor technical support team to escalate all the issues that might arise.

Cost -> The annual monetary spending for maintaining the current server infrastructure.

Repairs -> The frequency rate of fixing any hardware component or software.

Call center -> The 24/7 surveillance center for monitoring any server failure and coordinating the failure to the appropriate teams.

Upgrades -> The rate of upgrading the current server infrastructure design or functionalities.

APPENDIX C: SAS SOURCE CODE FOR COMPILING MIT CIO SURVEY

```
*************************************
***** Program: MIT_Survey.SAS    *****
***** Author: Ken Huang          *****
***** Date: OCT. 08 2006         *****
*************************************;
libname MIT '.';
options ls=80 ps=60 pageno=1 nonumber nodate fmtsearch=(MIT) formdlim='';


data MIT.ReadIn;
     infile 'C:\Ken\MIT\Master\10132006\MIT_Survey.txt'
     dlm='09'x dsd truncover LRECL=1000 FIRSTOBS=2;
     input FName:$30. LName:$30. DDBU:$100. Address:$100. Email:$100.
          Contact:$8. Corp:$30. Industry:$30. Size:$30. ITExper:8.
          ExperYear:8. ITBudget:$30. BUD_SYS:8. BUD_SW:8. BUD_HW:8.
          BUD_SUP:8. SUP_U:8. SUP_SL:8. SUP_S:8. SUP_DA:8.
          Rank_Rack:8. Rank_Lice:8. Rank_KVMS:8. Rank_Sites:8.
          Rank_Users:8. Rank_Feet:8. Rank_Appli:8. Rank_DBS:8.
          Rank_Data:8.    SCRT:8. RSCRT:8. CS:8. SECS:8. SRS:8. BS:8. MS:8. TS:8. SLAS:8. RNR:8.
          CR:8. SECR:8. SRR:8. BR:8. MR:8. TR:8. SLAR:8. CSC:8. SECC:8. SRC:8. BC:8. MC:8. TC:8.
          SLAC:8. SEC:8. SRSEC:8. BSEC:8. MSEC:8. TSEC:8. SLASEC:8.
          SR:8. BRED:8. MRED:8. TRED:8. SLARED:8. BC:8. MBC:8. TBC:8.
          SLABC:8. MTTR:8. TMTTR:8. SLAMTTR:8. TCO:8. SLATCO:8.
          SLA:8. SYS_OTH:$100. SI:8. VS:8. CACS:8. IS:8. PS:8. ES:8.
          LS:8. RS:8. CLS:8. CV:8. CACC:8. IC:8. PC:8. EC:8. LC:8.
          RC:8. CLC:8. CAC:8. ICAC:8. PCAC:8. ECAC:8. LCAC:8. RCAC:8.
          CLCAC:8. IC:8. PIC:8. EIC:8. LIC:8. RIC:8. CLIC:8. PS:8.
```

```
            EPS:8. LPS:8. RPS:8. CLPS:8. EC:8. LEC:8. REC:8. CLEC:8.
            LR1:8. RLR:8. CLLR:8. RELI:8. CLR:8. CL1:8. HARD_OTH:$100.
            CL2:8. LCL:8. RCL:8. RRCL:8. REQCL:8. TCL:8. LC:8. RLC:8.
            RRLC:8. REQLC:8. TLC:8. RR:8. RRRED:8. REQRED:8. TREDREQ:8.
            RETEST:8. REQRET:8. TRET:8. REIM:8. TREIM:8. TC:8.
            SW_OTH:$100. LR2:8. PLR:8. CLRATE:8. RLRATE:8. CCLRATE:8.
            ULRATE:8. PE:8. CPE:8. RPE:8. CCPE:8. UPE:8. COST:8.
            RCOST:8. CCCOST:8. UCOST:8. REPAIR:8. CCR:8. UR:8. CC:8.
            UCALL:8. UP:8. SUP_OTH:$100.;
run;

***** Clean data sets *****;
data MIT.Survey;
      set MIT.ReadIn;

      ID=FName||LName;

***** Create new variable for Table 1 ~ 4 *****;
      array x[*] RSCRT CS CR SecS SecR SecC SRS SRR SRC SRSec BS BR
            BC BSec BRed MS MR MC MSec MRed MBC TS TR TC TSec
            TRed TBC TMTTR SLAS SLAR SLAC SLASec SLARed SLABC
            SLAMTTR SLATCO VS CACS CACC IS IC ICAC PS PC PCAC PIC
            ES EC ECAC EIC EPS LS LC LCAC LIC LPS LEC RS RC RCAC
            RIC RPS REC RLR CLS CLC CLCAC CLIC CLPS CLEC CLLR CLR
            LCL RCL RLC RRCL RRLC RRRed ReqCL ReqLC ReqRed ReqRet
            TCL TLC TRedReq TRet TReim PLR CLRate CPE RLRate RPE
            RCost CCLRate CCPE CCCost CCR ULRate UPE UCost UR UCall;
      array y[*] RSCRT_R CS_R CR_R SecS_R SecR_R SecC_R SRS_R SRR_R
```

```
                SRC_R SRSec_R BS_R BR_R BC_R BSec_R BRed_R MS_R
                MR_R MC_R MSec_R MRed_R MBC_R TS_R TR_R TC_R TSec_R
                TRed_R TBC_R TMTTR_R SLAS_R SLAR_R SLAC_R SLASec_R
                SLARed_R SLABC_R SLAMTTR_R SLATCO_R VS_R CACS_R
                CACC_R IS_R IC_R ICAC_R PS_R PC_R PCAC_R PIC_R ES_R
                EC_R ECAC_R EIC_R EPS_R LS_R LC_R LCAC_R LIC_R LPS_R
                LEC_R RS_R RC_R RCAC_R RIC_R RPS_R REC_R RLR_R CLS_R
                CLC_R CLCAC_R CLIC_R CLPS_R CLEC_R CLLR_R CLR_R LCL_R
                RCL_R RLC_R RRCL_R RRLC_R RRRed_R ReqCL_R ReqLC_R
                ReqRed_R ReqRet_R TCL_R TLC_R TRedReq_R TRet_R
                TReim_R PLR_R CLRate_R CPE_R RLRate_R RPE_R RCost_R
                CCLRate_R CCPE_R CCCost_R CCR_R ULRate_R UPE_R UCost_R UR_R UCall_R;
        Do I=1 to dim(X);
                Y[I]=1/X[I];
                drop i;
        end;
run;

***** Get average of Table 1 - 4 *****
***** Create Base for it         *****;
proc freq data=MIT.Survey noprint;
        table ID / out=ID (drop=percent);
run;
proc summary data=ID;
        var Count;
        output out=MIT.Average(drop=_type_ _freq_) sum=Count;
run;
```

```
%macro Ave(Var);
proc univariate data=Mit.Survey noprint;
        var &var;
        output out=&var.new mean=&var.m;
run;
data MIT.Average;
        merge MIT.Average &var.new;
run;
%mend;
%Ave(RSCRT) %Ave(CS) %Ave(CR) %Ave(SecS) %Ave(SecR) %Ave(SecC) %Ave(SRS)
%Ave(SRR) %Ave(SRC) %Ave(SRSec) %Ave(BS) %Ave(BR) %Ave(BC) %Ave(BSec)
%Ave(BRed) %Ave(MS) %Ave(MR) %Ave(MC) %Ave(MSec) %Ave(MRed) %Ave(MBC)
%Ave(TS) %Ave(TR) %Ave(TC) %Ave(TSec) %Ave(TRed) %Ave(TBC) %Ave(TMTTR)
%Ave(SLAS) %Ave(SLAR) %Ave(SLAC) %Ave(SLASec) %Ave(SLARed) %Ave(SLABC)
%Ave(SLAMTTR) %Ave(SLATCO) %Ave(RSCRT_R) %Ave(CS_R) %Ave(CR_R) %Ave(SecS_R)
%Ave(SecR_R) %Ave(SecC_R) %Ave(SRS_R) %Ave(SRR_R)
%Ave(SRC_R) %Ave(SRSec_R) %Ave(BS_R) %Ave(BR_R) %Ave(BC_R) %Ave(BSec_R)
%Ave(BRed_R) %Ave(MS_R) %Ave(MR_R) %Ave(MC_R) %Ave(MSec_R) %Ave(MRed_R)
%Ave(MBC_R) %Ave(TS_R) %Ave(TR_R) %Ave(TC_R) %Ave(TSec_R) %Ave(TRed_R)
%Ave(TBC_R) %Ave(TMTTR_R) %Ave(SLAS_R) %Ave(SLAR_R) %Ave(SLAC_R)
%Ave(SLASec_R) %Ave(SLARed_R) %Ave(SLABC_R) %Ave(SLAMTTR_R)
%Ave(SLATCO_R) %Ave(VS) %Ave(CACS) %Ave(CACC) %Ave(IS) %Ave(IC)
%Ave(ICAC) %Ave(PS) %Ave(PC) %Ave(PCAC) %Ave(PIC) %Ave(ES) %Ave(EC)
%Ave(ECAC) %Ave(EIC) %Ave(EPS) %Ave(LS) %Ave(LC) %Ave(LCAC) %Ave(LIC)
%Ave(LPS) %Ave(LEC) %Ave(RS) %Ave(RC) %Ave(RCAC) %Ave(RIC) %Ave(RPS)
%Ave(REC) %Ave(RLR) %Ave(CLS) %Ave(CLC) %Ave(CLCAC) %Ave(CLIC)
%Ave(CLPS) %Ave(CLEC) %Ave(CLLR) %Ave(CLR) %Ave(VS_R) %Ave(CACS_R)
%Ave(CACC_R) %Ave(IS_R) %Ave(IC_R) %Ave(ICAC_R) %Ave(PS_R) %Ave(PC_R)
```

```
%Ave(PCAC_R) %Ave(PIC_R) %Ave(ES_R) %Ave(EC_R) %Ave(ECAC_R) %Ave(EIC_R)
%Ave(EPS_R) %Ave(LS_R) %Ave(LC_R) %Ave(LCAC_R) %Ave(LIC_R) %Ave(LPS_R)
%Ave(LEC_R) %Ave(RS_R) %Ave(RC_R) %Ave(RCAC_R) %Ave(RIC_R) %Ave(RPS_R)
%Ave(REC_R) %Ave(RLR_R) %Ave(CLS_R) %Ave(CLC_R) %Ave(CLCAC_R) %Ave(CLIC_R) %Ave(CLPS_R)
%Ave(CLEC_R) %Ave(CLLR_R) %Ave(CLR_R)
%Ave(LCL) %Ave(RCL) %Ave(RLC) %Ave(RRCL) %Ave(RRLC) %Ave(RRRed)
%Ave(ReqCL) %Ave(ReqLC) %Ave(ReqRed) %Ave(ReqRet) %Ave(TCL)
%Ave(TLC) %Ave(TRedReq) %Ave(TRet) %Ave(TReim) %Ave(LCL_R) %Ave(RCL_R)
%Ave(RLC_R) %Ave(RRCL_R) %Ave(RRLC_R) %Ave(RRRed_R) %Ave(ReqCL_R)
%Ave(ReqLC_R) %Ave(ReqRed_R) %Ave(ReqRet_R) %Ave(TCL_R) %Ave(TLC_R)
%Ave(TRedReq_R) %Ave(TRet_R) %Ave(TReim_R) %Ave(PLR) %Ave(CLRate)
%Ave(CPE) %Ave(RLRate) %Ave(RPE) %Ave(RCost) %Ave(CCLRate) %Ave(CCPE)
%Ave(CCCost) %Ave(CCR) %Ave(ULRate) %Ave(UPE) %Ave(UCost) %Ave(UR)
%Ave(UCall) %Ave(PLR_R) %Ave(CLRate_R) %Ave(CPE_R) %Ave(RLRate_R)
%Ave(RPE_R) %Ave(RCost_R) %Ave(CCLRate_R) %Ave(CCPE_R) %Ave(CCCost_R)
%Ave(CCR_R) %Ave(ULRate_R) %Ave(UPE_R) %Ave(UCost_R) %Ave(UR_R)
%Ave(UCall_R);

ods listing close;
ods rtf file='Average.doc' path='C:\Ken\MIT\Result\10132006';
proc print data=MIT.Average;
	title 'Average of all the variables from Table 1 ~ 4';
run;
ods rtf close;
ods listing;

ods listing close;
ods rtf file='EigenValue.doc' path='C:\Ken\MIT\Result\10132006';
```

```
***** Get the eigenvalue of each table *****
***** Table 1 *****;
Proc IML;
