FACTORS THAT AFFECT CASE TOOLS USAGE BY SYSTEMS DEVELOPERS AND ANALYSTS

Jawad Al-Khairy

A Thesis

In

The John Molson School of Business

Presented in Partial Fulfilment of the Requirements
for the Degree of Master of Science in Administration at
Concordia University
Montreal, Quebec, Canada

April 2001

0-612-59276-6

ABSTRACT

Factors that Affect CASE Tools Usage by System Developers and Analysts

Jawad Al-Khairy

This research is aimed at identifying factors that affect CASE tools usage by system developers and analysts. The Technology Acceptance Model (TAM) developed by Davis (1989) is used as the main building block. Certain modifications are made to the original TAM resulting in a final model that investigates the effect of four main factors on CASE usage.

The data was collected electronically through a web site designed specifically for this purpose. The research respondents were Systems developers and analysts that have CASE technology available to them in their work place, but at the same time using it is on a voluntary basis. Path Analysis was used to test and explain the final model. The strength and direction of relationships between each factor and CASE usage as well as relationships among the factors themselves were also studied.

The research provides a general understanding of the factors that affect system professionals CASE usage, how these factors interact among each other, and their collective effect on the technology usage. The study also highlights certain recommendations for future research.

ACKNOWLEGEMENTS

To my parents, who stood by me, not only throughout this thesis, but throughout my life, I owe it all to you. I hope one day I'll be able to prove worthy of all the love and support you gave, and have the ability to put you before I. My success is only an extension of yours.

As for my brother and sister, I believe they ought to thank me for all the dishes I washed when I wanted to do anything but work on my thesis. Hopefully one day I'll be able to do to you what you did to me, and much more.

Thank you Professor Kira for all the support and direction you gave. Your door was always open. You gave me enough rope to make my own mistakes, yet not enough for me to hang my self. Your flexibility and understanding made a long journey much more pleasant.

Professor Croteau, thank you for all what you did. You made me believe that I could do it. You found way to motivate the big baby I was. I thank you for that.

Thank you Professor Tomberlin for being there when I needed help and advise. Your busy schedule never got in the way.

Thank you Heather and Theresa for babysitting me for over three years. I really appreciate your efforts in taking care of me.

TABLE OF CONTENTS

List of Figures

List of Tables

INTRODUCTION

Performance was always a major concern for organisations. Since the early years of the nineteenth century, Tylor, father of Scientific Management, was concerned with the productivity and efficiency of workers, and their effect on the workers overall performance. He developed a very simple, yet effective incentives system. The main idea was to get the workers to reach a certain level of production. To accomplish this, each worker was paid X amount of dollars for every piece produced. The average worker produced around 8 pieces per day. Tylor set his aim at 10 pieces per worker per day. To motivate his workers to achieve this level of 10 pieces, each worker that produced 10 or more pieces per day was paid X+1 for every piece he produced that day. Aside from being a success, the system was a landmark in the management field.

In other efforts to enhance performance, Mayo conducted his famous Hawthorne study. He analysed the productivity of textile workers under different work environments. The study found that the workers productivity increased when working in a better environment and under better conditions.

Several decades later, and after all the breakthroughs in Management Science, Communications, and Information Technology, performance is still a top priority. During this era, that many calls the era of Information Revolution, organisations, decision makers, and strategists are resorting to information technology seeking its potential for a substantial improvement in performance (Edelman, 1981).

In the early days of the information revolution, use of information systems was very scarce and mainly aimed at replacing few basic functions of the paper system. The information system was more or less an automation of the data storing and processing activities, which in several cases worked with and parallel to the manual paper system.
As organisations developed, and more sophisticated information systems were made available through the rapid breakthroughs in hypermedia and information technology, powered by better and more powerful computer hardware, the role of information systems began to exceed that of support to the existing business functions. Organisations realised the need to keep abreast of technology if they were to surf the wave of the new era. Information systems functions moved out of the accounting department. Organisations created specialised departments for information technology. The role of information systems became that of creating and sustaining competitive advantage for the organisation. More importantly, it became a means of survival.

Nowadays the IT department is a major player in the organisational setting. Information systems became the nerve system of the organisation. Along with this, the need for accurate, efficient and vigilant information systems became of vital importance. Organisations realised that the way to achieve this lies in two aspects. One was the human aspect, which focuses on the interaction between the human system users and the system. Practitioners as well as academics recognised that for an information system to have a positive influence, it must be properly utilised (Goodhue and Thompson, 1995). If organisations were to realise the full potential of such systems, their users should have a high level knowledge and expertise in the use of the system. Decision-makers made current and potential system users undergo training sessions and workshops aimed at familiarising them with the system and helping them explore its capabilities in order to develop the necessary competence.

The second aspect had to do with the information system itself. To add value, the information system should be properly designed to serve the requirements of the organisations, groups, and individuals that use it (Dishaw and Strong, 1998). This led decision-makers to steer their attention towards the development process of information systems. Researchers in this field reported several pitfalls in systems development practices. Delays, inaccuracies, and improper documentation were among other problems that hindered systems development (Nelson and Rottman, 1996).

Specialised information technology organisations and consulting firms responded to this market need in several ways. One of their efforts resulted in the development of specialised software for systems development. Such a package is known as CASE tool (Computer Aided Software Engineering tool). In general, these tools are used to support and assist systems developers in all aspects of development efforts throughout the entire phases of the system development life cycle. Both researchers and practitioners showed great interest in CASE technology for easing software development practices and solving other existing problems in software development such as developers productivity and long delivery times (Finlay and Mitchell, 1994, Orlikowski, 1993).

A significant number of research and studies reported a positive impact of CASE technology usage on systems development. In an empirical study of the benefits of CASE tools, Finlay and Mitchell (1994) found that CASE introduction "has led to productivity gains of 85 percent, system delivery rate increases of around 200 percent, and quality improvements". Other studies reported similar enhancements in developers' performance, shorter delivery periods, and better quality of information systems (Norman and Nunamaker 1989, Banker and Kauffman1991, Aaen et al. 1992).

Reviewing the above-mentioned benefits of CASE tools, one would expect them to be widely used. Yet, a study by Kemerer (1992) shows that 70 percent of the CASE tools are never used, 25 percent are used by only one group, and the remaining 5 percent widely used, but not to capacity. Another study conducted in Denmark and Finland revealed that less than 20 percent of CASE user organisations can be classified as routine users of CASE (Aaen et al. 1992).

The proposed research is geared to investigate CASE adoption behaviour and usage by system developers. In an environment where improving performance is of the essence, what is the missing link that rendered benefits of CASE usage negligible? Throughout this research, and by developing a good understanding of CASE adoption behaviour, we should be able to identify and isolate those factors causing or promoting low CASE adoption and usage. Findings of this study should help CASE vendors better design their software to meet user requirements, thus, enabling organisations to reap the promised benefits.

LITERATURE REVIEW

The Definition of CASE

CASE stands for Computer Aided Software Engineering[1]. They are software programs that help computer systems developers in specifying, designing, and constructing software systems. The tools automatically implement commonly accepted development procedures -to different extents- to help developers complete specific tasks. Using these tools stimulates the thinking process of developers, whether working within a development team or on their own (Scott, 1995).

Vessey, Jarvenpaa, and Tractinski (1992) defined CASE as "software that provides automated assistance for software development, maintenance, or project management activity". In general, a CASE tool is a combination of "tools and methods to support an engineering approach to software development at all stages of the process" (Forte and Norman, 1992). This technology assists system analysts and developers in documenting and modelling an information system from its initial user requirements through design and implementation.

A very important and pivotal characteristic of CASE is that its usage is recognised long after the implementation phase, that is in the maintenance activities of an information system. In other words, although there are benefits introduced by CASE during the early

[1]The letter S in CASE is used as abbreviation for Software or Systems in different literature

stages of system development, the major contributions of CASE, like complete and clear system documentation, are recognised when the need to maintain the system arises.

About CASE tools

Computer Aided Systems Engineering dates back to the early-to-mid 1970's, but their use was limited due to the fact that enormous power is needed to run these applications. However, in the mid-eighties, advancement in computer technology and the increased need for successful and efficient information systems made CASE tools more popular (Beck and Perkins, 1988).

In 1990 the annual worldwide market of CASE was estimated to be $4.8 billion. In five years, the market size increased by around 270% to become $12.11 billion in 1995 (Jarzabek and Huang, 1998). Nowadays, there are hundreds of CASE products available that support a variety of systems development needs, such as requirements elicitation and analysis, design and communication, enforcement of standards and methodologies, prototyping and RAD (Rapid Application Development), reverse engineering, and software maintenance and reengineering (Post, Kagan and Keim, 1998).

Just like any other technology, different CASE products contain different strength and weaknesses in terms of specific functions. In an empirical study, IEF for example showed strong prototyping and coding capabilities, while Excelerator rated better for consistency of graphs and support for different methodologies in analysis and design. Although poor graphics and analysis capabilities were considered major weaknesses for Pacbase, it scored highly in provisions for audit trail (Post, Kagan and Keim, 1998).

Within the context of this research, the adoption of CASE technology -not a CASE product in particular- is addressed. Although specific facts for specific CASE tools like Oracle or Visible Analyst might be cited, the issue addressed in the context of this research is CASE technology in general.

Categories of CASE

Researchers tried to group the different CASE tools in different categories. They used different aspects and criteria as basis for their categorisation schemes. The frameworks used included functionality, life cycle support, and methodology support.

Henderson and Cooprider (1990) identified two major dimensions in CASE tool functionality, Production and Co-ordination.

Production functionality provides support to the individual developer in software development or maintenance. It has three parts:

a- Representation, which helps the user develop a better understanding of the problem by aiding him in representing the problem.

b- Analysis functionality. This part aims at enabling the user to build further knowledge and understanding about the problem, diagnosing problems, and ultimately planning solutions. The functionality achieves this by furthering the users' exploration, evaluation, and representation of the problem at hand.

c- Transformation functionality, which supports the actual changes or additions to the software.

Co-ordination functionality has to do with providing support to the co-ordination activities necessary when the user is working within a development team or has to share information with other groups in the organisation (Dishaw and Strong, 1998).

Similar work was done by Sharma and Rai, (2000), however through a different approach. The framework they used classified CASE according to the area it supports. In addition to support for production and support for co-ordination, a third functionality was added, CASE support for Organisation.

Other attempts to classify CASE were based on their support for a specific development technique. The focus was on the Systems Development Life Cycle (SDLC) in terms of the number of life cycle phases supported and the level of integration the tool provided between the phases. Using this framework, Selamat and Rahim (1996) suggested two categories:

- Upper CASE, which includes functions such as graphics, data dictionary, and design analysis. It provided support for the early stages of the SDLC.
- Lower CASE, with functions like prototyping and code generation. This category of CASE supported the later stages of the SDLC.

They concluded that upper CASE tools resulted in higher productivity for the analysis stage, followed by the design and planning stages, while lower CASE users realised the tools' impact during the coding and testing phases.

This classification of systems development functions into early phases and later phases is in sink with Rai and Patnayakuni's (1996) work. They suggested three categories, Front-end system work, Back end system work, and project management.

In their study, Flynn, Vagner and Vecchio (1995) cited the work of Mair (1993) in categorising CASE into four segments according to the CASE market. The first segment is Simple tools, which are designed to perform one particular task, like data flow diagramming or project management. These tools usually work on a stand-alone basis. However, some do offer a limited scope of interface capabilities, which allow them to share data with other tools.

The second segment is called Workbenches. Workbenches are known to support one full phase or function of the SDLC. Tools within a workbench have the capability of working together by sharing databases and exchanging information via compatible interaction facilities.

Integrated CASE (ICASE) makes the third segment. They support the SDLC to a greater extent than workbenches. The integration of different tool sets allows them to provide support to technical as well as managerial aspects of a development project.

The last segment is the Open Environment (IPSEs). This set of tools support the user throughout the entire SDLC. They usually have multiple customisation and interaction capabilities.

In his empirical study, Jankowski (1995) classified CASE tools on the basis of their methodology support. He suggested two categories, Restrictive and Guided. Similar analysis was done by Vessey, Jarvepaa and Tractinski (1992), yet they argued that there are three philosophies for methodology support:

a- Restrictive: which forces the user of the tool to follow the development methodology supported by the tool in a normative manner. Should a methodology violation occur, the tool will prompt the user to fix it by issuing an error message. The user cannot proceed unless his work conforms to the tools' development methodology. For example, the developer must start with a context diagram and then explode it into a lower level. Working the data flow diagrams from the bottom up is not permitted.

b- Guided philosophy: which is designed to encourage the user of the tool to follow the development methodology in a normative manner. The tool would issue a warning message whenever it detects a methodology violation. A guided philosophy tool would suggest rather than enforce proper methodological development procedures. Referring to the example used in a, a user starting with a lower level data flow diagram can override the warning message and continue his work.

c- Flexible philosophy provides the user with complete freedom in following its supported methodology. The user of such tool is allowed to work without any interruption from the tool. If a methodology violation occurs, the tool will not communicate that to the user unless he/she requests the tool to verify his/her work. For example, once a developer finishes all levels of data flow diagrams, he/she can prompt the tool to verify the work for internal and hierarchical consistency.