T1={1.0000 1.4699 2.4000 2.5757 4.0667 2.6952 3.6667 2.4967 2.7633,
  3.7549 1.0000 3.8027 2.9429 6.0286 3.7347 3.7619 4.4286 2.4966,
  1.9683 2.5178 1.0000 2.9729 2.7633 1.0000 3.2585 1.0000 2.7905,
  3.7183 2.3696 2.0837 1.0000 4.1429 3.1714 4.3333 4.4286 3.0952,
  1.8995 0.8440 2.1788 0.5329 1.0000 2.5048 5.2857 4.1429 5.1905,
  1.7919 1.3072 1.0000 1.1252 1.5252 1.0000 5.0000 4.4286 4.4286,
  1.1271 0.7397 1.9776 0.7125 0.3950 0.3125 1.0000 3.2857 4.6190,
  1.8106 0.2490 1.0000 0.3252 0.3442 0.3252 0.4667 1.0000 4.6000,
  2.0918 1.7236 1.4191 1.1253 1.4272 0.3361 0.6935 0.9138 1.0000};

        TT1=T1*T1;
        create tempTT1 from TT1;
        append from TT1;

        TTTT1=TT1*TT1;
        create tempTTTT1 from TTTT1;
        append from TTTT1;

        T8_1=TTTT1*TTTT1;
        create tempT8_1 from T8_1;
        append from T8_1;

        T16_1=T8_1*T8_1;
        create tempT16_1 from T16_1;
        append from T16_1;
```

```
        print TT1 TTTT1 T8_1 T16_1;


Quit;


%macro NewFile(file, title=);
data &file;
        set &file;
        Total=Sum(of col1 col2 col3 col4 col5 col6 col7 col8 col9);
        retain ID 0;
        ID+1;
run;
proc tabulate data=&file;
        class ID;
        var Total;
        table ID All, Total*(sum pctsum);
        title "&Title";
run;
%mend;
%NewFile(TempTT1, title=Summary from Table 1 Matrix 1)
%NewFile(TempTTTT1, title=Summary from Table 1 Matrix 2)
%NewFile(TempT8_1, title=Summary from Table 1 Matrix 3)
%NewFile(TempT16_1, title=Summary from Table 1 Matrix 4);


***** Table 2 *****;
Proc IML;
T2={1.0000 3.6667 4.7778 3.7238 1.0000 3.8667 2.4222 3.8572 3.7143,
   1.5758 1.0000 2.6222 1.0000 3.8667 1.0000 1.0000 5.2000 4.1111,
```

```
1.1038 2.3810 1.0000 2.7090 3.7556 3.5556 2.6667 4.2000 3.5556,
2.2420 1.0000 2.9331 1.0000 3.6400 3.0889 2.6444 5.2000 4.1111,
1.0000 1.2423 1.5757 1.3289 1.0000 2.8889 3.5556 4.2000 3.8889,
1.2423 1.0000 0.8128 1.5905 1.0477 1.0000 3.6667 3.3852 3.4000,
1.6223 1.0000 1.6129 1.9461 0.9143 0.5810 1.0000 2.8889 2.3333,
1.5753 0.9884 1.0296 0.9757 1.1143 2.4922 0.9461 1.0000 5.6667,
2.5750 1.1313 0.9143 1.1313 0.8032 0.5352 0.7333 0.4222 1.0000};

    TT2=T2*T2;
    create tempTT2 from TT2;
    append from TT2;

    TTTT2=TT2*TT2;
    create tempTTTT2 from TTTT2;
    append from TTTT2;

    T8_2=TTTT2*TTTT2;
    create tempT8_2 from T8_2;
    append from T8_2;

    T16_2=T8_2*T8_2;
    create tempT16_2 from T16_2;
    append from T16_2;

    print TT2 TTTT2 T8_2 T16_2;

Quit;
```

```
%macro NewFile(file, title=);
data &file;
        set &file;
        Total=Sum(of col1 col2 col3 col4 col5 col6 col7 col8 col9);
        retain ID 0;
        ID+1;
run;
proc tabulate data=&file;
        class ID;
        var Total;
        table ID All, Total*(sum pctsum);
        title "&Title";
run;
%mend;
%NewFile(TempTT2, title=Summary from Table 2 Matrix 1)
%NewFile(TempTTTT2, title=Summary from Table 2 Matrix 2)
%NewFile(TempT8_2, title=Summary from Table 2 Matrix 3)
%NewFile(TempT16_2, title=Summary from Table 2 Matrix 4);

***** Table 3 *****;
Proc IML;
T3={1.0000 2.7778 2.8572 4.2889 2.2646 3.2286,
   1.1779 1.0000 2.9265 3.7460 2.6281 3.8934,
   1.6124 2.9259 1.0000 3.1111 2.6635 3.3302,
   2.4860 1.8029 1.1557 1.0000 2.3778 2.9220,
   3.3903 3.4295 2.6492 2.6682 1.0000 4.2000,
   1.7348 2.5218 2.2599 3.2121 1.2328 1.0000};
```

```
        TT3=T3*T3;
        create tempTT3 from TT3;
        append from TT3;

        TTTT3=TT3*TT3;
        create tempTTTT3 from TTTT3;
        append from TTTT3;

        T8_3=TTTT3*TTTT3;
        create tempT8_3 from T8_3;
        append from T8_3;

        T16_3=T8_3*T8_3;
        create tempT16_3 from T16_3;
        append from T16_3;

        print TT3 TTTT3 T8_3 T16_3;

Quit;

%macro NewFile(file, title=);
data &file;
        set &file;
        Total=Sum(of col1 col2 col3 col4 col5 col6);
        retain ID 0;
        ID+1;
run;
proc tabulate data=&file;
```

```
        class ID;
        var Total;
        table ID All, Total*(sum pctsum);
        title "&Title";
run;
%mend;
%NewFile(TempTT3, title=Summary from Table 3 Matrix 1)
%NewFile(TempTTTT3, title=Summary from Table 3 Matrix 2)
%NewFile(TempT8_3, title=Summary from Table 3 Matrix 3)
%NewFile(TempT16_3, title=Summary from Table 3 Matrix 4);

***** Table 4 *****;
Proc IML;
T4={1.0000 3.6077 2.3333 4.2109 3.6349 3.8889,
  2.4502 1.0000 2.5238 3.3619 2.6553 4.4400,
  0.9620 1.2395 1.0000 4.5556 2.4730 3.4000,
  1.5657 1.3266 0.7757 1.0000 2.8730 3.6667,
  2.2379 2.9346 2.3824 1.7158 1.0000 4.6000,
  0.9048 1.1638 0.5352 0.5810 0.4908 1.0000};

        TT4=T4*T4;
        create tempTT4 from TT4;
        append from TT4;

        TTTT4=TT4*TT4;
        create tempTTTT4 from TTTT4;
        append from TTTT4;
```

```
T8_4=TTTT4*TTTT4;
create tempT8_4 from T8_4;
append from T8_4;

T16_4=T8_4*T8_4;
create tempT16_4 from T16_4;
append from T16_4;

print TT4 TTTT4 T8_4 T16_4;

Quit;

%macro NewFile(file, title=);
data &file;
set &file;
Total=Sum(of col1 col2 col3 col4 col5 col6);
retain ID 0;
ID+1;
run;
proc tabulate data=&file;
class ID;
var Total;
table ID All, Total*(sum pctsum);
title "&Title";
run;
%mend;
%NewFile(TempTT4, title=Summary from Table 4 Matrix 1)
%NewFile(TempTTTT4, title=Summary from Table 4 Matrix 2)
```

```
%NewFile(TempT8_4, title=Summary from Table 4 Matrix 3)

%NewFile(TempT16_4, title=Summary from Table 4 Matrix 4);

ods rtf close;

ods listing;


***** Labels *****;

data MIT.SurveyLabel;

        set MIT.Survey;


        /*

        format  RSCRT CS CR SecS SecR SecC SRS SRR SRC SRSec BS BR BC

        BSec BRed MS MR MC MSec MRed MBC TS TR TC TSec TRed TBC

        TMTTR SLAS SLAR SLAC SLASec SLARed SLABC SLAMTTR SLATCO

        VS CACS CACC IS IC ICAC PS PC PCAC PIC ES EC ECAC EIC

        EPS LS LC LCAC LIC LPS LEC RS RC RCAC RIC RPS REC RLR

        CLS CLC CLCAC CLIC CLPS      CLEC CLLR CLR LCL RCL RLC RRCL

        RRLC RRRed ReqCL ReqLC ReqRed ReqRet TCL TLC TRedReq

        TRet TReim PLR CLRate CPE RLRate RPE RCost CCLRate CCPE

        CCCost CCR ULRate UPE UCost UR UCall fscale.

        RSCRT_R CS_R CR_R SecS_R SecR_R SecC_R SRS_R SRR_R SRC_R

        SRSec_R BS_R BR_R BC_R BSec_R BRed_R MS_R MR_R MC_R

        MSec_R MRed_R MBC_R TS_R TR_R TC_R TSec_R TRed_R TBC_R

        TMTTR_R SLAS_R SLAR_R SLAC_R SLASec_R SLARed_R SLABC_R

        SLAMTTR_R SLATCO_R VS_R CACS_R CACC_R IS_R IC_R ICAC_R

        PS_R PC_R PCAC_R PIC_R ES_R EC_R ECAC_R EIC_R EPS_R LS_R

        LC_R LCAC_R LIC_R LPS_R LEC_R RS_R RC_R RCAC_R RIC_R

        RPS_R REC_R RLR_R CLS_R CLC_R CLCAC_R CLIC_R CLPS_R

        CLEC_R CLLR_R CLR_R LCL_R RCL_R RLC_R RRCL_R RRLC_R
```

```
RRRed_R ReqCL_R ReqLC_R ReqRed_R ReqRet_R TCL_R TLC_R
TRedReq_R TRet_R TReim_R PLR_R CLRate_R CPE_R RLRate_R
RPE_R RCost_R CCLRate_R CCPE_R CCCost_R CCR_R ULRate_R
UPE_R UCost_R UR_R UCall_R fscale.;
*/

label FName='First Name'
    LName='Last Name'
    DDBU='Division/Department/Business Unit'
    Address='Location'
    EMail='Email address'
    Contact='May we contact you for follow-up?'
    Corp='Name of the corporation'
    Industry='Industry'
    Size='Company size'
    ITExper='Years of experience in IT'
    ExperYear='Total number of years of your work experience'
    ITBudget='Approximate annual IT budget of your division or business unit'
    BUD_SYS='System'
    BUD_SW='Software'
    BUD_HW='Hardware'
    BUD_SUP='Support'
    SUP_U='Users'
    SUP_SL='Software licenses'
    SUP_S='Sites'
    SUP_DA='Distinct applications'
    Rank_Rack='Number of servers or racks'
    Rank_Lice='Number of software licenses'
```