CASE as Methodology Companion

A major role of CASE technology is its ability to serve as a methodology companion. "To provide active assistance, a CASE tool should have built-in knowledge of software methods, software processes, application domain, and developer behaviour" (Jarzabek and Huang, 1998). Jankowski (1995) stated that "support for a particular systems development methodology is one of the most important criteria for choosing a CASE tool".

The embedded support of CASE to a certain methodology allows the operationalization and facilitation of complex development methods such as Information Engineering and System Software and Application Design Methods (SSADM). The guidance provided by the tool to the developers through the various functions, such as completeness and consistency checks, results in more accurate and complete systems (Flynn, Vangner and Vecchio, 1995). By encouraging detailed systems analysis and design, "the quality of the delivered system is improved because the requirements of the users are better met" (Selamat and Rahim, 1996).

Phillipson (1990) argued that systems developed using CASE tools are invariably easier to maintain because they are constructed in a much cleaner and more logical fashion. A study conducted by Freeman (1992) showed that the average growth in functionality for information systems developed without the use of CASE was around 12%. The rate dropped significantly to 2% for CASE assisted projects.

This is considered of great importance, especially in light of the results provided by Banker et al (1991) which estimated maintenance costs to make 50-80% of IS expenditures.

By following proper systems development practices, CASE users spend more time on feasibility study, conceptual analysis, and detailed design. This investment of time and effort in the early stages of the SDLC pays off by reducing the time required for later phases or backend work such as coding, testing, and most importantly, maintenance, as well as providing solid overall quality improvements (Nelson and Rottman, 1996).

However, the relation between CASE tools and structured methods is interdependent. Guinan, Cooprider and Sawyer (1997) found that structured method usage influenced the amount of CASE usage. They also stated that "higher levels of CASE tool usage in combination with increased use of structured methods increases the likely hood that the project stakeholders will be pleased with the system".

Yang (1999) emphasised that successful implementation of CASE significantly depends on the methodology usage. This was evident in Guinan, Cooprider and Sawyer's (1997) work where CASE usage by teams with well-structured processes enhanced their output and improved their performance, while on the other hand, teams with ad hoc processes were in a state of chaos when the CASE tool was introduced.

As a methodology companion, a CASE tool can implement methodology support in two ways, automated control and information feedback. In the automated control support, the tool is active in terms of processing the developers' input. For example the tool performs functions like code generation and registry updating in the background while the developer is working on something else. In the information feedback method, the tool plays more of a reactive role. For example whenever a methodology violation occurs, the tool communicates the violation to the user (Vessey, Jarvenpaa and Tractinsky, 1992).

The degree of support that the different tools provide also differs. In the previous section "Categories of CASE", three different philosophies were presented, Restricted, Guided, and Flexible. Each of these philosophies has its weaknesses, which is considered a form of strength for the other philosophies.

Windsor (1986) suggested lack of flexibility as one disadvantage of some CASE tools. Supporting a specific systems development methodology prevents the tool from working well with other methods. The organisation is also forced to change its standards in order to accommodate those incorporated with the tool. This is supported in Rai and Patnayakuni's (1996) argument in explaining fear and resistance associated with adoption of the technology. Yang (1999) emphasised that "using a new methodology after implementation must be accompanied with communication and training". It was empirically proven that appropriate methodology training facilitated and increased structured methods' usage by systems analysts (Guinan, Cooprider and Sawyer, 1997).

Another point of view argued that CASE tools are customised to support methods instead of soft aspects of systems development. Users often find CASE diagrams too dry, notations and editors too restrictive, impeding rather than aiding them.

Jarzabek and Huang (1998) concluded that such method-centred tools are not attractive, in other cases not helpful to the users, and should be reshaped -as well as transformed- into more user-oriented packages.

On the other hand, being too flexible was also criticised for not providing proper and enough guidance in methods usage. Being too flexible, the tool is said to induce a false belief of method confidence. By allowing the users to create inadequate conceptual models out of flexibility, it may "support developers in justifying bad models" (Eriksen and Stage, 1998).

In their study, Vessey, Jarvenpaa and Tractinsky (1992) found that through direct methodology support, restrictive and guided philosophy tools are the most likely to increase software productivity as well as quality. It is worth mentioning however, that they also found that "non of the tools studied provided a totally consistent approach". Each tool used more than one philosophy for the different support functions it provided.

CASE Enhancements to Systems Development

Various spectrums of research, literature, and empirical studies reported solid enhancements to systems development associated with CASE adoption and usage. Researchers argued that in general, CASE users experienced fewer problems during systems development and had better output. These positive effects of CASE impacted the different phases of the SDLC in different ways. In general, CASEs' impact on systems development was through six main aspects; Productivity, Quality, Maintenance, Standardisation, User Communication, and Cost.

a. Productivity is the "ratio of quality work products per unit of time" (Selamat and Rahim, 1996). Enhancing systems developer productivity is one of the major concerns for both academics and practitioners. From its definition, one way for improving productivity is by producing the same quality work in a shorter period of time. Rothstein et al. (1993) reported that CASE usage shortened the system delivery period. Several other studies also found that CASE usage brought about productivity gains by cutting down the lead-time for systems development (Feuche, 1989, Martin, 1989, Norman and Nunamaker, 1989). Although the time invested in the early phases of the SDLC is longer for CASE assisted projects than non-CASE assisted ones, the overall system delivery time was found to be shorter (Selamat and Rahim, 1996). Finlay and Mitchell (1994) reported a 70% decrease in delivery time due to CASE usage.

   Flynn, Vagner and Vecchio (1995) suggested three areas where CASE can reduce the time required for systems development:

1. Speeding up the design process.
2. Making rapid, iterative prototyping a practical way of building systems to meet user requirements.
3. Reducing time required for maintenance, because the program specifications, not the code, are maintained.

Another way for enhancing productivity is by building a better quality system during the same time period. In this sense, Martin (1995) argued that through graphics generation, data dictionary creation, explosion, and code creation, CASE was able to increase its user productivity. Upper CASE usage enhanced systems developers productivity in the analysis stage as well as design and planing stages. On the other hand, lower CASE users realised dramatic productivity improvements during coding and testing phases (Selamat and Rahim, 1996). In one study, the use of Excelerator resulted in 30-40% productivity enhancement at the analysis and design stages (Chikofsky and Rubenstein, 1989). A 25% improvement in productivity was also reported by Loh and Nelson (1989). Furthermore, an 85% productivity improvement was reported by Finlay and Mitchell (1994) and even as high as 600% was reported by Coolidge (1988).

In a study conducted by Coupe and Onodu (1997), CASE was reported higher than 4 Generation Languages (4GLs) in terms of analyst/programmer productivity. Several other studies also reported variable degrees of productivity enhancement in systems development brought about by CASE usage (Iivari, 1996, Nelson and Rottman, 1996, Selamat and Rahim, 1996, Flynn, Vangner and Vecchio, 1995, Whitten, Bentley and Barlow, 1994, Orlikowski 1993, Stobart, Reeken and Low, 1993, Banker and kauffamn, 1991, Low and Jeffery, 1991, Stobart, Thompson and Smith, 1991, Swanson et al, 1991, Yellen, 1990, Burkhard, 1989, Necco, Tsai and Holgeson, 1989, Norman and Nunamaker, 1989).

b. Quality is "concerned with the system meeting user requirements" (Flynn, Vangner and Vecchio, 1995). It also involves the development of accurate systems that require little maintenance and has a relatively low growth in functionality. As Martin (1995) puts it, " with CASE think quality, not quantity". He argues that through capabilities like Data Flow Diagrams consistency, data dictionary completeness, linkages among the different levels, and structured check lists, a CASE tool is very well equipped to enhance the quality of the developed systems.

Loh and Nelson (1989) estimated a 20% improvement in requirement definition to be brought about by CASE. Eriksen and Stage (1998) also reported that CASE user groups ended up with better quality documents as they kept improving them from analysis to design stages. Quality gains were also realised at the code generation phase by British and Australian respondents (Stobart, Reeken and Low, 1993).

Part of the improvement in systems quality introduced by CASE results from the emphasis the tool puts on the early stages of the SDLC. In the previous section, it was apparent that CASE users invest more time in the early stages of the SDLC. The main idea is that the more you do in analysis, the fewer problems you face across the rest of the SDLC, especially maintenance (Coupe and Onodu, 1997).

Another source of quality improvement is CASE support for structured methods[2]. Case developed systems are said to be cleaner and more logical in terms of documentation and data dictionaries (Philipson, 1990). Organisations using upper CASE reported considerable improvement in the quality of the developed systems (Selamat and Rahim, 1996). Norman and Nunamaker (1989) concluded that software designers had improved their output with the use of CASE tools. Several other studies also reported solid systems quality improvements through CASE usage (Guinan, Cooprider and Sawyer, 1997, Iivari, 1996, Nelson and Rottman, 1996, Selamat and Rahim, 1996, Flynn, Vangner and Vecchio, 1995, Whitten, Bentley and Barlow, 1994, Kusters and Wijers, 1993, Banker and Kauffman, 1991, Low and Jeffery, 1991, Yellen, 1990, Burkhard, 1989, Necco, Tsai and Holgeson, 1989, Norman and Nunamaker, 1989, Kim and Westin 1988).

c. Maintenance is the "process of changing the existing production software" (Dishaw and Strong, 1998). It is the last, yet an ongoing phase of the SDLC. Although its functions might seem as merely routine system services, this stage is of great importance to the organisation. Swanson and Beath (1989) estimated maintenance costs to make around 70% of the MIS department budget. They suggest that this

---

[2] See "CASE as Methodology Companion"

number is even higher in mature organisations. There is a clear need to pay good attention to the impact of all other SDLC stages on the maintenance phase, as all mistakes, flaws, and pitfalls will have to be addressed during this phase.
CASE researchers emphasised the significant positive influence of CASE usage on the maintenance phase. Dishaw and Strong (1998) argued that CASE support to the maintenance phase assists the users in discovering and understanding the physical and logical designs of the system being maintained. This aspect of CASE is of great importance especially for maintaining large-scale information systems where the individual performing the maintenance has never dealt with the system before (Dishaw and Strong, 1998, Schneidwind 1987).

Philipson (1990) relates the easy maintenance of CASE built systems to the clean and logical fashion in which they were built. In fact, the way CASE works, which is altering the specifications instead of altering the code, is one of the keys to its success. This was evident in Martin's (1995) study where users rated CASE better than 4GLs in terms of maintenance productivity. Through capabilities like ease of search and information retrieval, CASE makes it very practical to alter specifications stored in the repository. In one empirical study, this approach to software maintenance resulted in a 30% improvement in maintenance productivity (Loh and Nelson, 1989). Similar findings were reported by several other studies (Nelson and Rottman, 1996, Flynn, Vangner and Vecchio, 1995, Whitten, Bentley and Barlow, 1994, Banker and Kauffman, 1991, Low and Jeffery, 1991, Yellen, 1990, Burkhard, 1989, McClure, 1989, Necco, Tsai and Holgeson, 1989, Norman and Nunamaker, 1989).

d. The development of information systems is a complex task that includes multiple processes. There are many issues and details that have to be addressed by the development team. At the end, the different parts of the system that were built by the individuals separately will have to mesh together in harmony to make the final product. In several instances, the output of one individual is the input for another, and of course the finished product of one phase is the raw material for the next.

This process of having the various parts of an information system built by different people in isolation will create a state of chaos if left without control. If each was left to do his/her job in the way he/she sees best, the different segments of the system, although correct on a stand-alone basis, they might not work together properly. The process of understanding each other's work and trying to build on it will be too hard and time consuming.

CASE technology addressed this problem by standardising systems development procedures so that each team member is able to understand the outputs of his colleagues, and at the same time produce an output that is accurate and proper in terms of fitting within the overall context of the entire endeavour. This support for standardisation was evident in Jones and Arnetts' (1992) work. CASE usage enabled the creation of accurate and complete system documentation that is widely understood within the IT community. Instead of creating the system documentation after the system is completed, allowing for inconsistencies and ambiguities to occur, the process was performed continuously throughout the SDLC.