Rank_KVMS='Number of KVMS(terminal servers)'

Rank_Sites='Number of distince sites(data centers)'

Rank_Users='Number of users/PC equipments'

Rank_Feet='Number of feet of FA-CL or CAT5 cables'

Rank_Appli='Number of software applications that need to be supported'

Rank_DBS='Number of databases'

Rank_Data='Number of data & phone jacks'

SCRT='Service call response time'

RSCRT='Reimplementation/Re-design vs Service call response time'

RNR='Reimplementation/Re-design'

CS='Client/Server compatibility vs Service call response time'

CR='Client/Server compatibility vs Reimplementation/Re-design'

CSC='Client/Server compatibility'

SecS='Security vs Service call response time'

SecR='Security vs Reimplementation/Re-design'

SecC='Security vs Client/Server compatibility'

SEC='Security'

SRS='Server redundancy vs Service call response time'

SRR='Server redundancy vs Reimplementation/Re-design'

SRC='Server redundancy vs Client/Server compatibility'

SRSec='Server redundancy vs Security'

SR='Server redundancy'

BS='Business continuity vs Service call response time'

BR='Business continuity vs Reimplementation/Re-design'

BC='Business continuity vs Client/Server compatibility'

BSec='Business continuity vs Security'

BRed='Business continuity vs Server redundancy'

BC='Business continuity'

MS='Meantime to recovery vs Service call response time'

MR='Meantime to recovery vs Reimplementation/Re-design'

MC='Meantime to recovery vs Client/Server compatibility'

MSec='Meantime to recovery vs Security'

MRed='Meantime to recovery vs Server redundancy'

MBC='Meantime to recovery vs Business continuity'

MTTR='Meantime to recovery'

TS='Total cost of ownership vs Service call response time'

TR='Total cost of ownership vs Reimplementation/Re-design'

TC='Total cost of ownership vs Client/Server compatibility'

TSec='Total cost of ownership vs Security'

TRed='Total cost of ownership vs Server redundancy'

TBC='Total cost of ownership vs Business continuity'

TMTTR='Total cost of ownership vs Meantime to recovery'

TCO='Total cost of ownership'

SLAS='Service level agreement vs Service call response time'

SLAR='Service level agreement vs Reimplementation/Re-design'

SLAC='Service level agreement vs Client/Server compatibility'

SLASec='Service level agreement vs Security'

SLARed='Service level agreement vs Server redundancy'

SLABC='Service level agreement vs Business continuity'

SLAMTTR='Service level agreement vs Meantime to recovery'

SLATCO='Service level agreement vs Total cost of ownership'

SLA='Service level agreement'

Sys_Oth='Other cost drivers under system'

SI='Seamless integration'

VS='Component volatility vs Seamless integration'

CV='Component volatility'

CACS='Component application complexity vs Seamless integration'

CACC='Component application complexity vs Component volatility'

CAC='Component application complexity'

IS='Interface complexity vs Seamless integration'

IC='Interface complexity vs Component volatility'

ICAC='Interface complexity vs Component application complexity'

IC='Interface complexity'

PS='Product support vs Seamless integration'

PC='Product support vs Component volatility'

PCAC='Product support vs Component application complexity'

PIC='Product support vs Interface complexity'

PS='Product support'

ES='Experience with component vs Seamless integration'

EC='Experience with component vs Component volatility'

ECAC='Experience with component vs Component application complexity'

EIC='Experience with component vs Interface complexity'

EPS='Experience with component vs Product support'

EC='Experience with component'

LS='Learning rate vs Seamless integration'

LC='Learning rate vs Component volatility'

LCAC='Learning rate vs Component application complexity'

LIC='Learning rate vs Interface complexity'

LPS='Learning rate vs Product support'

LEC='Learning rate vs Experience with component'

LR1='Learning rate'

RS='Reliability vs Seamless integration'

RC='Reliability vs Component volatility'

RCAC='Reliability vs Component application complexity'

RIC='Reliability vs Interface complexity'

RPS='Reliability vs Product support'

REC='Reliability vs Experience with component'

RLR='Reliability vs Learning rate'

RELI='Reliability'

CLS='Confidence level vs Seamless integration'

CLC='Confidence level vs Component volatility'

CLCAC='Confidence level vs Component application c
omplexity'

CLIC='Confidence level vs Interface complexity'

CLPS='Confidence level vs Product support'

CLEC='Confidence level vs Experience with component'

CLLR='Confidence level vs Learning rate'

CLR='Confidence level vs Reliability'

CL1='Confidence level'

Hard_Oth='Other cost drivers under hardware'

CL2='Confidence level'

LCL='Lines of code vs Confidence level'

LC='Lines of code'

RCL='Redesign required vs Confidence level'

RLC='Redesign required vs Lines of code'

RR='Redesign required'

RRCL='Retest required vs Confidence level'

RRLC='Retest required vs Lines of code'

RRRed='Retest required vs Redesign required'

RETEST='Retest required'

ReqCL='Reimplementation required vs Confidence level'

ReqLC='Reimplementation required vs Lines of code'

ReqRed='Reimplementation required vs Redesign required'

ReqRet='Reimplementation required vs Retest required'

REIM='Reimplementation required'

TCL='Time constraints vs Confidence level'

TLC='Time constraints vs Lines of code'

TRedReq='Time constraints vs Redesign required'

TRet='Time constraints vs Retest required'

TReim='Time constraints vs Reimplementation required'

TC='Time constraints'

SW_Oth='Other cost drivers under software'

LR2='Learning rate'

PLR='Professional experience vs Learning rate'

PE='Professional experience'

CLRate='Cost vs Learning rate'

CPE='Cost vs Professional experience'

COST='Cost'

RLRate='Repairs vs Learning rate'

RPE='Repairs vs Professional experience'

RCost='Repairs vs Cost'

REPAIR='Repairs'

CCLRate='Call center vs Learning rate'

CCPE='Call center vs Professional experience'

CCCost='Call center vs Cost'

CCR='Call center vs Repairs'

CC='Call center'

ULRate='Upgrades vs Learning rate'

UPE='Upgrades vs Professional experience'

UCost='Upgrades vs Cost'

```
                    UR='Upgrades vs Repairs'
                    UCall='Upgrades vs Call'
                    UP='Upgrades'
                    Sup_Oth='Other cost drivers under support';
run;

ods listing close;
ods rtf file='MIT Survey.doc' path='C:\Ken\MIT\Result\10132006';
proc print data=mit.surveylabel label;
          title 'MIT Survey - Ken Huang';
run;
ods rtf close;
ods listing;

ods listing close;
ods rtf file='MIT Frequency.doc' path='C:\Ken\MIT\Result\10132006';
proc freq data=MIT.Surveylabel;
          table Size itbudget;
          title 'Frequency output';
run;

proc means data=MIT.Surveylabel;
          var experyear itexper BUD_SYS BUD_SW BUD_HW BUD_SUP
            SUP_U SUP_SL SUP_S SUP_DA Rank_Rack Rank_Lice Rank_KVMS
            Rank_Sites Rank_Users Rank_Feet Rank_Appli Rank_DBS Rank_Data;
          title 'Mean Output';
run;
ods rtf close;
```

```
ods listing;


ods html body='Size.htm';

ods listing close;

ods rtf file='MIT Graphs.doc' path='C:\Ken\MIT\Result\10132006';

proc gchart data=mit.surveylabel;

  vbar3d size itexper experyear itbudget;

  title 'Graphs';

run;

ods rtf close;

ods listing;

ods html close;

quit;


***** Pie Chart *****;

goptions colors=(blue yellow);


proc gchart data=mit.surveylabel;

  pie3d industry /noheading ctext=black percent=outside

                                                    slice=inside;

            title 'Testing title';

run;

quit;
```

APPENDIX D: DETAILS OF PEARSON COEFFICIENTS FOR THE FOUR COST DRIVERS

#1 Re-implementation/Redesign (Table 1 –Systems)

8 Variables:	RSCRT_R CR SECR SRR BR MR TR SLAR

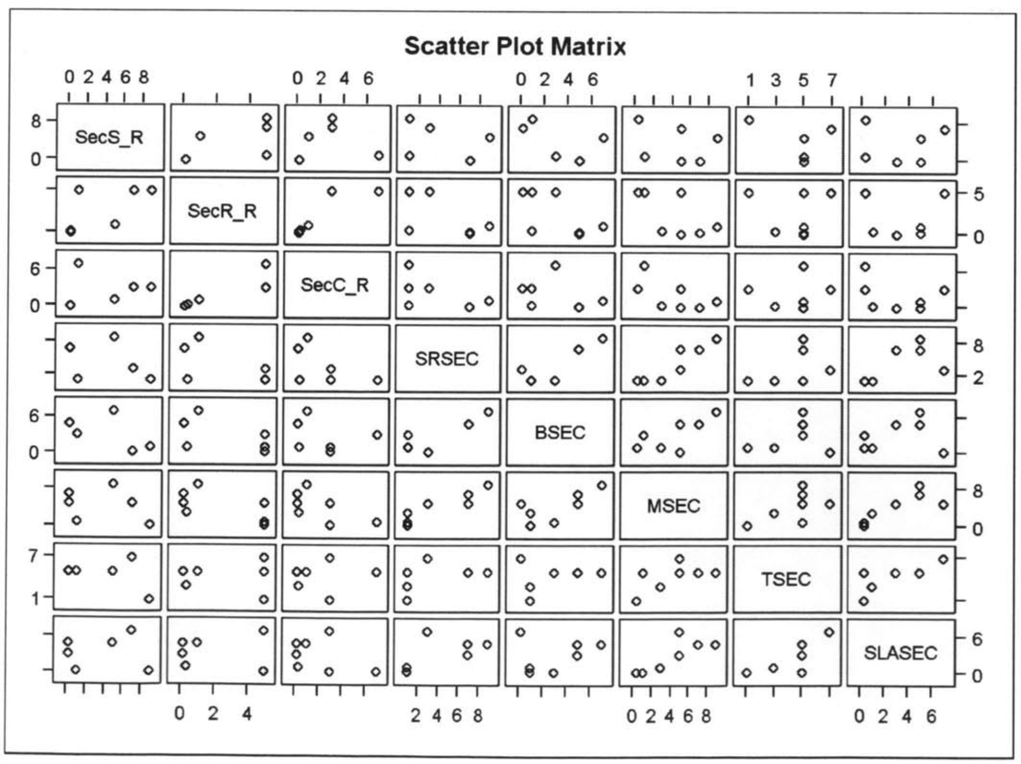
Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
RSCRT_R	6	3.75491	3.08215	22.52943	0.20000	6.99790	Reimplementation/Re-design vs. Service call response time
CR	7	3.80273	3.43819	26.61910	0.14290	7.00000	Reimplementation/Re-design vs. Client/Server compatibility
SECR	7	2.94286	3.63999	20.60000	0.20000	9.00000	Security vs. Reimplementation/Re-design
SRR	7	6.02857	3.04506	42.20000	0.20000	9.00000	Reimplementation/Re-design vs. Server redundancy
BR	7	3.73470	2.71982	26.14290	0.14290	7.00000	Reimplementation/Re-design vs. Business continuity
MR	7	3.76190	2.91684	26.33330	0.33330	9.00000	Reimplementation/Re-design vs. Meantime to recovery
TR	7	4.42857	1.51186	31.00000	3.00000	7.00000	Reimplementation/Re-design vs. Total cost of ownership
SLAR	7	2.49660	2.06157	17.47620	0.14290	5.00000	Reimplementation/Re-design vs. Service level agreement