In the sample studied by Aaen et al (1992), 72% of the respondents perceived CASE to have significantly improved systems development procedures and increased standardisation. Kuster and Wijers (1993) also found that CASE technology have served in enhancing the quality of documentation as well as ensuring the consistency, standardisation, and adaptability of the developed products.

These characteristics have a tremendous effect on the maintenance phase discussed earlier. They also have a big impact on communication within the development team and the organisation in general. Post, Kagan and Keim, (1999) reported that CASE technology was used to share information and data as well as co-ordinate tasks among the different groups. The technology enhanced communication between developers themselves and between developers and users. Prototyping for example allowed for better mutual understanding of what the final product should be for both the developers and the user community (Selamat and Rahim, 1996).

On the other hand, network versions of CASE that support multiple developers enabled them to share the same data dictionary and provided access to updated user requirements in a timely fashion (Nelson and Rottman, 1996, Martin, 1995).

e. The cost of CASE technology is considerably high. The tool itself does not make the bulk of the price. It is the supporting hardware, human training, and consulting costs that makes the technologies' initial cost very high. Another source of cost is the relatively long learning curve associated with CASE, which is the time required for the organisation to fully understand and master the various capabilities and functions of the technology. During this period, the general performance of the organisation tends to decrease adding to the overall technology adoption cost.

Having said this, once the organisation recuperates from the learning curve, some researchers suggest that the actual cost of systems development will decrease (Feuche, 1989, Martin, 1989, Norman and Nunamaker, 1989). A major source for system development cost reduction is the maintenance phase (Selamat and Rahim, 1996). The reduction of maintenance time and effort required brings down the entire system development cost for the organisation. The fact that this cost reduction factor usually goes unnoticed is twofold:

1. The initial investment in CASE technology is very expensive, which increases systems development costs for the organisation.
2. The technology cost reduction effects are only evident on the long run during the maintenance phase, which is way after the system is completed.

These factors make it hard to view CASE as cost reducing technology. Yet the fact is, on the long run, it is indeed (Nelson and Rottman, 1996, Selamat and Rahim, 1996, Feuche, 1989, Martin, 1989, Norman and Nunamaker, 1989).

The Case Against CASE

There are many controversies surrounding CASE technology. Throughout the literature, many empirical studies and research applauded the technology for its positive contributions and enhancements to systems development. On the other hand, numerous articles as well as empirical studies reported negative results, and even went as far as considering the technology a failure.

There was a wide view among researchers that CASE did not meet expectations (Nelson and Rottman, 1996). The technology "gained a reputation of not fulfilling its promises" (Post, Kagan and Keim, 1999). Even the improvements introduced by CASE were not those expected (Flynn, Vangner and Vecchio, 1995, Finlay and Mitchell, 1994). CASE users felt that the tool did not provide as much support as they expected (Guinan, Cooprider and Sawyer, 1997).

One explanation for this dissatisfaction with the technology is that CASE comes in with a high profile. Prospectus users usually develop inflated expectations in terms of the tools' contribution to systems development, thus resulting in disappointment (Coupe and Onodu, 1997, Nelson and Rottman, 1996). This explanation is strongly supported by Finlay and Mitchell (1994) who concluded that there are major differences between perceived and actual CASE outcomes. These exaggerated expectations played a major role in many CASE failure scenarios. Subramanian and Zarnich (1996) emphasised the organisations' need for realistic short-term and long-term expectations in order for the technology to be properly and fully utilised.

Although there were many accusations to CASE throughout the literature, all the fingers pointed more or less at one or more of three areas; inadequate tool functionality, cost, and associated change required.

Inadequate tool functionality

Inadequate tool functionality addresses the tools' low support for certain systems development functions. Researchers reported low levels of Task-Technology Fit for CASE (Dishaw and Strong, 1998). Previous research concluded that there were five main functions for which CASE did not provide adequate support:

a. Productivity: contrary to research claims that CASE usage improves systems development productivity, many studies concluded that there were no solid productivity enhancements associated with the tool. In their empirical study, Flynn, Vangner and Vecchio (1995) reported that CASE failed to deliver the productivity improvements it promised. Similar results were also reported by Guinan, Cooprider and Sawyer (1997), and Lewis (1996). Absences of significant effects on productivity were also reported by other researchers (Yellen, 1990, Lempp and Lauber, 1988, Card, McGarry and Page, 1987).

   One explanation for such findings was the long learning curve associated with CASE[3] (Post, Kagan and Keim, 1998). The technology defenders argued that CASE effects

[3]To be discussed in a later section

on productivity were studied in a premature stage, way before users gained sufficient knowledge of the tool.

1. CASE enhancements to the quality of the developed systems were also criticised. While some researchers indicated very limited quality improvements (Eriksen and Stage, 1998, Vessey and Sravanapudi, 1995, Yellen, 1990, Lempp and Lauber, 1988, Card, McGarry and Page, 1987), others reported no effect on the quality of the developed systems. In one study, higher levels of design quality were associated with lower levels of CASE usage (Guinan, Cooprider and Sawyer, 1997). The same arguments of long learning curve and insufficient technology experience were used to downplay these findings.

2. Although to a lesser extent, some also pointed at the delivery time required for CASE assisted projects. Studies suggested that CASE technology did not cut down systems delivery periods as promised (Guinan, Cooprider and Sawyer, 1997) (Case, 1985). They argued that the time invested in the analysis phase exceeded the time saved during later stages, thus, increasing the total time required for the project.

   There are two explanations for the increase in the required development time;

   a. For development teams with well established processes and good knowledge of structured methods CASE did not increase the delivery time. It was users with ad hoc development processes that suffered a state of chaos when CASE was introduced, resulting in an increase in the delivery time required (Guinan, Cooprider and Sawyer, 1997).

b. If reductions in the time required for the maintenance phase, which is way after the system is completed, were taken into consideration, the total time would be shorter. In other words, on the long run, CASE shortened the time required for systems development.

3. Integration and flexibility: several CASE tools were criticised for their lack of integration. Martin (1995) suggested that there is a major integration problem between lower and upper CASE tools. He argued that the transition from back-end to front-end CASE is not practical due to lack of proper tool communication abilities. CASE ability to work well with other systems development tools was also criticised (Ramsey, 1996). Post, Kagan and Keim (1999) concluded that CASE did not provide a practical method for sharing designs or data stored in its data dictionary with other applications. In a study conducted by Flynn, Vagner and Vecchio 1995) more than 60% of the organisations surveyed indicated the need for better integration between CASE tools and other methods or systems development techniques.

   Another source for lack of flexibility was CASE support for a particular methodology. Some researchers considered this a limitation of creative systems development, especially with tools adopting a restrictive methodology support approach (Bulc, 1992, Windsor, 1986).

4. CASE lack of support for co-ordination was considered by many the most disabling weakness. Guinan, Cooprider and Sawyer (1997) reported that many CASE tools "have great limitations in their aid to team work and co-operation". The study found that higher group co-ordination was associated with lower CASE usage. The same relation between co-ordination and CASE usage was concluded by Sharma and Rai (2000). In fact, most tools are designed mostly for single users. Except for few network versions of CASE, significant development is required in this aspect before such tools could provide effective collaborative support (Dishaw and Strong, 1998).

The technologies' support for co-ordination of tasks and information sharing among development teams is essential because the development process is conducted by multiple small teams. Studying several large development projects, De Marco and Lister (1987) found that developers spent 70% of their time working with others rather than individually. This implies that a CASE tool that does not support team co-ordination does not provide effective support 7 out of 10 times during the development process.

Cost

"Despite the tangible benefits, many companies cannot afford to invest the time, effort, and resources to implement CASE" (Nelson and Rottman, 1996). Although the price of the tool itself might be considerably affordable by many organisations, the overall expenses associated with adopting the technology are much higher (Huff, 1992). They can be as much as 5-8 times greater than the initial purchasing cost (Nelson and Rottman, 1996). The associated costs include:

1. Investment in hardware capable of running the CASE application properly.
2. Training for personnel that will use the technology.
3. Fees paid for consulting firms.
4. Cost of lost work time spent in training.
5. Low overall performance for the IT department caused by the introduction of a new technology. This will occur in early adoption stages.
6. Increase in number of working days required to complete development projects due to lack of experience.

These factors collectively, in addition to other unforeseen organisational factors make up the total cost of CASE adoption (Huff, 1992). Such high adoption costs were concluded by several researchers as a major barrier to CASE adoption (Nelson and Rottman, 1996, Flynn, Vangner and Vecchio, 1995, Martin, 1995, Huff, 1992).

Change required

There is more to CASE adoption than the introduction of a new technology. A substantial change in the way the firms do business is required along with the adoption process (Martin, 1995). A revolutionary change is needed in the way software development and maintenance is conducted (Ramiller, 1993, Windsor, 1986). Similar to other organisational change, a considerable level of resistance develops within the organisation, thus, jeopardising the successful adoption of the technology. In many cases, organisations have either steered away from CASE or abandoned it after a brief period due to such resistance (Nelson and Rottman, 1996).

A number of researchers in the software Engineering field suggest that CASE will completely replace the software developer (Darrow, 1992). It is ideas like this that ally fears among developers. It is only logical for them to show resistance to what they view as a direct threat to their job security. Those that don't perceive a direct threat soon realise that the tool is de-skilling them (Orlikowski, 1993, Chen, Nunamaker and Weber, 1989, Gibson, Snyder and Rainer, 1989). In other words, the developers feel that their hard-earned skills are no longer valuable, and they are incompetent in fully understanding the complex features of the tool (Flynn, Vangner and Vecchio, 1995, Rai and Howard, 1994).

Previous research supports the idea that adopting CASE requires significant changes in peoples' skill sets (Melone and Bayer, 1991, Wynekoop, 1991). Keen (1991) states that CASE "demands a major shift in IS professionals work skills, attitudes, and effort: they often show as much resistance to change as their users did when IS brought them new systems".

Trying to solve the de-skilling issue, organisations developed training sessions arranged with CASE consultants or vendors to help their developers in mastering the tool. But this too is not an easy process. "Developers need a substantial time to learn CASE products as well as the methodologies underlying these products" (Selamat and Rahim, 1996). Martin (1993) estimated the learning curve for CASE users to be at least 6 months. Many development projects fell way behind schedule and/or above budgets because of developers lack of experience in CASE usage, even after considerable periods of time dealing with the tool (Selamat and Rahim, 1996, Subramanian and Zarnich, 1996). Lewis (1996) argued that such high learning curves associated with CASE adoption played a major role in preventing organisations from considering the technology.

CASE ADOPTION

Contrary to what is reported by popular literature, CASE adoption is in fact at very low levels (Guinan, Cooprider, and Sawyer, 1997). Several empirical studies have reported alarming low levels of CASE technology adoption. Kemerer (1992) reported that 70% of the CASE tools are never used after one year of introduction, 25% are used by only one group, and the wide usage of the remaining 5% is not to capacity. Another study of 102 CASE user organisations in Denmark and Finland reported that less than 20% of the organisations could be classified as routine CASE users, and about half has used the tools for two projects or less. Furthermore, in the majority of the firms studied, less than 25% of the analysts had used CASE tools (Aaen et al. 1992).

In one market survey, Necco et al. (1988) noted that only 29% of the firms were using CASE. Similar findings of low adoption levels were also reported by several other studies (Dishaw and Strong, 1998, Flynn, Vagner and Vecchio, 1995, Russell, 1993).

Another side to the low adoption issue was the abandonment of CASE by organisations that tried, but failed, to adopt the technology properly. Out of the 40 CASE user organisations studied by Selamat and Rahim (1996), 13 did not develop any application using the technology. In fact, 10 out of the 40 organisations eventually abandoned CASE.

The researchers concluded that this premature abandonment of CASE is caused by management' expectations to reap returns on their CASE investment in the same fiscal year, where as mentioned previously, the main returns on CASE investment are realised on the long run.

Unlike other technologies, Chau (1996) suggests that CASE adoption is different in two ways:

a. The flexibility to switch back to the old system is much more restricted. This makes user satisfaction with the tool a critical productivity factor.
b. "CASE is a system for systems people". The change agents are themselves required to change. Keen (1991) stated that "it is not the software but the human side of the implementation cycle that will block the progress in seeing that the delivered systems are used effectively."

From an organisational perspective, Guinana, Cooprider, and Sawyer (1997) stated that successful adoption of CASE can happen only under the proper set of conditions, or else the tool can actually hinder application development productivity. Yourdon (1989) listed the following organisational criteria as predictors of successful CASE adoption:

a. "The organisation does a significant amount of new application development work.
b. The systems being developed are large, complex, and vital to the success of the organisation.
c. The organisation is accustomed to organising its large, complex projects into life cycles.

d. Management is concerned about the MIS function productivity" (Martin, 1995, Yourdon, 1989).