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations								
	RSCRT_R	CR	SECR	SRR	BR	MR	TR	SLAR
RSCRT_R	1.00000	-0.61584	-0.84553	-0.38491	-0.04178	0.38257	-0.50944	-0.02241
Reimplementation/Re-design vs Service call		0.1930	0.0340	0.4511	0.9374	0.4541	0.3019	0.9664
response time	6	6	6	6	6	6	6	6
CR	-0.61584	1.00000	0.56335	-0.03440	0.31974	-0.17138	0.32892	-0.23357
Reimplementation/Re-design vs. Client/Server	0.1930		0.1879	0.9416	0.4845	0.7133	0.4713	0.6142
compatibility	6	7	7	7	7	7	7	7
SECR	-0.84553	0.56335	1.00000	0.22031	0.23390	-0.02870	0.57456	0.38642
Reimplementation/Re-design vs. Security	0.0340	0.1879		0.6350	0.6137	0.9513	0.1773	0.3918
	6	7	7	7	7	7	7	7
SRR	-0.38491	-0.03440	0.22031	1.00000	0.41102	0.22231	0.35168	0.11140
Reimplementation/Re-design vs. Server	0.4511	0.9416	0.6350		0.3597	0.6318	0.4392	0.8120
redundancy	6	7	7	7	7	7	7	7
BR	-0.04178	0.31974	0.23390	0.41102	1.00000	0.44990	0.35072	0.28643
Reimplementation/Re-design vs. Business	0.9374	0.4845	0.6137	0.3597		0.3111	0.4405	0.5334
continuity	6	7	7	7	7	7	7	7
MR	0.38257	-0.17138	-0.02870	0.22231	0.44990	1.00000	0.61911	0.76601
Reimplementation/Re-design vs. Meantime to	0.4541	0.7133	0.9513	0.6318	0.3111		0.1382	0.0446
recovery	6	7	7	7	7	7	7	7
TR	-0.50944	0.32892	0.57456	0.35168	0.35072	0.61911	1.00000	0.69698
Reimplementation/Re-design vs. Total cost of	0.3019	0.4713	0.1773	0.4392	0.4405	0.1382		0.0818
ownership	6	7	7	7	7	7	7	7
SLAR	-0.02241	-0.23357	0.38642	0.11140	0.28643	0.76601	0.69698	1.00000
Reimplementation/Re-design vs. Service level	0.9664	0.6142	0.3918	0.8120	0.5334	0.0446	0.0818	
agreement	6	7	7	7	7	7	7	7

#2 Security (Table 1 –Systems)

8 Variables:	SecS_R SecR_R SecC_R SRSEC BSEC MSEC TSEC SLASEC

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
SecS_R	6	3.71832	3.82319	22.30991	0.11111	9.00090	Security vs. Service call response time
SecR_R	7	2.36961	2.47785	16.58730	0.11111	5.00000	Security vs. Reimplementation/Re-design
SecC_R	7	2.08369	2.50752	14.58580	0.11111	6.99790	Security vs. Client/Server compatibility
SRSEC	7	4.14286	3.43650	29.00000	1.00000	9.00000	Security vs. Server redundancy
BSEC	7	3.17143	2.57016	22.20000	0.20000	7.00000	Security Business continuity
MSEC	7	4.33333	3.12695	30.33330	0.33330	9.00000	Security vs. Meantime to recovery
TSEC	7	4.42857	1.90238	31.00000	1.00000	7.00000	Security vs. Total cost of ownership
SLASEC	7	3.09523	2.65076	21.66660	0.33330	7.00000	Security vs. Service level agreement



Pearson Correlation Coefficients

Prob > |r| under H0: Rho=0

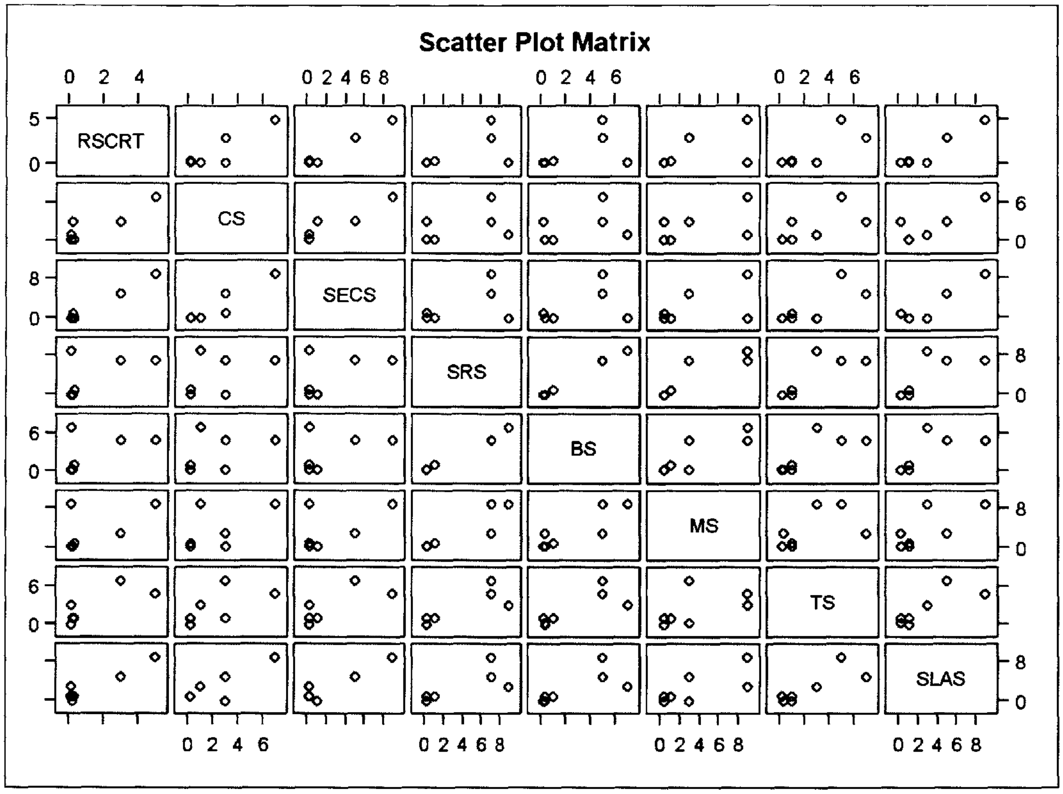
Number of Observations

	SecS_R	SecR_R	SecC_R	SRSEC	BSEC	MSEC	TSEC	SLASEC
SecS_R	1.00000	0.61202	0.14101	-0.39393	-0.57807	-0.24395	-0.38764	0.03473
Security vs Service call response time		0.1966	0.7899	0.4397	0.2295	0.6413	0.4477	0.9479
	6	6	6	6	6	6	6	6
SecR_R	0.61202	1.00000	0.84694	-0.63823	-0.60198	-0.61641	-0.04201	-0.16817
Security vs Reimplementation/Re-design	0.1966		0.0162	0.1229	0.1527	0.1404	0.9287	0.7185
	6	7	7	7	7	7	7	7
SecC_R	0.14101	0.84694	1.00000	-0.58637	-0.33776	-0.60899	0.05826	-0.36185
Security vs Client/Server compatibility	0.7899	0.0162		0.1665	0.4587	0.1467	0.9013	0.4251
	6	7	7	7	7	7	7	7
SRSEC	-0.39393	-0.63823	-0.58637	1.00000	0.87988	0.90992	0.42247	0.62033
Security vs. Server redundancy	0.4397	0.1229	0.1665		0.0090	0.0045	0.3450	0.1372
	6	7	7	7	7	7	7	7
BSEC	-0.57807	-0.60198	-0.33776	0.87988	1.00000	0.70786	0.24153	0.24510
Security vs. Business continuity	0.2295	0.1527	0.4587	0.0090		0.0751	0.6018	0.5963
	6	7	7	7	7	7	7	7
MSEC	-0.24395	-0.61641	-0.60899	0.90992	0.70786	1.00000	0.56035	0.79537
Security vs. Meantime to recovery	0.6413	0.1404	0.1467	0.0045	0.0751		0.1907	0.0325
	6	7	7	7	7	7	7	7
TSEC	-0.38764	-0.04201	0.05826	0.42247	0.24153	0.56035	1.00000	0.76174
Security vs. Total cost of ownership	0.4477	0.9287	0.9013	0.3450	0.6018	0.1907		0.0466
	6	7	7	7	7	7	7	7
SLASEC	0.03473	-0.16817	-0.36185	0.62033	0.24510	0.79537	0.76174	1.00000
Security vs. Service level agreement	0.9479	0.7185	0.4251	0.1372	0.5963	0.0325	0.0466	
	6	7	7	7	7	7	7	7

#3 Service call response time (Table 1 –Systems)

8 Variables:	RSCRT CS SECS SRS BS MS TS SLAS

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
RSCRT	6	1.46985	2.06054	8.81910	0.14290	5.00000	Service call response time vs. Reimplementation/Re-design
CS	6	2.40000	2.58921	14.40000	0.20000	7.00000	Service call response time vs. Client/Server compatibility
SECS	6	2.57567	3.66793	15.45400	0.11110	9.00000	Service call response time vs. Security
SRS	6	4.06667	4.02128	24.40000	0.20000	9.00000	Service call response time vs. Server redundancy
BS	7	2.69523	2.86974	18.86660	0.20000	7.00000	Service call response time vs. Business continuity
MS	7	3.66666	3.81033	25.66660	0.33330	9.00000	Service call response time vs. Meantime to recovery
TS	7	2.49660	2.62997	17.47620	0.14290	7.00000	Service call response time vs. Total cost of ownership
SLAS	7	2.76327	3.25604	19.34290	0.14290	9.00000	Service call response time vs. Service level agreement