Primary and secondary CASE adoption units

Other researchers looked into CASE adoption by analysts within the user organisations. Findings in this area also indicated low levels of CASE adoption. In one study, less the 24% of the firms analysts were using CASE on a more than casual basis (Martin, 1995). Kusters and Wijers (1993) also found that in the majority of CASE user organisations, less than 25% of the analysts actually used the tool on a regular basis.

Martin (1995) argued that there is a difference between adoption and implementation. He stated that adoption "gauges the degree to which a technology has been introduced into the organisation". So based on this definition, adoption gauges the organisations' adoption of the technology. On the other hand, implementation assesses the degree of usage of the technology that has been achieved. So it measures the users' adoption of the technology.

This implies that CASE adoption has two levels. The organisation, which is the primary adopting unit, and the users of the technology which are the secondary adopting unit.

The adoption levels in these two units are interdependent. An organisational decision to adopt the technology will affect the analysts' usage of the technology. On the other hand, analysts recommending the technology to their superiors can influence management to invest in it.

Leonard-Barton and Deschamps (1988) suggest that after the organisational adoption decision, successful implementation still depends on numerous individual secondary adoption decisions by target users. White and Nelson (1990) also asserted that it was the individual user who must adopt the technology for successful implementation.

Concluding from their review of previous literature, Flynn, Vagner, and Vecchio (1995) stated that "CASE tools have reached every IT department. The fact that they are not used is therefore a deliberate choice". This implies that organisations have somehow opened the door for CASE technology, but the tool was not welcomed by the users.

This makes utilisation the practical problem faced by organisations (Dishaw and Strong, 1998). The aim of this research is to study factors that affect CASE adoption and usage by systems analysts within CASE user organisations.

Nevertheless, it is of vital importance to establish the fact that in order to be able to measure the level of CASE usage by IS professionals, and try to relate its variance to the hypothesised factors, CASE usage should be on a voluntary basis. In other words, the respondents should have freedom of using or not using CASE technology within their organisations. The voluntary CASE usage criterion is a fundamental concept for the scope of this research.

MODEL BUILDING

Technology Acceptance Model (TAM)

Building on the assumption of a positive relation between technology usage and performance, several empirical studies investigated individual, organisational, and technological variables in order to isolate key factors determining information technology usage behaviour. One of the most practical and influential models was the Technology Acceptance Model (TAM) developed by Davis (1989) (Figure 1).

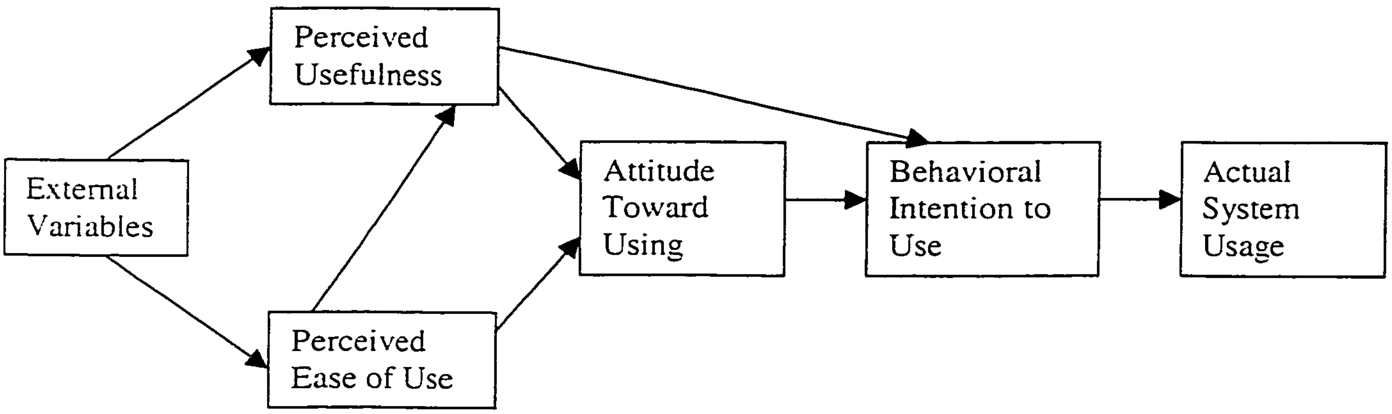


Figure 1: Technology Acceptance Model

The aim of TAM was to "provide an explanation of the determinants of computer acceptance that is general, capable of explaining user behaviour across a broad range of end-user computing technologies and user population, while at the same time being parsimonious and theoretically justified" (Davis, Bagozzi, and Warshaw 1989).

It was built on the grounds of the Theory of Reasoned Action (TRA) developed by Fishbein and Ajzen (1975). TRA suggests that beliefs influence attitudes, which in turn leads to intentions, which then generate behaviours. TAM adapted this belief -attitude-intention- behaviour relationship to model user acceptance of IT. It included two constructs, Perceived Ease of Use and Perceived Usefulness.

According to Davis (1989) perceived ease of use is "the degree to which a person believes that using a particular system would be free of effort". The other construct, perceived usefulness, was defined as "the degree to which a person believes that using a particular system would enhance his or her job performance" (Davis 1989).

Bridging several theories and models, such as self-efficacy theory, expectancy theory, cost-benefit research, and channel disposition model, TAM theorised that Perceived Ease of Use and Perceived Usefulness, which are determined by external factors such as system features, training, and documentation, will influence a persons' Attitude Toward Using a system. Both Perceived Usefulness and Attitude Toward Using the system will influence the person's Behavioural Intention to use the system, which will in turn influence the actual system usage. The model also theorised that Perceived Ease of Use positively influences the Perceived Usefulness of a system.

The model was empirically tested and results showed that perceived usefulness have a strong influence on people's intentions, while perceived ease of use had a smaller but still significant effect.

The validity of TAM was also tested in comparison with another model built on the Theory of Planned Behaviour (TPB) which anticipated users' intention to use a certain technology. Although the study found that both TAM and TPB provided very good predictions of an individuals' intention to use an information system, the researchers noted that TAM had some empirical advantage as well as being easier to apply.

The reliability of the two constructs, perceived usefulness and perceived ease of use was also tested by Hendrickson et al. (1994). The study showed that TAM exhibited a high degree of test-re-test reliability. Several other empirical studies examined the reliability and validity of TAM and their findings showed varying levels of support to the model (Segars and Grover 1993, Adams, Nelson, and Todd 1992, Moore and Benbasat 1991).

TAM Modification

The proposed model is built on the Technology Acceptance Model developed by Davis (1989). Based on the literature review and the findings of previous studies however, key modifications were made to the TAM by adding two new constructs, Non-CASE System Development Experience and Management Support, and dropping another two, Attitude Toward Using the system and Behavioural Intention to use the system.

Perceived Usefulness:

In their empirical study, Dishaw and Strong (1998) concluded that the low CASE tool usage was due to a low Task-Technology fit between CASE capabilities and the required maintenance and co-ordination activities of IS. Nelson and Rottman (1996) also reported that users were disappointed by CASE because it does not support their methods.

This suggests that there is a positive relation between the users' perceived usefulness of CASE technology and the level of CASE usage, which is strongly supported by the Technology Acceptance Model. Therefore, the proposed model hypothesise that the users perceived usefulness of CASE technology is positively related to CASE usage.

H1: Perceived Usefulness has a direct positive effect on CASE usage.

Perceived Ease of Use:

Previous studies suggest that the tremendous effort required to learn a CASE tool has been a major cause for avoiding its adoption. According to Chau (1996), Ease of Use had the strongest total effect on CASE acceptance. Kemerer (1992) also concluded that CASE complexity played a major role in the tools' low usage by system developers. More recent studies similarly indicated that tool complexity and the high learning curve associated were main reasons behind the low CASE tool adoption by developers as well as organisations (Iivari, 1996, Lewis, 1996).

Tornatzky and Klein (1982) reported that the complexity of an innovation, that is the degree to which an innovation was perceived as relatively difficult to understand and use, always had a significant effect on its adoption. The proposed model hypothesis that the Perceived Ease of Use is positively related to CASE usage.

H2: Perceived Ease of Use has a direct positive effect on CASE usage.

Management Support:

This construct is one of the modifications to the Technology Acceptance Model. For the purpose of this research, Management Support is defined as the degree of support and commitment to CASE technology exerted by the management. Previous studies indicated that the degree of management involvement and support to any IT implementation has a significant effect on the technologies' successful implementation and adoption by the prospect users (Orlikowski, 1993, Van de Ven, 1986, Zmud, 1984).

Chau (1996) indicated that management support have significant effects on the technologies' adoption and recommended that it should be studied in future research. In fact, Selamat and Rahim (1996) identified lack of management support as a major problem in CASE adoption. They argued that MIS management did not like to find their developers spending their time learning CASE instead of performing their normal daily functions. They also noted that inadequate management support was a major obstacle in CASE implementation in the US, UK, and Malaysia. Their conclusions are inline with Loh and Nelsons' (1989) findings, which also attributed CASE failure to lack of management support.

On the other hand, Simons (1992) indicated that Canadian organisations that successfully adopted CASE had a strong management commitment to the technology. Management provided a full-time on-site technology consultants during the first year of CASE implementation.

The above-mentioned arguments provide reasonable grounds to hypothesise that Management Support is positively related to CASE usage.

H3: Management Support has a direct positive effect on CASE usage.

Experience:

This construct is also one of the modifications made to the Technology Acceptance Model. It refers to the level of non-CASE experience of the users. The phrase "non-CASE" is used in the context of this research as acronym to traditional systems development methods used by IS professionals. In other words, an IS professional with high non-CASE experience is one who has high traditional systems development methods experience. On the other hand, IS professionals with low non-CASE experience are those whose have low traditional development methods experience.

It is important to note however, that this construct measures only the level of traditional methods experience possessed by the respondents. It does not capture the levels of CASE experience of the respondents.

Previous empirical studies on CASE adoption and implementation pointed out that introducing CASE into an organisation requires a major shift in the kind of skills and functions required by system developers. Keen (1991) indicated that "CASE demands major shifts in IS professional work skills, attitudes, and efforts". Adopting the technology will dictate new skills and knowledge from its users (Chau, 1996). A revolutionary change in system development and maintenance practices is required

(Ramiller, 1993). System developers will have to unlearn old practices (Iivari, 1996, Melone and Bayer, 1990).

These fundamental changes brought about by CASE introduction created a negative attitude toward the technology by its potential users. Selamat and Rahim (1996) mentioned that users faced with CASE realised that their hard-earned experience would be challenged. Others warned that CASE is perceived by developers as a de-skilling technology, limiting their creativity, and considered it a threat to their job security (Orlikowsky, 1993, Chen and Nunamaker, 1989, Gibson et al, 1989). According to Keen (1991), "IS professionals often show as much resistance to change as their users did when IS brought them new systems".

This effect of experience on CASE acceptance is very well supported by Chau (1996), he stated that "experience has been suggested and found to be an important factor on user acceptance of an information system". A strong direct relation between the degree of traditional systems analysis and design skills and the respondents' attitude toward CASE was concluded by Yang (1997).

The negative relation between non-CASE experience and the attitude toward CASE was emphasised by Post et al. (1998). In their conclusion, they stated that "the sign of the coefficient is negative, that is, more experienced developers tend to have lower opinions of the value of the CASE tool".

Hence, based on previous empirical support, the proposed model hypothesis a negative relation between the users' non-CASE experience and their CASE usage.

H4: Non-CASE experience has a direct negative effect on CASE usage.

Usage:

Usage is the dependent variable hypothesised to be determined by the four independent variables, Perceived Ease of Use, Perceived Usefulness, Management Support, and Non-CASE Experience. This construct is designed to assess the level of CASE technology adoption by measuring the developers/ analysts' usage of the technology. It is directly adopted from the Technology Acceptance Model as it reflects the usage level reported by the developers/ analysts themselves.

Although some researchers argue that there exist some discrepancies between actual and self-reported usage, these discrepancies are minimal and considered negligible within the context of this research for the following reasons:

- There are no solid logical motives for the users to deliberately misreport their usage of CASE technology, especially when CASE usage is not strictly coercive in their work environment.

- The line between using CASE and not using it is very well defined. Users can clearly recognise when they are using CASE and when they are not. This rules out, or at least significantly minimises the risk of unintentional mis-reporting of usage, which makes the users a credible source of information.

- Although measuring the actual usage of CASE for a small sample of developers is doable, it is practically impossible to measure the actual usage of CASE for all users involved in this research. This makes self-reported usage the best practical way for such a study.

Attitude Toward Using and Behavioural Intention to Use:

Theses two constructs link perceived usefulness and perceived ease of use to usage in the Technology Acceptance Model. For the purpose of this research these two constructs will be dropped.

One reason for this modification is that the path from Attitude Toward Using a certain technology to the Behavioural Intention to Use the technology, and from the later construct to Usage of the technology has already been proved and strongly supported by several empirical studies (Chau, 1996, Davis, 1989, Davis et al, 1989). There is no added value from reinventing the wheel.

Another reason for dropping the two constructs is embedded in the intent of this research, which is to identify factors that affect CASE adoption and usage by system developers. The main focus is on the four independent variables that affect usage. The mechanism through which they affect usage has already been studied as mentioned earlier, and is considered beyond the scope of this research.

As a result of dropping the Behavioural Intention to Use construct, the path from Perceived Usefulness to Behavioural Intention to Use the technology will disappear. It is now implied in the hypothesised path between perceived usefulness and usage.

TAM also supported a direct relation between Perceived Ease of Use and Perceived Usefulness. One explanation for this path was that "efforts saved by the improved ease of use may be re-deployed, thus enabling a person to accomplish more work for the same effort" (Chau, 1996). Davis (1989) stated that the regression results obtained "suggest that ease of use may be an antecedent to usefulness".

Hence, the proposed model hypothesis a direct relation between perceived ease of use and perceived usefulness.

H5: Perceived Ease of use has a positive direct effect on Perceived Usefulness.

The proposed model

The proposed model constitutes of four parallel independent variables, Perceived Ease of Use, Perceived Usefulness, Management Support, and Non-CASE experience. Perceived Ease of Use and Perceived Usefulness are hypothesised to have a positive effect on CASE usage. Management Support is also hypothesised to have a positive direct effect on CASE usage. On the other hand, the fourth independent variable, Non-CASE Experience, is hypothesised to have a direct negative effect on CASE usage.

The fifth hypothesise suggests a direct relation between the two independent variables perceived ease of use and perceived usefulness. The proposed model is depicted in Figure2.

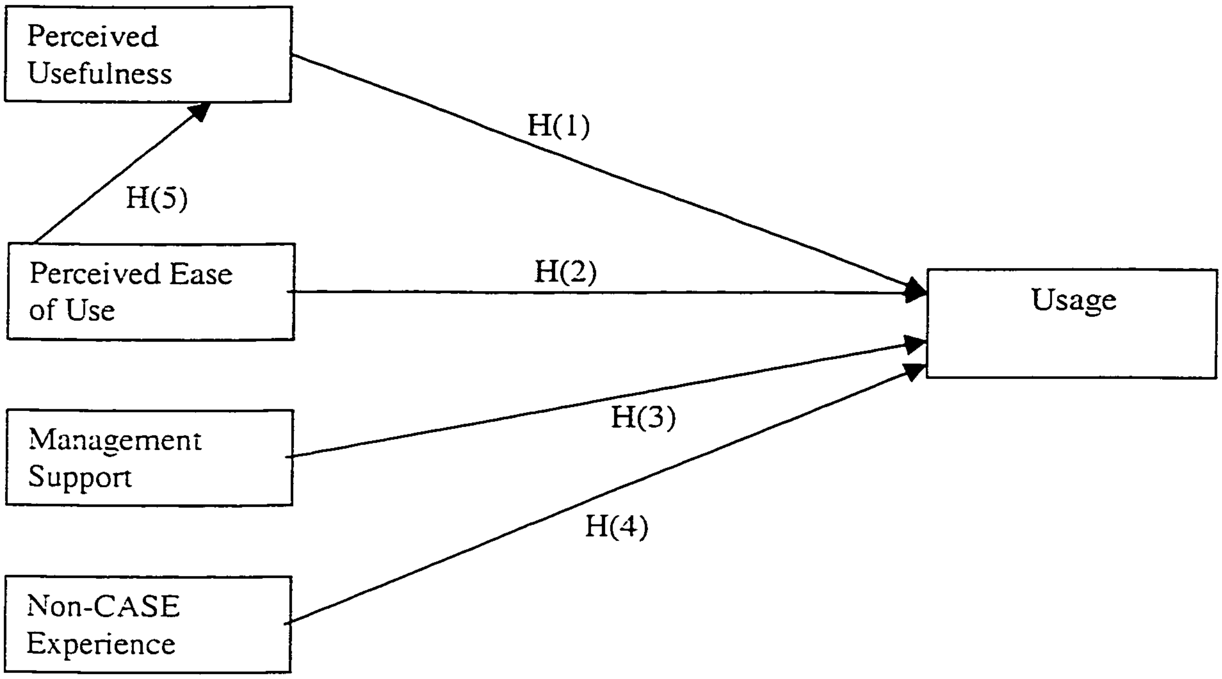


Figure 2: The Proposed Model

RESEARCH DESIGN

Instrument Development

Since the proposed model was built on the Technology Acceptance Model (TAM), measurement of constructs used in the original TAM was adopted from Davis's (1989) questionnaire with slight rewarding in order for the questions to fit the purpose of this research. Both constructs, Perceived Ease of Use and Perceived Usefulness were measured through six questions for each construct (Appendix 1).

Chau (1996) studied Management Support as a factor affecting CASE acceptance by system developers. He included four items in his questionnaire to measure Management Support. Out of those four, three items were used in developing this instrument.

The respondents' level of Experience was measured by Iivari (1995) in two ways. One was the number of years of experience each respondent had. The other was the number of relevant projects the respondent was involved in throughout his career.

Another method of measuring the respondents' experience was the level he/she occupied within the departmental structure in his/her organisation. Orlikowski (1993) used this method and argued that the level or title of the respondent, which ranges from junior levels up to senior management, provides a good indication of the experience possessed by the individual. The logic behind her argument was derived from the notion that for the individual to achieve a certain level in the organisational structure, he/she must have climbed up the organisational ladder gaining experience along the way. Hence, higher levels would have more experience than lower levels.

In order to measure the level of the respondents' experience, the instrument collected data on the number of years of experience, the number of relevant projects they worked on, and where the respondents fell on the departmental structure of their organisations.

The dependant variable, Usage, which refers to the respondents' level of CASE usage, was also adopted from Davis (1989). Davis resorted to self-reported usage and argued that it reasonably indicates the actual usage of an information system. Several other studies used self-reported usage and supported Davis's claim that this method provides a clear indication of the respondents actual usage due to the high correlation between the two (Baroudi et al. 1986, Kim and Lee, 1986, Rodriguez, 1981, Lucas, 1974).

Pre-test and Revision

As the measurement of constructs was adopted from other studies that used these constructs in a different context, it was deemed necessary to test the developed questionnaire before administering it to respondents in order to enhance the reliability and validity of the collected data (Emery and Cooper, 1991).

The developed questionnaire was sent via e-mail to 6 IS professionals that were asked to review it and respond with their feedback. Three major points were raised. One was the ambiguity of some questions, which could mislead the respondents or cause them not to answer such questions. Each of the items in question was reworded in order to eliminate any ambiguity. The reworded questions were shorter, more direct, and hence, easier to understand.

The sequence of the questions was also raised. Questions measuring the same construct were put together in the initial questionnaire. This might introduce some bias in the way the respondents answer the questions so that they match. To correct this, the reworded questions were re-arranged so that questions which measure the same construct are far apart.

The third major point raised by the pre-test had to do with the method of instrument administration to the respondents. This will be discussed in the following section, Method of Instrument Administration.

Method of Instrument Administration

The initial questionnaire administration method was e-mail based. Target respondents were to receive the questionnaire as a Word attachment file via e-mail, answer the questionnaire, and e-mail it back. The process would require each respondent to go through the following six steps:

1. Download the questionnaire that is attached as a Word file.
2. Start MS Word Application.
3. Answer the questionnaire.
4. Save the answered questionnaire in a known directory.

5. Start up the e-mail application.
6. Attach the answered questionnaire as a Word file and send the e-mail.

The pre-test indicated that such an administration method is lengthy and could discourage participation. Other than that, the IS professionals suggested that due to widespread computer viruses, people usually hesitate to download attachments from unknown sources. Such an attitude could severely decrease the response rate.

In order to avoid these pitfalls, a web-based questionnaire administration method was developed. The questionnaire was posted on a web-site www.casethesis.com (Appendices 2a and 2b). Target respondents would be asked to access the web-site, answer the questionnaire, and submit their answers by clicking on a button on the screen. The new method requires respondents to go through fewer steps, which are:

1. Click on the hyperlink www.casethesis.com.
2. Click on the "Continue" button to answer the questionnaire.
3. Answer the questionnaire.
4. Click on the "Submit" button.

In addition to having fewer steps, the web-based method is simpler in that all the respondents have to do is use their mouse to access the web-site, answer the questionnaire, and submit their answers. Keyboard use is not required. It also eliminates the risk of respondents not participating in fear of computer viruses because there are no attachment files involved in this method.

Upon submission of the answers, the web-site automatically e-mails the answers to a pre-defined e-mail address, which forms the initial database.

Since the web-based data collection method is a deviation from the traditional post-mail data collection method which is usually used for this kind of research, the method was verified with Professor Linda Dyer, a professor in the Msc. in Administration program at Concordia University. Professor Dyer indicated that there is no inherent bias in electronic data collection, given that respondents are reassured of the confidentiality of the information they provide (Appendix 3).

The same IS professionals were contacted again and were given the hyperlink www.casethesis.com. Again they were asked to re-evaluate the questionnaire as well as the administration method. They all reported their satisfaction with the changes. This final verification also served as a test of the functionality of the web-site in terms of e-mailing the submitted responses to the pre-determined e-mail address (Appendix 4).

The final test was to ensure that the answers received from the web-site are the same answers submitted to the web-site. Five pre-answered hardcopies of the questionnaire were inputted to the web-site at one-hour intervals apart. Each time, the initial database was accessed and its contents matched against the responses submitted. The site passed all five sub-tests. The database was then cleared of all testing data in preparation for receiving the actual research data.

Sampling

Several associations within the Information Technology industry were contacted and asked if they could provide a list of e-mail addresses of their members (See Appendix 5 for the Script). Yet, due to Member Information Confidentiality agreements that such associations have with their subscribers, such a list could not be obtained. Nevertheless, one of the associations, Canadian Information Processing Society (CIPS) offered to include three paragraphs in their weekly electronic newsletter describing the research and asking the members to participate (Appendix 6). The hyperlink www.casethesis.com would be included in the last paragraph as the link between the newsletter and the questionnaire.

Although such a sampling technique is considered less aggressive in that the questionnaire is not personally directed to each respondent, the large target population, which exceeds 5000 subscribers, was considered satisfactory to offset any decrease in response rate due to such indirect method. Thus, a letter describing CASE, the aim of the research, and asking the members to participate was sent to CIPS. The last paragraph included the confidentiality clause as recommended by Professor Dyer as well as the hyperlink www.casethesis.com. The letter was edited and posted in CIPS electronic newsletter (Appendix 7).

Variables Measurement and Coding

Since all constructs were adopted from previous researches, the same measurement method used to measure each construct in those researches was used in this study. This was done to ensure the validity of the scale (Appendix 8).

Each item was coded using a Likert scale ranging 1-7 or 1-5 depending on the probable answers provided in the data collection instrument. The coding scheme used the code 1 to indicate the most positive attitude toward the specific item, and the code 7 to indicate the most negative attitude toward the item.

Items used to measure the Experience construct, which used a Likert scale ranging 1-5, also applied the same scheme where the code 1 indicated high level of experience and code 5 indicated low level of experience. The entire coding scheme for all the variables is depicted in the questionnaire coding directory (Appendix 9).

Items concerning descriptive data about the study sample characteristics, which are Age, Type of Industry, and Number of Employees were also coded using Likert scales ranging 1-7. Item 26, which measures the Education Level of the respondents was coded 1-6 since the respondents had six categories to choose from. The coding scheme for the descriptive items is also depicted in the questionnaire coding directory (Appendix 9) under Sample Characteristics.

DATA ANALYSIS

Data cleaning and treatment

As mentioned previously, the questionnaire was posted on a web site designed specifically for data collection purposes. A description of the research containing a hyperlink to the questionnaire web site was posted for one week in the CIPS electronic newsletter. The total response added to 73 sets of data. Nevertheless, out of the 73 responses, only 49 were adequate in terms of CASE usage policy adopted by the respondents' organisations[4].

One reason for the low total response achieved was due to the fact that the posting of the research in CIPS newsletter coincided with the Christmas holidays. It is logical to relate the low response to the fact that during holiday periods people tend minimise contact with issues relating to their jobs, such as reading professional newsletters related to their jobs.