Scatter Plot Matrix
RSCRT
CS
SECS
SRS
BS
MS
TS
SLAS

Pearson Correlation Coefficients

Prob > |r| under H0: Rho=0

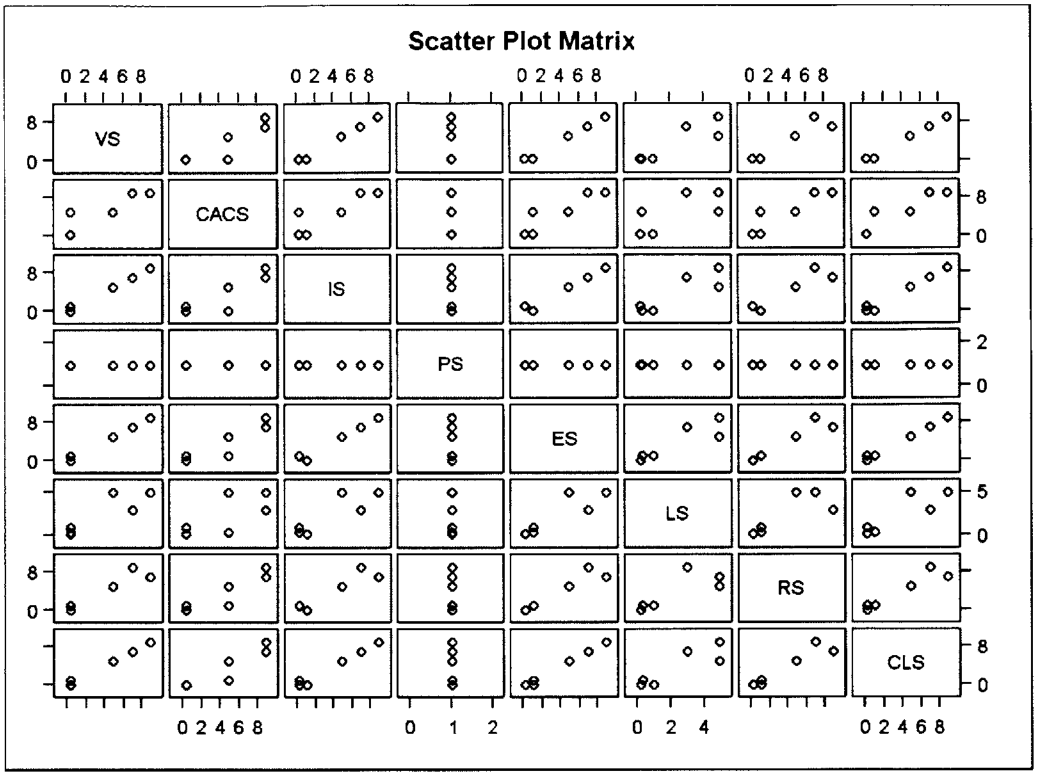
Number of Observations

	RSCRT	**CS**	**SECS**	**SRS**	**BS**	**MS**	**TS**	**SLAS**
RSCRT	1.00000	0.88741	0.99421	0.52676	0.46985	0.52285	0.78758	0.95165
Service call response time vs. Reimplementation/Re-design		0.0183	<.0001	0.2829	0.3471	0.2872	0.0629	0.0035
	6	6	6	6	6	6	6	6
CS	0.88741	1.00000	0.93000	0.40876	0.35300	0.51724	0.60665	0.82868
Service call response time vs. Client/Server compatibility	0.0183		0.0072	0.4210	0.4925	0.2933	0.2017	0.0415
	6	6	6	6	6	6	6	6
SECS	0.99421	0.93000	1.00000	0.50568	0.44717	0.52292	0.76317	0.94042
Service call response time vs. Security	<.0001	0.0072		0.3061	0.3739	0.2871	0.0775	0.0052
	6	6	6	6	6	6	6	6
SRS	0.52676	0.40876	0.50568	1.00000	0.99729	0.88386	0.78757	0.71935
Service call response time vs. Server redundancy	0.2829	0.4210	0.3061		<.0001	0.0195	0.0629	0.1071
	6	6	6	6	6	6	6	6
BS	0.46985	0.35300	0.44717	0.99729	1.00000	0.85356	0.77856	0.71873
Service call response time vs. Business continuity	0.3471	0.4925	0.3739	<.0001		0.0145	0.0392	0.0688
	6	6	6	6	7	7	7	7
MS	0.52285	0.51724	0.52292	0.88386	0.85356	1.00000	0.52060	0.72619
Service call response time vs. Meantime to recovery	0.2872	0.2933	0.2871	0.0195	0.0145		0.2309	0.0646
	6	6	6	6	7	7	7	7
TS	0.78758	0.60665	0.76317	0.78757	0.77856	0.52060	1.00000	0.81863
Service call response time vs. Total cost of ownership	0.0629	0.2017	0.0775	0.0629	0.0392	0.2309		0.0243
	6	6	6	6	7	7	7	7
SLAS	0.95165	0.82868	0.94042	0.71935	0.71873	0.72619	0.81863	1.00000
Service call response time vs. Service level agreement	0.0035	0.0415	0.0052	0.1071	0.0688	0.0646	0.0243	
	6	6	6	6	7	7	7	7

#1 Seamless integration (Table 2 – Hardware)

8 Variables:	VS CACS IS PS ES LS RS CLS

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
VS	6	3.66665	3.86438	21.99990	0.33330	9.00000	Component volatility vs Seamless integration
CACS	6	4.77777	3.87969	28.66660	0.33330	9.00000	Component application complexity vs Seamless integration
IS	6	3.72382	3.81732	22.34290	0.14290	9.00000	Interface complexity vs Seamless integration
PS	7	1.00000	0	7.00000	1.00000	1.00000	Product support
ES	6	3.86667	3.66970	23.20000	0.20000	9.00000	Experience with component vs Seamless integration
LS	6	2.42222	2.23375	14.53330	0.20000	5.00000	Learning rate vs Seamless integration
RS	6	3.85715	3.68116	23.14290	0.14290	9.00000	Reliability vs Seamless integration
CLS	6	3.71430	3.82792	22.28580	0.14290	9.00000	Confidence level vs Seamless integration

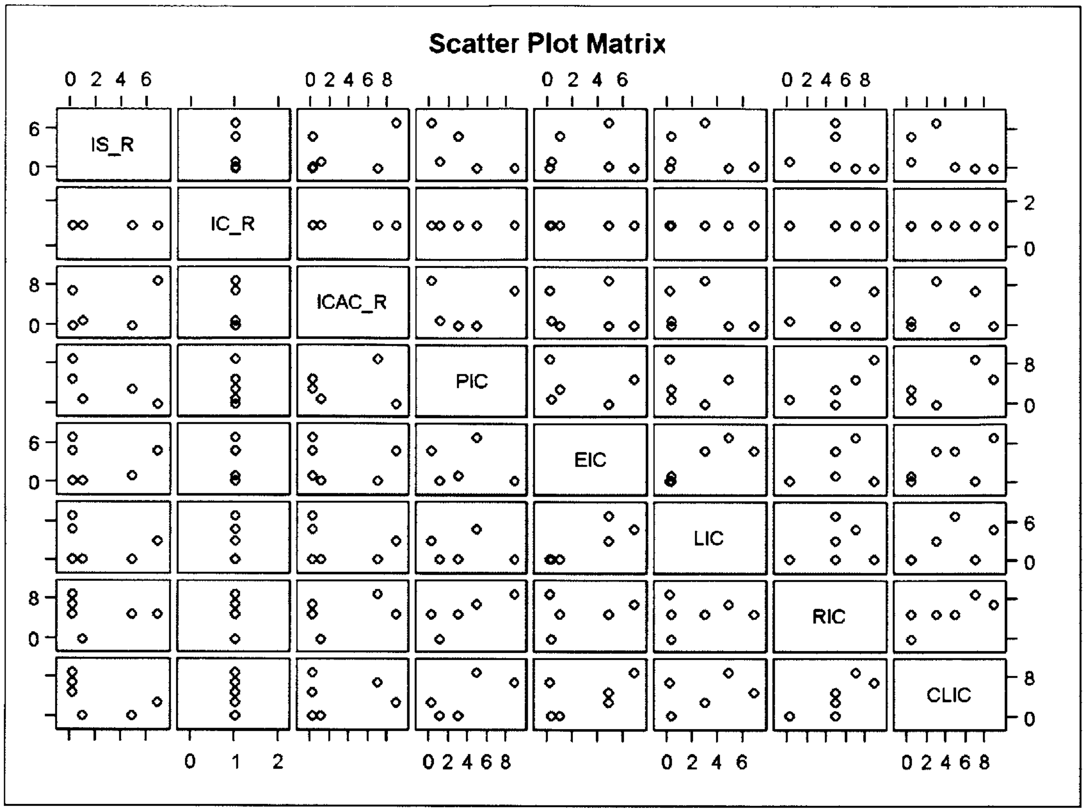

Scatter Plot Matrix
VS
CACS
IS
PS
ES
LS
RS
CLS
0 2 4 6 8
0 1 2
0 2 4

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	VS	CACS	IS	PS	ES	LS	RS	CLS
VS Seamless integration vs Component volatility	1.00000 6	0.87748 0.0216 6	0.99683 <.0001 6	. . 6	0.99664 <.0001 6	0.88559 0.0189 6	0.93997 0.0053 6	0.99664 <.0001 6
CACS Seamless integration vs Component application complexity	0.87748 0.0216 6	1.00000 6	0.85931 0.0283 6	. . 6	0.89281 0.0166 6	0.68379 0.1342 6	0.89358 0.0164 6	0.91062 0.0116 6
IS Seamless integration vs Interface complexity	0.99683 <.0001 6	0.85931 0.0283 6	1.00000 6	. . 6	0.98727 0.0002 6	0.87464 0.0226 6	0.92948 0.0073 6	0.99060 0.0001 6
PS Seamless integration vs Product support	. . 6	. . 6	. . 6	. . 7	. . 6	. . 6	. . 6	. . 6
ES Seamless integration vs Experience with component	0.99664 <.0001 6	0.89281 0.0166 6	0.98727 0.0002 6	. . 6	1.00000 6	0.88269 0.0198 6	0.94076 0.0052 6	0.99663 <.0001 6
LS Seamless integration vs Learning rate	0.88559 0.0189 6	0.68379 0.1342 6	0.87464 0.0226 6	. . 6	0.88269 0.0198 6	1.00000 6	0.78574 0.0639 6	0.87768 0.0215 6
RS Seamless integration vs Reliability	0.93997 0.0053 6	0.89358 0.0164 6	0.92948 0.0073 6	. . 6	0.94076 0.0052 6	0.78574 0.0639 6	1.00000 6	0.93965 0.0054 6
CLS Seamless integration vs Confidence level	0.99664 <.0001 6	0.91062 0.0116 6	0.99060 0.0001 6	. . 6	0.99663 <.0001 6	0.87768 0.0215 6	0.93965 0.0054 6	1.00000 6

#2 Interface complexity (Table2 – Hardware)

8 Variables:	IS_R IC_R ICAC_R PIC EIC LIC RIC CLIC

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
IS_R	6	2.24198	2.99615	13.45187	0.11111	6.99790	Interface complexity vs Seamless integration
IC_R	7	1.00000	0	7.00000	1.00000	1.00000	Interface complexity vs Component volatility
ICAC_R	6	2.93313	3.98716	17.59880	0.20000	9.00090	Interface complexity vs Component application complexity
PIC	5	3.64000	3.52817	18.20000	0.20000	9.00000	Product support vs Interface complexity
EIC	6	3.08888	2.92930	18.53330	0.20000	7.00000	Experience with component vs Interface complexity
LIC	6	2.64443	2.87416	15.86660	0.20000	7.00000	Learning rate vs Interface complexity
RIC	6	5.20000	2.92575	31.20000	0.20000	9.00000	Reliability vs Interface complexity
CLIC	6	4.11110	3.54444	24.66660	0.33330	9.00000	Confidence level vs Interface complexity

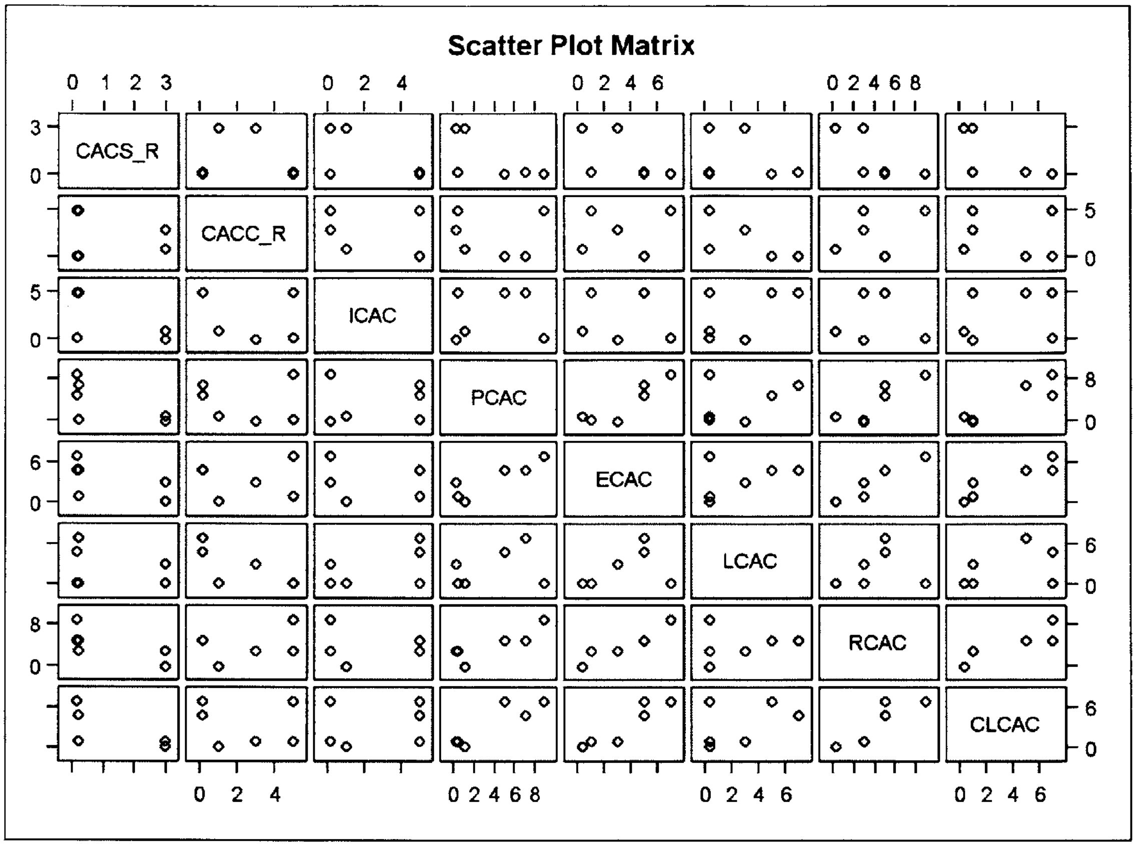

Scatter Plot Matrix
IS_R
IC_R
ICAC_R
PIC
EIC
LIC
RIC
CLIC

Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations								
	IS_R	IC_R	ICAC_R	PIC	EIC	LIC	RIC	CLIC
IS_R	1.00000	.	0.44522	-0.65069	0.01317	-0.24607	-0.15279	-0.55197
Interface complexity vs Seamless integration		.	0.3763	0.2344	0.9802	0.6383	0.7726	0.2561
	6	6	6	5	6	6	6	6
IC_R	.	.	.	.	.	.	.	.
Interface complexity vs Component volatility	.	.	.	.	.	.	.	.
	6	7	6	5	6	6	6	6
ICAC_R	0.44522	.	1.00000	0.06834	-0.08602	-0.26766	0.34413	0.09676
Interface complexity vs Component application	0.3763	.		0.9131	0.8713	0.6081	0.5042	0.8553
complexity	6	6	6	5	6	6	6	6
PIC	-0.65069	.	0.06834	1.00000	-0.18559	-0.11656	0.79830	0.69091
Interface complexity vs Product support	0.2344	.	0.9131		0.7651	0.8519	0.1054	0.1964
	5	5	5	5	5	5	5	5
EIC	0.01317	.	-0.08602	-0.18559	1.00000	0.86645	0.22154	0.55192
Interface complexity vs Experience with component	0.9802	.	0.8713	0.7651		0.0256	0.6731	0.2562
	6	6	6	5	6	6	6	6
LIC	-0.24607	.	-0.26766	-0.11656	0.86645	1.00000	0.14334	0.49852
Interface complexity vs Learning rate	0.6383	.	0.6081	0.8519	0.0256		0.7865	0.3142
	6	6	6	5	6	6	6	6
RIC	-0.15279	.	0.34413	0.79830	0.22154	0.14334	1.00000	0.76116
Interface complexity vs Reliability	0.7726	.	0.5042	0.1054	0.6731	0.7865		0.0788
	6	6	6	5	6	6	6	6
CLIC	-0.55197	.	0.09676	0.69091	0.55192	0.49852	0.76116	1.00000
Interface complexity vs Confidence level	0.2561	.	0.8553	0.1964	0.2562	0.3142	0.0788	
	6	6	6	5	6	6	6	6

#3 component application complexity (Table 2 – Hardware)

8 Variables:	CACS_R CACC_R ICAC PCAC ECAC LCAC RCAC CLCAC

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
CACS_R	6	1.10380	1.46956	6.62282	0.11111	3.00030	Component application complexity vs Seamless integration
CACC_R	6	2.38100	2.28156	14.28601	0.14286	5.00000	Component application complexity vs Component volatility
ICAC	6	2.70900	2.52985	16.25400	0.11110	5.00000	Interface complexity vs Component application complexity
PCAC	6	3.75555	3.78223	22.53330	0.20000	9.00000	Product support vs Component application complexity
ECAC	6	3.55555	2.57913	21.33330	0.33330	7.00000	Experience with component vs Component application complexity
LCAC	6	2.66665	2.85192	15.99990	0.33330	7.00000	Learning rate vs Component application complexity
RCAC	6	4.20000	2.93939	25.20000	0.20000	9.00000	Reliability vs Component application complexity
CLCAC	6	3.55555	3.13877	21.33330	0.33330	7.00000	Confidence level vs Component application complexity

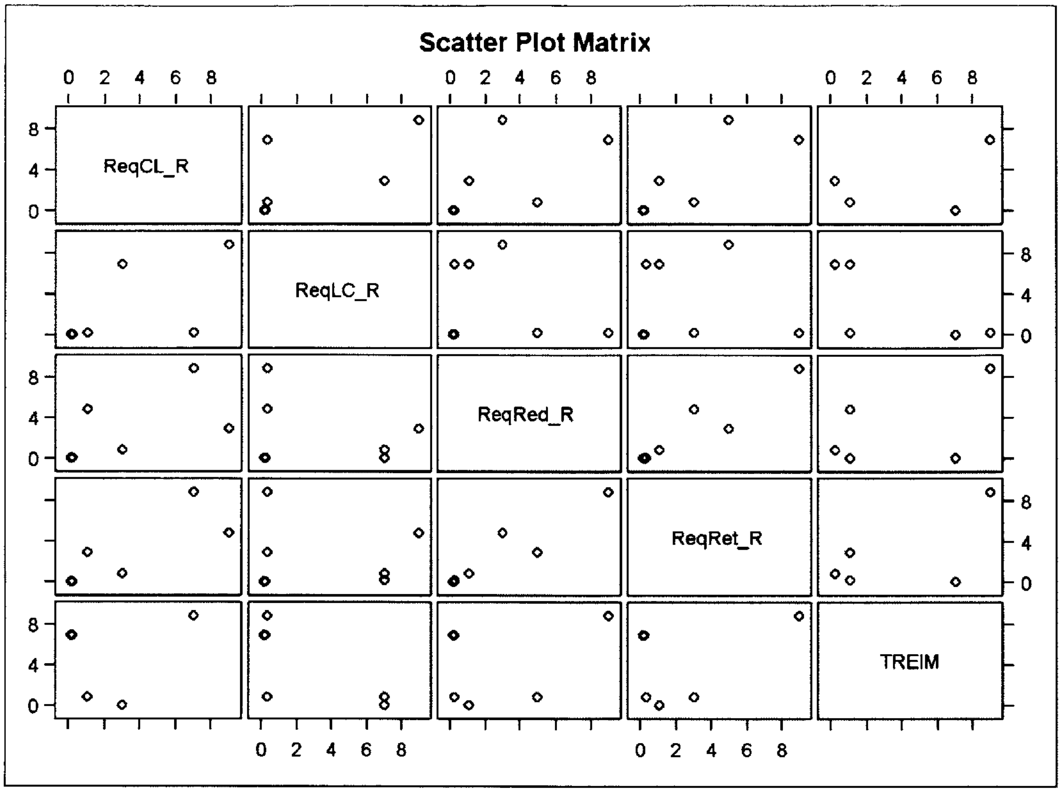

Scatter Plot Matrix
CACS_R
CACC_R
ICAC
PCAC
ECAC
LCAC
RCAC
CLCAC

Pearson Correlation Coefficients, N = 6 Prob > |r| under H0: Rho=0								
	CACS_R	**CACC_R**	**ICAC**	**PCAC**	**ECAC**	**LCAC**	**RCAC**	**CLCAC**
CACS_R Component application complexity vs Seamless integration	1.00000	-0.12925 0.8072	-0.64750 0.1645	-0.65668 0.1566	-0.58116 0.2264	-0.26726 0.6086	-0.69726 0.1236	-0.72809 0.1009
CACC_R Component application complexity vs Component volatility	-0.12925 0.8072	1.00000	-0.35428 0.4908	-0.08508 0.8727	-0.00108 0.9984	-0.72891 0.1003	0.31696 0.5405	-0.16049 0.7613
ICAC Component application complexity vs Interface complexity	-0.64750 0.1645	-0.35428 0.4908	1.00000	0.06129 0.9082	-0.03448 0.9483	0.52771 0.2819	-0.03893 0.9416	0.21794 0.6783
PCAC Component application complexity vs Product support	-0.65668 0.1566	-0.08508 0.8727	0.06129 0.9082	1.00000	0.90516 0.0131	0.33293 0.5191	0.86639 0.0256	0.90550 0.0130
ECAC Component application complexity vs Experience with component	-0.58116 0.2264	-0.00108 0.9984	-0.03448 0.9483	0.90516 0.0131	1.00000	0.40484 0.4259	0.93567 0.0061	0.92052 0.0092
LCAC Component application complexity vs Learning rate	-0.26726 0.6086	-0.72891 0.1003	0.52771 0.2819	0.33293 0.5191	0.40484 0.4259	1.00000	0.13997 0.7914	0.42203 0.4045
RCAC Component application complexity vs Reliability	-0.69726 0.1236	0.31696 0.5405	-0.03893 0.9416	0.86639 0.0256	0.93567 0.0061	0.13997 0.7914	1.00000	0.85555 0.0298
CLCAC Component application complexity vs Confidence level	-0.72809 0.1009	-0.16049 0.7613	0.21794 0.6783	0.90550 0.0130	0.92052 0.0092	0.42203 0.4045	0.85555 0.0298	1.00000