To overcome this hurdle, a letter was sent to CIPS via e-mail asking them to post the request for participation in the study for another week. This measure proved to be sufficient, yielding a total response of 172 sets of data.

---

[4] See "Primary and Secondary adoption Units" P. 32

Again, out of the 172 responses, only 109 met the voluntary CASE usage criterion referred to earlier. Given the exploratory type of this study, this sample size was considered sufficient to conduct the research. Hence, it was decided to proceed with the analysis of the collected data.

Upon reviewing the 109 sets of data, only two sets were found to contain missing values. Although their exists a statistical technique to replace the missing values, it was decided to eliminate these sets for the following reasons:

1. The two values referring to the level of CASE Usage (the dependent variable) were among the missing values in both sets, and all three values referring to Management Support were missing in one of them.

2. The number of sets with missing values is a very small fraction of the total number of data sets (less than 2%).

Data coding and standardisation

The remaining 107 sets of data were inputted into a SPSS (statistical software) database according to the data coding directory (Appendix 9). The data was again checked for completeness to ensure that all 107 data sets were inputted without any missing values.

The validity of the each data item was also checked to ensure that no data item contained an invalid value. This was done by ensuring that variables that were measured using a certain Likert scale did not contain any values outside the proper range (i.e. If a variables was measured using a 1-5 Likert scale and displayed a value of 6, it would be considered invalid). Sets that contained invalid codes were deleted and re-inputted with the proper corresponding values.

The values of each variable were standardised using SPSS according to the following formula:

$$Z = (X - \mu) / \sigma$$

Where:

X is the value we want to normalise.
μ is the arithmetic mean of the distribution.
σ is the standard deviation of the distribution.

The standardisation of the data was done to ensure an accurate output of the analysis. It was also deemed necessary due to the fact that two different ranges of Likert scales were used, 1-7 and 1-5, in some cases to measure variables that assess the same construct (Neter et al., 1985).

Sample characteristics

Respondents were asked to provide information regarding their Age, Education level, Type of Industry they belong to, as well as the Number of employees in their organisations.

The study sample of 107 respondents ranged from ages 20 years to above 45 years old. The distribution of the respondents' ages closely followed the normal distribution. A close look at the numbers indicates that 77.5% of the respondents were above 30 years old. This is evident in the slight skewness of the distribution to the right (figure 3).

The average age of respondents ranged between 36 to 40 years old with a standard deviation of 6.24 years. The highest percentage of responses was 25.5% which came from IS professionals between ages 36 - 40 years, and the lowest was zero from the age group under 20 years. A histogram depicting the number of responses by each age group was developed to help visualise the age characteristics of the respondents (figure 4). The actual number of respondents from each age group is indicated in table 1.

Age group	Number of respondents	Percentage	Cumulative	
			Respondents	Percentage
Under 20	0	0%	0	0%
20 - 25	11	10.3%	11	10.3%
26 - 30	13	12.1%	24	22.4%
31 - 35	18	16.8%	42	39.3%
36 - 40	27	25.2%	69	64.5%
41 - 45	20	18.7%	89	83.2%
Above 45	18	16.8%	107	100.0%

Table 1: Description of respondents by age group

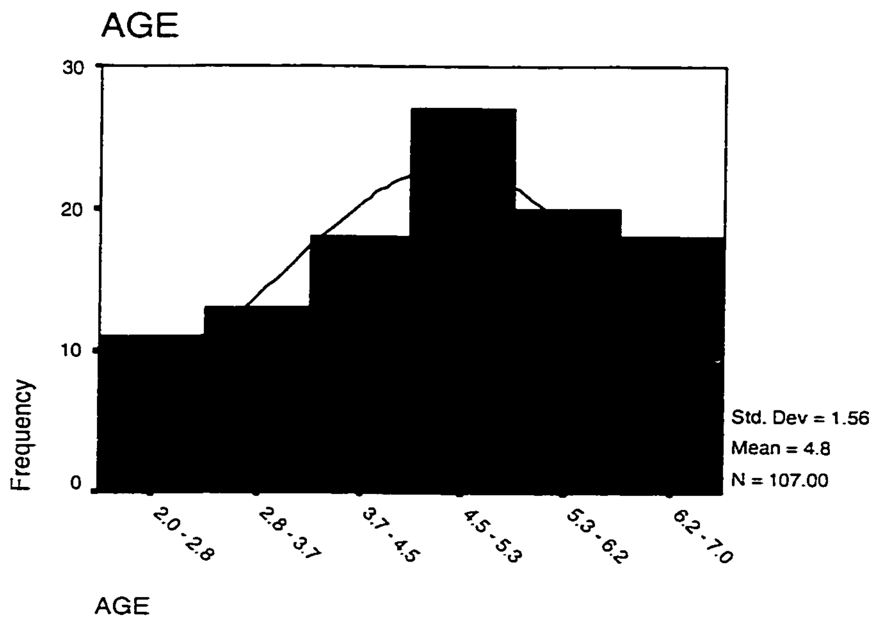


Figure 3: Distribution of respondents by age group

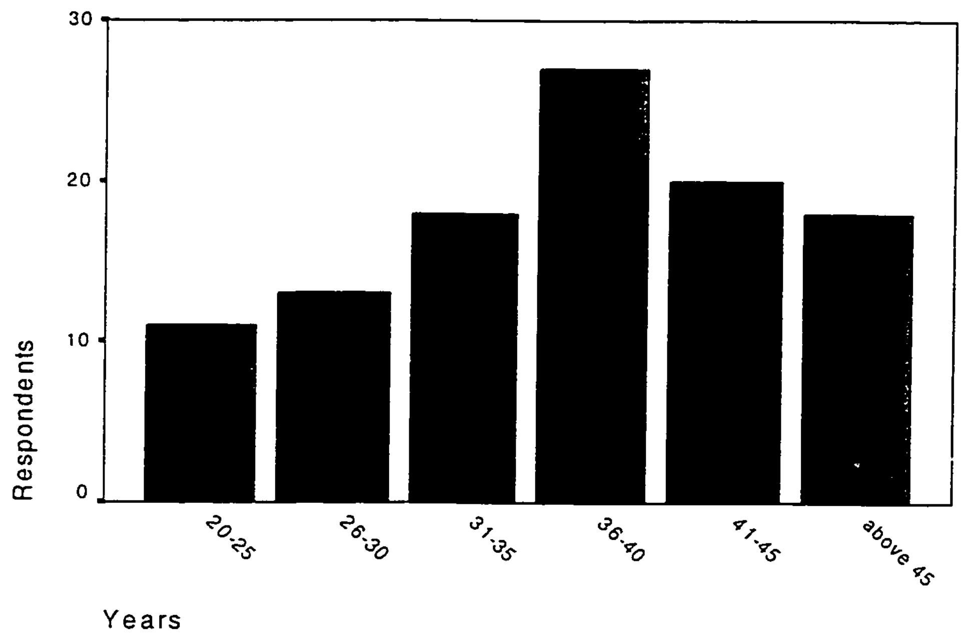


Figure 4: Number of respondents by age group

The respondents were also asked to provide information regarding their education. The questionnaire provided them with six categories to choose from, ranging from high school up to Ph.D. Nevertheless, none of the respondents fell in any of the two extreme categories. The lowest education level reported was CEGEP or equivalent with 15 respondents, which is 14% of the sample, falling in this category. The highest education level was reported by respondents with a Master degree, a total of 25 respondents, making 23.3% of the total respondents.

The majority of the respondents had a Bachelors degree. A total of 41 respondents, making 38.4% of the total study sample fell in this category. Respondents with a University certificate made 24.3% of the study sample, with a total of 26 respondents falling in this category (Figure 5).

The number of respondents with the same education level is depicted in table 2.

Education level	Respondents	Percentage	Accumulate	
			Respondents	Percentage
High school or less	0	0%	0	0%
CEGEP or equivalent	15	14%	15	14%
University certificate	26	24.3%	41	38.3%
Bachelors degree	41	38.4%	82	76.7%
Masters degree	25	23.3%	107	100%
Ph.D. degree	0	0	107	100%

Table 2: Description of respondents by level of education

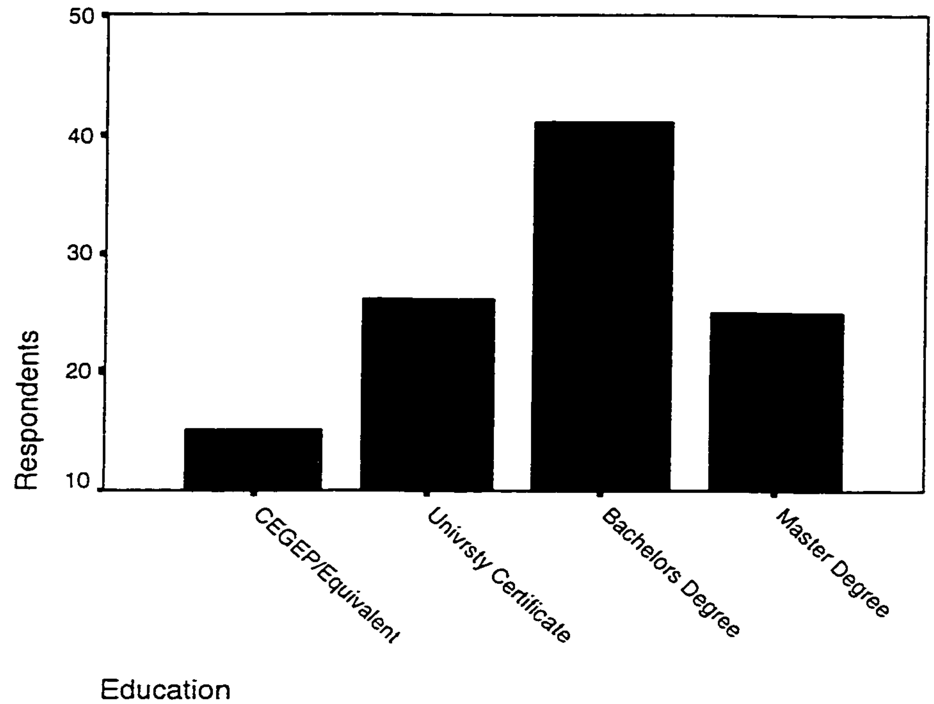


Figure 5: Number of respondents by education level

Although all of the respondents work within the information technology domain, they were asked to indicate the main type of industry their organisations belong to.

The study sample contained respondents from several different industries (Table 3). Yet, a majority of 42 respondents, 39.2% of the total sample, came from the Information Technology industry. Such an outcome was more anticipated than surprising, since all the respondents are Information Systems professionals. Also the fact that CASE is a tool created for Information Technology professionals, it is reasonable for such a tool to be more used within the Information Technology industry relative to other types.

The numbers of respondents belonging to Financial Services, Manufacturing, and the Public Sector were relatively close to each other with 19, 15, and 12 respondents from each industry respectively. Theses three industry types made 17.8%, 14%, and 11.2% of the total sample respectively.

The amount of respondents from Merchandising, Health Care, and other industry types were relatively low. A total of 7 respondents, 6.5% of the total sample, worked within the Merchandising industry, while Health Care and other industries contributed 5 and 7 respondents from each respectively.

One way to explain the relative low response rates from industries other than Information Technology is the reliance of these industries on outsourcing IT related services to IT specialised organisations. Such outsourcing of IT related services usually results in smaller IS departments within the outsourcing organisation, causing heavy IT related projects such as software development to be conducted outside the organisation.

A histogram showing the number of respondents from each industry was included to help visualise the response levels buy the different industries (Figure 6).