#1 Re-implementation required (Table 3 – Software)

5 Variables:	ReqCL_R ReqLC_R ReqRed_R ReqRet_R TREIM

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
ReqCL_R	6	3.39033	3.77049	20.34196	0.14286	9.00090	Reimplementation required vs Confidence level
ReqLC_R	7	3.42946	4.01895	24.00623	0.14286	9.00090	Reimplementation required vs Lines of code
ReqRed_R	7	2.64915	3.34147	18.54406	0.14286	9.00090	Reimplementation required vs Redesign required
ReqRet_R	7	2.66820	3.32552	18.67739	0.14286	9.00090	Reimplementation required vs Retest required
TREIM	6	4.20000	3.87814	25.20000	0.20000	9.00000	Time constraints vs Reimplementation required


Scatter Plot Matrix
ReqCL_R
ReqLC_R
ReqRed_R
ReqRet_R
TREIM
0 2 4 6 8
0 4 8

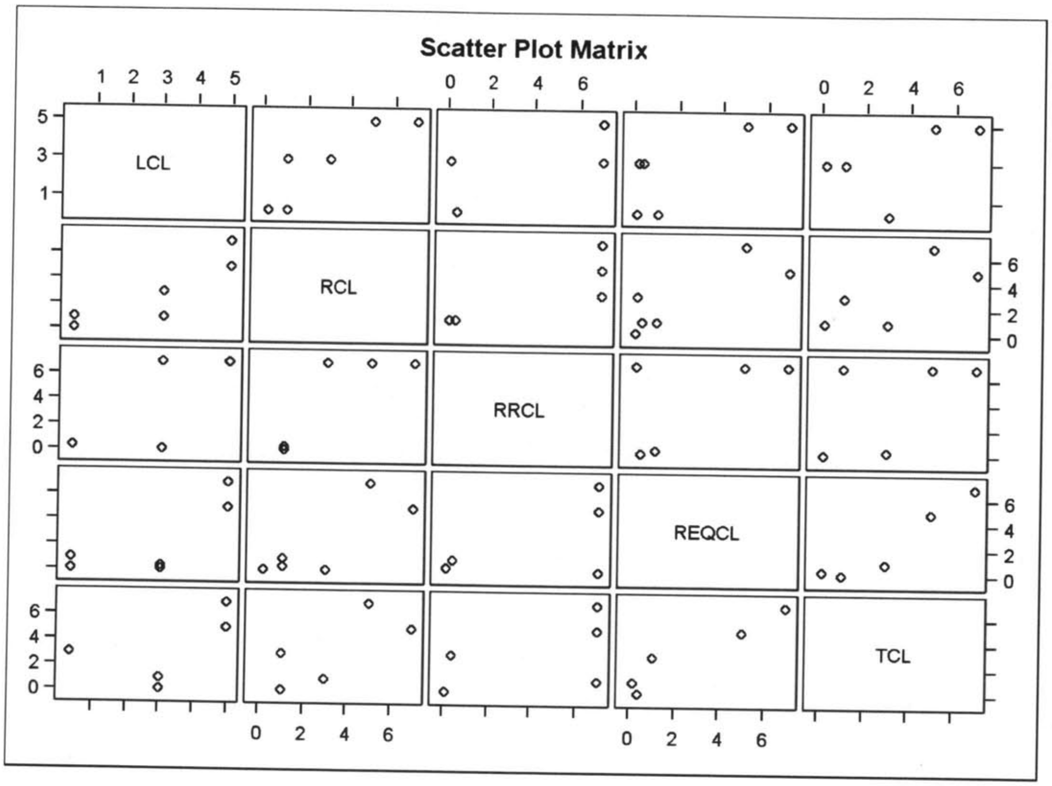
Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations					
	ReqCL_R	**ReqLC_R**	**ReqRed_R**	**ReqRet_R**	**TREIM**
ReqCL_R Reimplementation required vs Confidence level	1.00000 6	0.61638 0.1925 6	0.54909 0.2591 6	0.79414 0.0592 6	0.26412 0.6677 5
ReqLC_R Reimplementation required vs Lines of code	0.61638 0.1925 6	1.00000 7	-0.29127 0.5262 7	-0.07055 0.8805 7	-0.72367 0.1040 6
ReqRed_R Reimplementation required vs Redesign required	0.54909 0.2591 6	-0.29127 0.5262 7	1.00000 7	0.93987 0.0016 7	0.33179 0.5206 6
ReqRet_R Reimplementation required vs Retest required	0.79414 0.0592 6	-0.07055 0.8805 7	0.93987 0.0016 7	1.00000 7	0.43909 0.3837 6
TREIM Reimplementation required vs Time constraints	0.26412 0.6677 5	-0.72367 0.1040 6	0.33179 0.5206 6	0.43909 0.3837 6	1.00000 6

#2 Confidence Level (Table3 – Software)

5 Variables:	LCL RCL RRCL REQCL TCL

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
LCL	6	2.77777	2.09410	16.66660	0.33330	5.00000	Lines of code vs Confidence level
RCL	6	2.85715	2.68479	17.14290	0.14290	7.00000	Redesign required vs Confidence level
RRCL	5	4.28888	3.71319	21.44440	0.11110	7.00000	Retest required vs Confidence level

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
REQCL	6	2.26455	2.97908	13.58730	0.11110	7.00000	Reimplementation required vs Confidence level
TCL	5	3.22858	2.82408	16.14290	0.14290	7.00000	Time constraints vs Confidence level



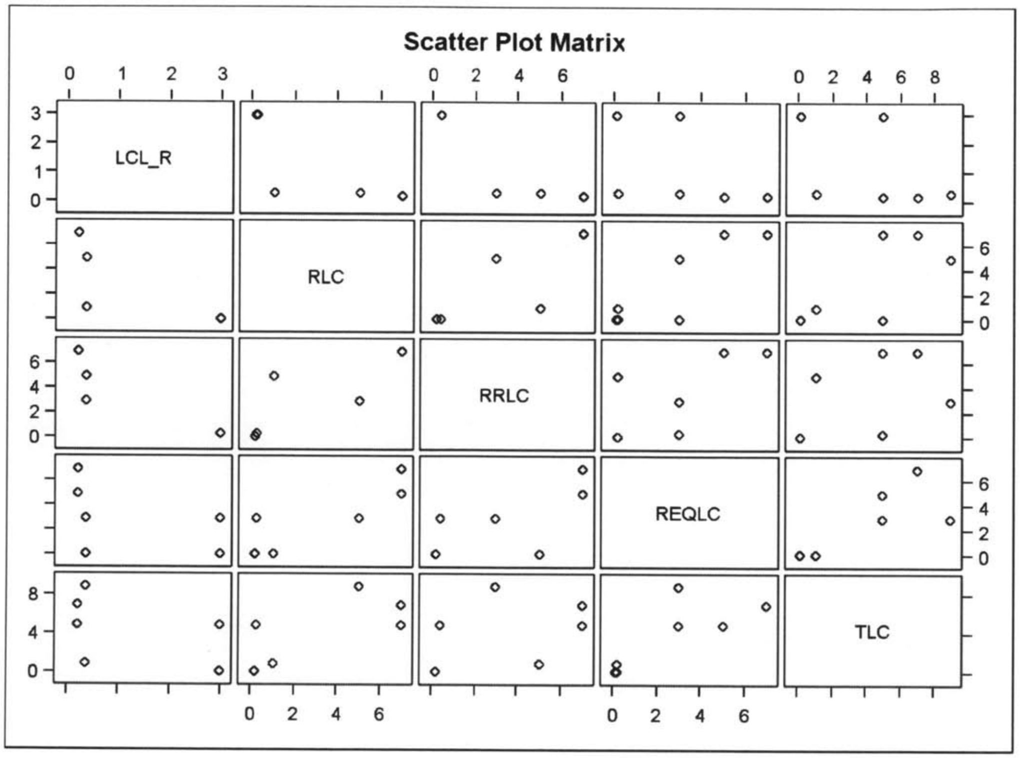
Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	LCL	RCL	RRCL	REQCL	TCL
LCL Confidence level vs Lines of code	1.00000 6	0.88086 0.0204 6	0.74991 0.1444 5	0.77123 0.0725 6	0.53904 0.3485 5
RCL Confidence level vs Redesign required	0.88086 0.0204 6	1.00000 6	0.83998 0.0750 5	0.82411 0.0437 6	0.72162 0.1688 5
RRCL Confidence level vs Retest required	0.74991 0.1444 5	0.83998 0.0750 5	1.00000 5	0.59598 0.2889 5	0.54311 0.3442 5
REQCL Confidence level vs Reimplementation required	0.77123 0.0725 6	0.82411 0.0437 6	0.59598 0.2889 5	1.00000 6	0.95722 0.0106 5
TCL Confidence level vs Time constraints	0.53904 0.3485 5	0.72162 0.1688 5	0.54311 0.3442 5	0.95722 0.0106 5	1.00000 5

#3 Lines of code (Table 3 – Software)

5 Variables:	LCL_R RLC RRLC REQLC TLC

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
LCL_R	6	1.17788	1.41290	7.06727	0.20000	3.00030	Lines of code vs Confidence level
RLC	7	2.92654	3.26926	20.48580	0.14290	7.00000	Redesign required vs Lines of code
RRLC	6	3.74603	3.09629	22.47620	0.14290	7.00000	Retest required vs Lines of code

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
REQLC	7	2.62813	2.69888	18.39690	0.11110	7.00000	Reimplementation required vs Lines of code
TLC	7	3.89343	3.53368	27.25400	0.11110	9.00000	Time constraints vs Lines of code