Industry	Respondents	Percentage	Accumulate	
			Respondents	Percentage
Information Technology	42	39.2%	42	39.2%
Financial Services	19	17.8	61	57.0%
Manufacturing	15	14%	76	71.0%
Public Sector	12	11.2%	88	82.2%
Merchandising	7	6.6%	95	88.8%
Health Care	5	4.6%	100	93.4%
Other	7	6.6%	107	100%

Table 3: Description of respondents by industry type

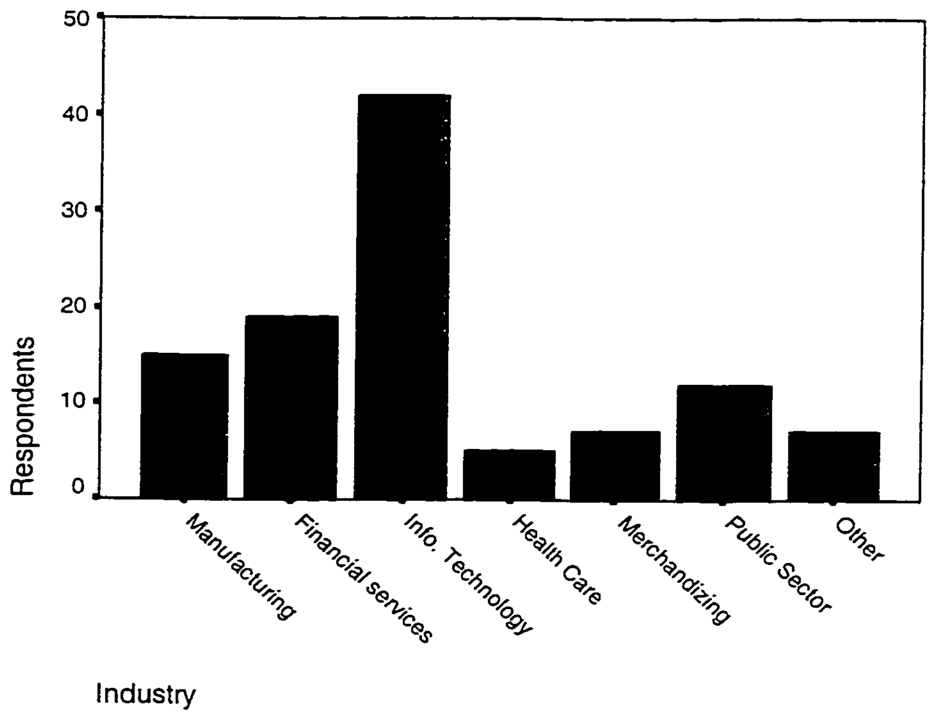


Figure 6: Number of respondents by industry type

The size of organisations the respondents belong to was assessed by the organisations' number of employees. Respondents were provided with seven categories ranging from less than 50 employees up to more than 4000 employees. A slight majority of 30 respondents came from big organisations with more than 4000 employees (Figure 7).

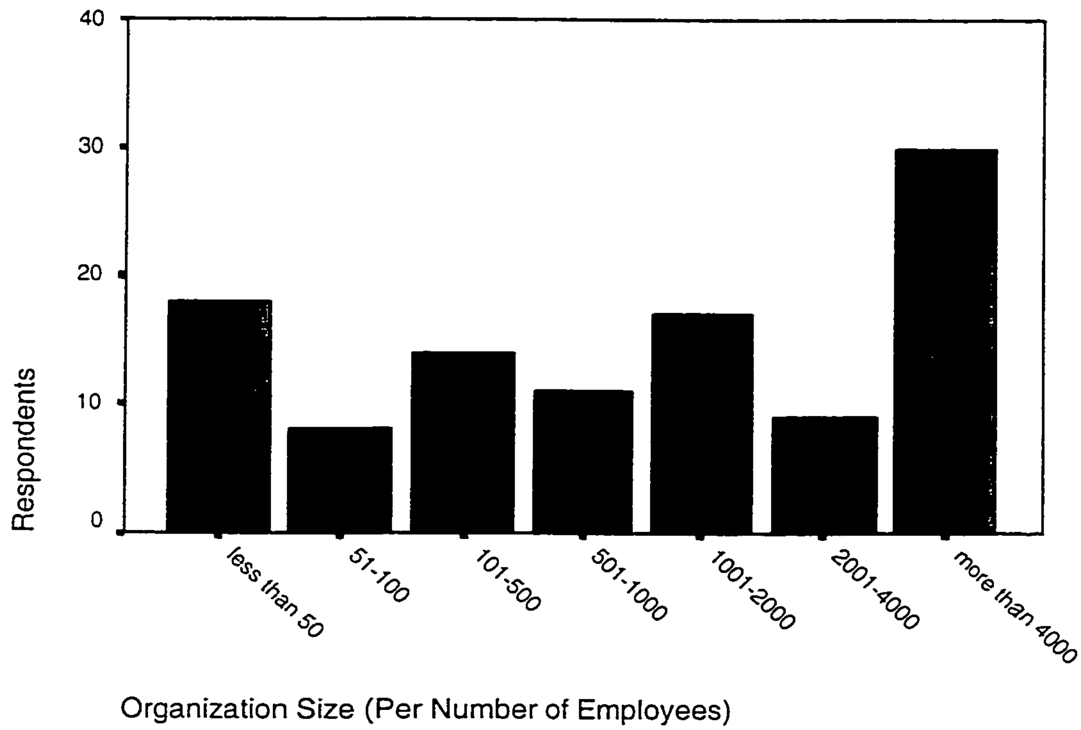


Figure 7: Number of respondents by organisation size (number of employees)

The numbers of respondents from each organisation size category are depicted in table 4. The table also provides the break down of each organisation size category by the industry type the organisation belongs to.

Organisation Size (# of employees)	Respondents	Percentage	Respondents per Industry Type						
			Manufacturing	Fin. Services	Inf. Technology	Health Care	Merchandising	Public Sector	Other
Less than 50	18	16.8%	0	2	12	0	0	4	0
51 -100	8	7.5%	0	1	2	0	1	0	4
101 - 500	14	13.1%	4	1	6	1	2	0	0
501 - 1000	11	10.3%	2	2	5	0	2	0	0
1001 - 2000	17	15.9%	1	5	4	4	0	0	3
2001 - 4000	9	8.4%	2	0	4	0	2	1	0
More than 4000	30	28%	6	8	9	0	0	7	0
Total	**107**	**100%**	**15**	**19**	**42**	**5**	**7**	**12**	**7**

Table 4: Description of respondents by organisation size (number of employees) crossed by industry type of the organisation

Preparing the data for Path Analysis

Reliability

The reliability of the data was verified using the reliability verification measure provided by SPSS (Appendix 10). Correlation coefficients for items within the same factor were all above 0.60 which indicates a high level convergent validity.

To further verify the reliability of the factors, Cronbach's Alpha was calculated for each factor (Table 5). A Cronbach's Alpha coefficient of 0.90 or higher is an indication of a very good level of reliability. Nevertheless, Alpha Coefficients higher than 0.80 is usually accepted as a good level of factor reliability. Alpha coefficients for all factors were above 0.80 indicating an acceptable level of reliability.

A correlation matrix of all twenty items was also created (Appendix 11). Correlation coefficients between items from different factors provide a good indication of discriminant validity. A high level of discriminant validity is obtained when correlations between variables in different factors are low. An item should have higher correlation with items from the same factor than with items from other factors.

Examining the correlation coefficients in the correlation matrix indicated that the items tend to have higher correlation within the same factor than with items in different factors.

Factor	Cronbach's Alpha Coefficient
Perceived Usefulness	0.9733
Perceived Ease of Use	0.9649
Management Support	0.8491
Experience	0.8373
CASE Usage	0.9421

Table 5: Cronbach's Alpha Coefficients (Measure of reliability)

Normality

"Multivariate normality is the assumption that each variable and all linear combinations of variables are normally distributed" (Tabachnick and Fidell; 1989). Tests of normality of the distribution include Skewness and Kurtosis.

Skewness measures the extent to which the distribution follows the normal bell shaped curve. .The closer the distribution is to normality the closer its skewness is to zero. A perfect normal distribution would have zero skewness where the mean, the median, and the mode of the distribution are equal.

The skewness of the distribution could be positive or negative. A positive skewness indicate that there are more cases to the left of the mean while a negative skewness indicates the presence of more cases to the right of the mean.

Kurtosis measures the peak of the distribution versus its flatness. The more cases in the middle of the distribution the more peaked it is. More cases away from the centre of the distribution will cause it to be more flat.

The normality of the data was checked using frequency histograms with a distribution curve. Skewness and Kurtosis were also calculated for each variable. In general, all variables had an acceptable level of normality. All Usefulness items as well as one of the items measuring Experience showed a slight positive skewness. Another item measuring Experience was also slightly peaked. Nevertheless, none of the above was serious enough to pose a threat to the normality of the data (Appendix 12).

Linearity

The test of linearity of the data was done using bi-variate scattter plots. This method entails that for the variables to have a linear relation the scatter plot should have an oval shape (Tabachnick and Fidell, 1989).

The examination of the scatter plots of the variables (Appendix 13) indicated that all of the variables closely follow the linearity assumption.

Multicollinearity

"When the predictor variables are...highly correlated among themselves, multicollinearity is said to exit. A formal method of detecting the presence of multicollinearity is the Variance Inflation Factor. VIF values in excess of 10 are frequently taken as an indication of serious multicollinearity. Mean VIF values considerably larger than 1 are indicative of serious multicollinearity problems" (Neter et al. 1985).

VIF values were calculated for all four independent variables (Appendix 14). All VIF values were below 3 and the mean VIF was 1.872, thus, indicating no multicollinearity problems.

Path Analysis

"The method of path analysis was developed to explain causal relations. Its goal is to provide plausible explanations of observed correlations by constructing models of cause-and-effect relations among variables" (Neter et al. 1985).

Path analysis method is made of two components, the path diagram, which makes the final model of causal relations between dependent and independent variables, and the path coefficients, which determine the strength and direction of the relations between each predictor and the dependent variable (Neter et al. 1985).

The construction of the path diagram is based on the following rules:

1. "A straight arrow is drawn to each dependent variable from each of its sources.
2. A curved, double-headed arrow is drawn between each pair of independent variables thought to have non-zero correlation" (Neter et al. 1985).

Since the proposed model has four independent variables that are hypothesised to affect one dependent variable the first step of constructing the path diagram is drawing straight arrows from each independent variable to the independent variable (Figure 8).

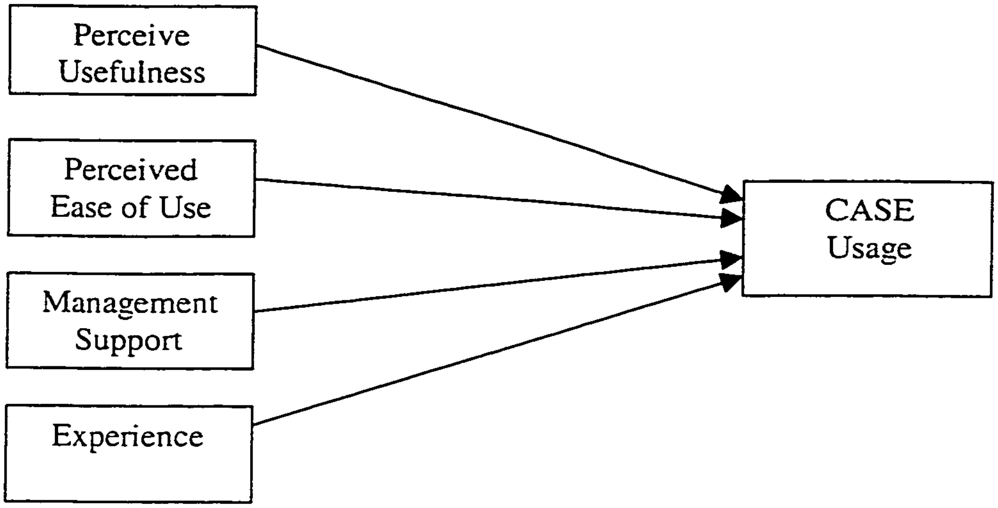


Figure 8: Path diagram construction

The proposed model also hypothesised a direct relation between the two independent variables perceived ease of use and perceived usefulness. To account for the path between the two independent variables, a curved double-headed arrow is drawn between them. This completes the construction of the path diagram (Figure 9).

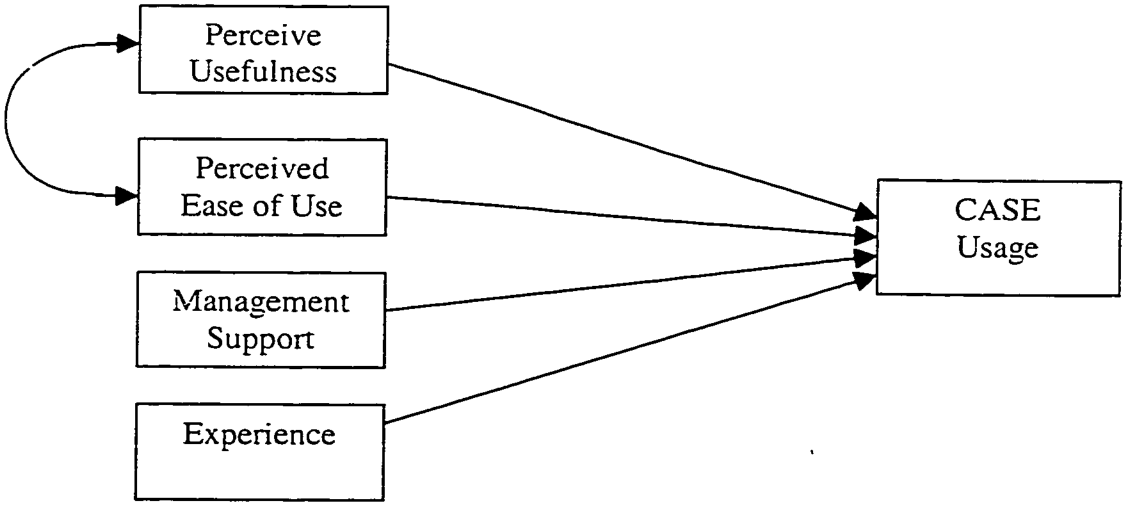


Figure 9: Full Path diagram

In order to determine the path coefficients, a multiple linear regression was conducted between the four independent variables and the dependent variable. The stepwise regression method was used in order to determine which of the independent variables contribute significantly to the dependent variable. A 0.01 significance level was used to determine which variables are included and which are omitted from the regression model. The output of the stepwise regression is depicted in tables 6 - 8.