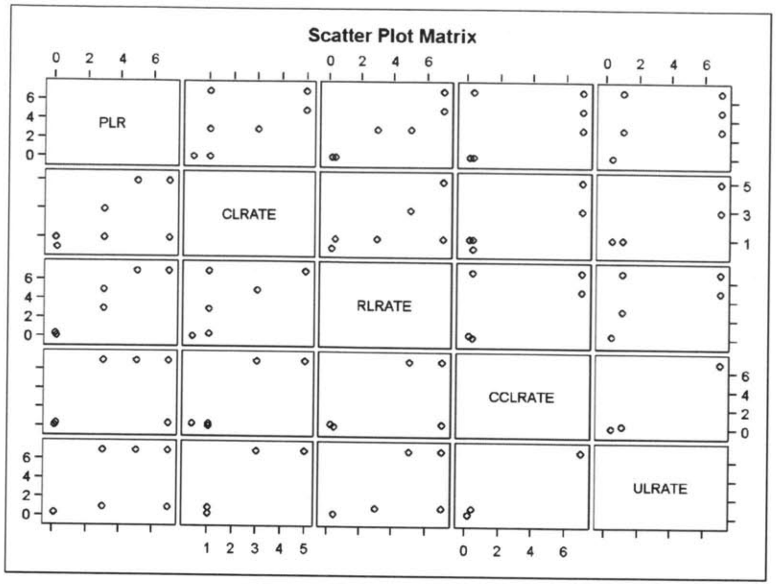


Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	LCL_R	RLC	RRLC	REQLC	TLC
LCL_R Lines of code vs Confidence level	1.00000 6	-0.77328 0.0713 6	-0.83994 0.0750 5	-0.45682 0.3624 6	-0.44961 0.3710 6
RLC Lines of code vs Redesign required	-0.77328 0.0713 6	1.00000 7	0.81503 0.0482 6	0.86777 0.0114 7	0.73976 0.0573 7
RRLC Lines of code vs Retest required	-0.83994 0.0750 5	0.81503 0.0482 6	1.00000 6	0.62884 0.1811 6	0.30301 0.5594 6
REQLC Lines of code vs Reimplementation required	-0.45682 0.3624 6	0.86777 0.0114 7	0.62884 0.1811 6	1.00000 7	0.77852 0.0392 7
TLC Lines of code vs Time constraints	-0.44961 0.3710 6	0.73976 0.0573 7	0.30301 0.5594 6	0.77852 0.0392 7	1.00000 7

#1 Learning Rate (Table4 – Support)

5 Variables:	PLR CLRATE RLRATE CCLRATE ULRATE

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
PLR	7	3.60771	2.88454	25.25400	0.11110	7.00000	Professional experience vs Learning rate
CLRATE	7	2.33333	2.00001	16.33330	0.33330	5.00000	Cost vs Learning rate
RLRATE	7	4.21089	3.08249	29.47620	0.14290	7.00000	Repairs vs Learning rate
CCLRATE	6	3.63492	3.68692	21.80950	0.14290	7.00000	Call center vs Learning rate
ULRATE	6	3.88888	3.41674	23.33330	0.33330	7.00000	Upgrades vs Learning rate

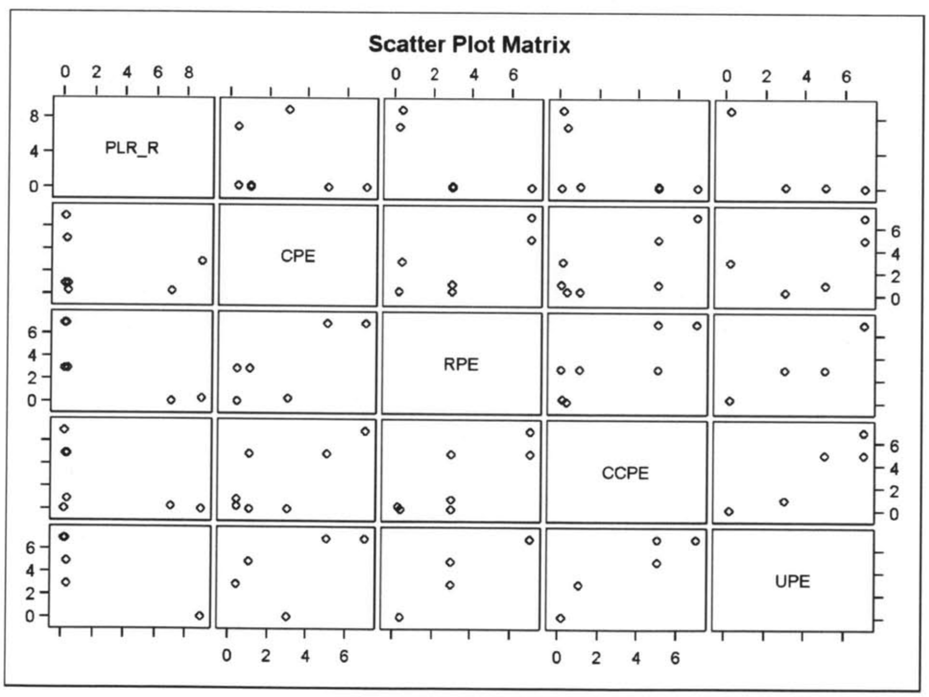


Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	PLR	CLRATE	RLRATE	CCLRATE	ULRATE
PLR	1.00000	0.58452	0.95063	0.45699	0.37964
Learning rate vs Professional experience		0.1681	0.0010	0.3622	0.4579
	7	7	7	6	6
CLRATE	0.58452	1.00000	0.71920	0.92872	0.92612
Learning rate vs Cost	0.1681		0.0685	0.0074	0.0080
	7	7	7	6	6
RLRATE	0.95063	0.71920	1.00000	0.63923	0.61896
Learning rate vs Repairs	0.0010	0.0685		0.1718	0.1901
	7	7	7	6	6
CCLRATE	0.45699	0.92872	0.63923	1.00000	0.99877
Learning rate vs Call center	0.3622	0.0074	0.1718		<.0001
	6	6	6	6	5
ULRATE	0.37964	0.92612	0.61896	0.99877	1.00000
Learning rate vs Upgrades	0.4579	0.0080	0.1901	<.0001	
	6	6	6	5	6

#2 Professional experience (Table 4 – Support)

5 Variables:	PLR_R CPE RPE CCPE UPE

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
PLR_R	7	2.45017	3.83550	17.15118	0.14286	9.00090	Professional experience vs Learning rate
CPE	7	2.52380	2.60241	17.66660	0.33330	7.00000	Cost vs Professional experience
RPE	7	3.36190	2.76991	23.53330	0.20000	7.00000	Repairs vs Professional experience
CCPE	7	2.65533	2.90942	18.58730	0.11110	7.00000	Call center vs Professional experience
UPE	5	4.44000	2.89275	22.20000	0.20000	7.00000	Upgrades vs Professional experience



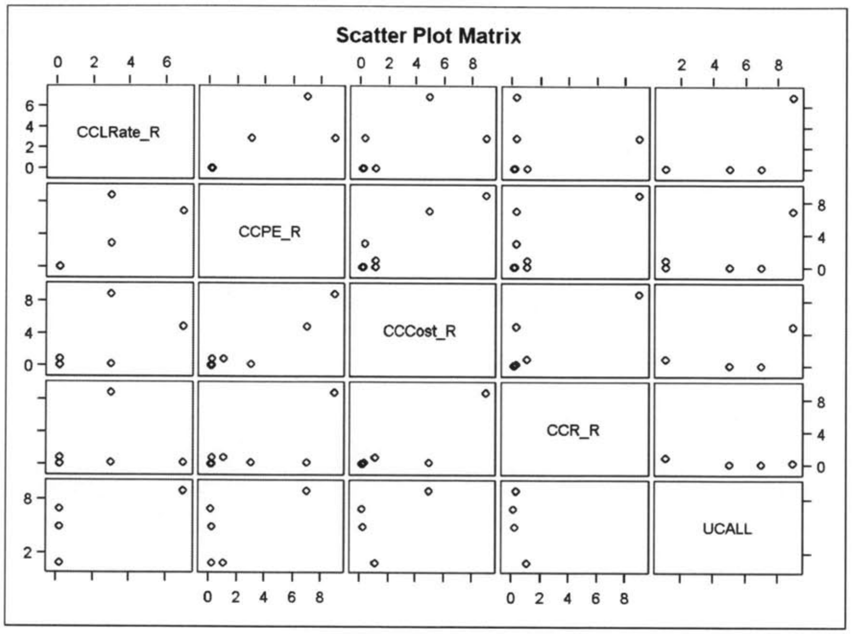
Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	PLR_R	CPE	RPE	CCPE	UPE
PLR_R Professional experience vs Learning rate	1.00000 7	-0.18975 0.6836 7	-0.75979 0.0475 7	-0.56617 0.1852 7	-0.82991 0.0820 5
CPE Professional experience vs Cost	-0.18975 0.6836 7	1.00000 7	0.75544 0.0495 7	0.72803 0.0636 7	0.56350 0.3225 5
RPE Professional experience vs Repairs	-0.75979 0.0475 7	0.75544 0.0495 7	1.00000 7	0.82640 0.0219 7	0.95036 0.0132 5
CCPE Professional experience vs Call center	-0.56617 0.1852 7	0.72803 0.0636 7	0.82640 0.0219 7	1.00000 7	0.93042 0.0218 5
UPE Professional experience vs Upgrades	-0.82991 0.0820 5	0.56350 0.3225 5	0.95036 0.0132 5	0.93042 0.0218 5	1.00000 5

#3 Call center (Table4 – Support)

5 Variables:	CCLRate_R CCPE_R CCCost_R CCR_R UCALL

Simple Statistics							
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**	**Label**
CCLRate_R	6	2.23785	2.71984	13.42707	0.14286	6.99790	Call center vs Learning rate
CCPE_R	7	2.93457	3.64708	20.54196	0.14286	9.00090	Call center vs Professional experience
CCCost_R	7	2.38244	3.37776	16.67709	0.14286	9.00090	Call center vs Cost

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
CCR_R	7	1.71577	3.23243	12.01042	0.14286	9.00090	Call center vs Repairs
UCALL	5	4.60000	3.57771	23.00000	1.00000	9.00000	Upgrades vs Call



Pearson Correlation Coefficients Prob > |r| under H0: Rho=0 Number of Observations	CCLRate_R	CCPE_R	CCCost_R	CCR_R	UCALL
CCLRate_R Call center vs Learning rate	1.00000 6	0.78228 0.0659 6	0.56784 0.2398 6	0.12291 0.8166 6	0.68313 0.3169 4
CCPE_R Call center vs Professional experience	0.78228 0.0659 6	1.00000 7	0.93793 0.0018 7	0.71202 0.0727 7	0.63097 0.2537 5
CCCost_R Call center vs Cost	0.56784 0.2398 6	0.93793 0.0018 7	1.00000 7	0.85850 0.0134 7	0.52732 0.3612 5
CCR_R Call center vs Repairs	0.12291 0.8166 6	0.71202 0.0727 7	0.85850 0.0134 7	1.00000 7	-0.86326 0.0594 5
UCALL Call center vs Upgrades vs	0.68313 0.3169 4	0.63097 0.2537 5	0.52732 0.3612 5	-0.86326 0.0594 5	1.00000 5