Model	Variables Entered	Variables Removed	Method
1	USEFUL	.	Stepwise (Criteria: Probability-of-F-to-enter <= .010, Probability-of-F-to-remove >= .050).
2	MGTSPT	.	Stepwise (Criteria: Probability-of-F-to-enter <= .010, Probability-of-F-to-remove >= .050).
3	EXP	.	Stepwise (Criteria: Probability-of-F-to-enter <= .010, Probability-of-F-to-remove >= .050).

a Dependent Variable: USAGE

Table 6: Variables Entered/Removed

As evident in Table 6, three independent variables, Perceived Usefulness, Management Support, and Experience were chosen in the regression model where Usage is the dependent variable. The independent variable Perceived Ease of Use was omitted from the regression model.

Since the independent variable Perceived Ease of Use was hypothesised to have a positive direct relation with the dependent variable, further investigation into its omission was conducted. Another regression of the four independent variables was conducted using the Enter regression method. As can be seen in Table 7, the three independent variables, Perceived Usefulness, Management Support, and Experience all had an observed significance level (P-value) of less than 0.01. In the case of Perceived Ease of Use, the P-value was 0.013, which is higher than 0.01 significance level used. Hence, it was omitted from the regression model.

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig.
		B	Std. Error	Beta		
1	**(Constant)**	-6.392E-07	.048		.000	1.000
	USEFUL	.663	.085	.640	7.804	.000
	EASE	-.205	.081	-.194	-2.518	**.013**
	MGTSPT	.487	.058	.439	8.327	.000
	EXP	-.226	.062	-.202	-3.639	.000

a Dependent Variable: USAGE

Table 7: Enter regression method coefficients

Although the stepwise regression did not support a direct positive effect of Perceived Ease of Use on the dependent variable, the path coefficient between Perceived Ease of Use and Perceived Usefulness had to be investigated.

There are two ways to determine the significance of the path coefficient between Perceived Ease of Use and Perceived Usefulness. One is by simple correlation, where a significant correlation coefficient is the path coefficient, which in this case is significant at 0.01 (Table 8).

		USEFUL	EASE
USEFUL	Pearson Correlation	1	**.764**
	Sig. (2-tailed)	.	.000
	N	107	107
EASE	Pearson Correlation	**.764**	1
	Sig. (2-tailed)	.000	.
	N	107	107

** Correlation is significant at the 0.01 level (2-tailed).

Table 8: Simple correlation between the two independent variables

The other way is by simple regression, where Perceived Ease of Use is the independent variable and Perceived Usefulness is the dependent variable. In this method, the P-value of the regression is used to decide if there exists a significant path between the two variables.

As can be seen in Table 9, the P-value is less than 0.01 indicating that a significant path does in fact exist. This is no surprise since the two ways of determining the path should yield similar results.

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	**(Constant)**	-1.239E-06	.058		.000	1.000
	USEFUL	.750	.062	.764	12.139	**.000**

a Dependent Variable: EASE

Table 9: Simple Regression between the two independent variables

"The path coefficients are the regression coefficients for the standardised predictors" (Neter et al. 1985). The path coefficient between Perceived Ease of Use and Perceived Usefulness is the correlation coefficient or the simple regression coefficient since they are the same.

The sign of the path coefficient determines the nature of the relation between the two variables. A positive coefficient indicates that increase in the independent variable will result in an increase in the dependent variable. Similarly, a decrease in the former will result in a decrease in the latter variable. The regression coefficients in Table 10 indicate the following:

1. Perceived Usefulness has a positive effect on CASE Usage
2. Management Support has a positive effect on CASE Usage
3. Experience has a negative effect on CASE Usage

Tables 8 and 9 indicate a positive relation between Perceived Ease of use and Perceived Usefulness.

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-8.436E-07	.067		.000	1.000
	USEFUL	.727	.072	.702	10.108	.000
2	(Constant)	-3.891E-07	.052		.000	1.000
	USEFUL	.585	.058	.565	10.009	.000
	MGTSPT	.523	.063	.472	8.357	.000
3	(Constant)	-3.837E-07	.049		.000	1.000
	USEFUL	.506	.059	**.489**	8.538	.000
	MGTSPT	.488	.060	**.439**	8.129	.000
	EXP	-.233	.063	**-.208**	-3.665	.000

a Dependent Variable: USAGE

Table 10: Stepwise Regression coefficients

The value of each path coefficient determines the amount of change in the dependent variable that is caused by the change in the independent variable when all other variables are held constant. Table 10 indicates that increasing Perceived Usefulness by one unit will result in a 0.489 increase in Usage. Similarly, increasing Management Support by one unit will result in a 0.439 increase in Usage. On the other hand, due to the negative sign of the coefficient, increasing Experience by one unit will result in a 0.208 decrease in Usage.

The significant relation between Perceived Ease of Use and Perceived Usefulness, accompanied by the significant relation between the latter and Usage translates into an indirect effect of Perceived Ease of Use on Usage, where Perceived Usefulness is considered a mediation variable in this indirect path (Venkatraman, 1989). The fact that both coefficients are positive indicates that the total indirect effect is also positive.

To calculate the total indirect effect of Perceived Ease of Use on Usage, the two coefficients are multiplied (Venkatraman, 1989). The product of the multiplication yields a total effect of 0.374, meaning that increasing Perceived Ease of Use by one unit, holding all other factors constant, will result in a 0.374 increase in Usage.

The way the indirect path works is very easy to follow. Increasing Perceived Ease of Use by one unit will result in a 0.764 increase in Perceived Usefulness since the correlation between the two is 0.764. A 0.764 increase in Perceived Usefulness will result in 0.764

times the path coefficient between Perceived Usefulness and Usage, which equals 0.489 yielding a total effect of 0.374.

The final path diagram with the path coefficients is depicted in Figure 10.

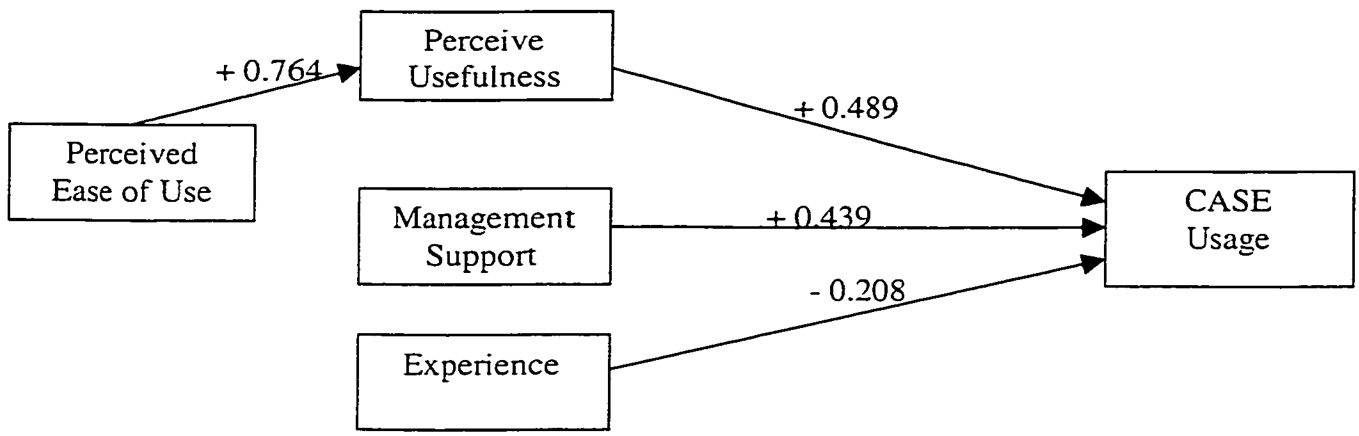


Figure 10: Final Path diagram

RESULTS

Analysis of findings

The proposed model had five hypothesised paths. Path analysis of the data supported four of the hypotheses. On the other hand, the fourth hypothesis was not supported by the data.

As can be seen in Table 11, 73.2% of the variance in CASE adoption behaviour by the IS professionals studied was explained by the three factors suggested to affect CASE adoption by those professionals.

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.702	.493	.488	.69545	.493	102.162	1	105	.000
2	.835	.697	.691	.54048	.204	69.845	1	104	.000
3	.855	**.732**	.724	.51082	.035	13.429	1	103	.000

a Predictors: (Constant), USEFUL
b Predictors: (Constant), USEFUL, MGTSPT
c Predictors: (Constant), USEFUL, MGTSPT, EXP

Table 7: Model Summary

The first path linked Perceived Usefulness of CASE by IS professionals to their level of CASE Usage through the following hypothesis:

H1: Perceived Usefulness has a direct positive effect on CASE Usage.

The coefficient of 0.489 obtained by the regression model (Table 10) supported this hypothesis at a 0.01 significance level. The direction of the relation was also supported since the sign of the coefficient was positive.

This finding fits very well among findings in previous studies. TAM strongly supported this positive path and indicated that the increase in users' perception of a certain technology as useful in their job will increase their level of usage of that particular technology.

Findings of this study also supported that the level of users' perceptions of CASE as useful in system analysis and development does in fact affect their level of adopting and using CASE. The findings strongly suggest that users' perception of CASE as useful is indeed an important factor that affects the technology usage.

The second hypothesised path linked perceived ease of use of CASE with its usage through the following hypothesis:

H2: Perceived Ease of Use has a direct positive effect on CASE usage.

The analysis results did not support the first aspect of a direct path between the users' perceptions of how easy CASE technology is and their level of CASE usage. This finding, although contradicts Davis's (1989) Technology Acceptance Model, is in mesh with other studies. Chau (1996) stated that "perceived ease of use had no statistically significant influence on use". His findings suggested that the easiness of CASE does not affect the level of its usage directly.

The fact that a direct effect was not supported maybe explained by the increase in CASE literacy among IS professionals. CASE technology has come a long way through the past years. On one hand, the technology has developed into a more user-friendly application. On the other hand, the IS community has become more knowledgeable about the technology. These two factors might have affected the way IS professionals build their decision about using or not using CASE. Since the easiness gap between CASE technology and its users seems to have been reduced, perceived ease of use had become a secondary concern.

The path between Management Support to CASE technology and CASE usage was tested through the following hypothesis:

H3: Management Support has a direct positive effect on CASE Usage.

The regression coefficient of 0.439 (Table 10) did indeed support a direct positive path between the level of management support to CASE technology and the level to which the technology is being used by the IS professionals.

It is of great importance to note that the study sample came from organisations within which CASE technology is available but its usage is on a voluntary basis. The findings

suggest that the level of management support to CASE, which could be seen as the primary adoption unit, does in fact affect influence the IS professionals' use of CASE, which is the secondary adoption unit.
If a system developer or analyst is given enough support by the management, in the sort of technical support or CASE seminars, most probably he or she will react positively towards using the technology.

This path could be explained from two angles. First, management support to CASE creates a work environment that encourages IS professionals to use and adopt the technology. This CASE-promoting environment could be in the form of on the job consulting, CASE seminars, or other ways of technical and professional support.

On the other hand, when an IS professionals notice that management supports CASE technology, as a team player, he or she would recognise that he/she can contribute better to the team by using a common language, that is CASE technology.

The fourth hypothesis suggested a direct, yet negative path between Non-CASE experience, or traditional systems development experience and CASE usage.

H4: Non-CASE Experience has a direct negative effect on CASE usage.

The obtained path coefficient of -0.298 (Table 10) supports this hypothesis at a 0.01 significance level. The negative sign of the coefficient clearly indicates that IS professionals that spent more time and collected more experience in traditional systems analysis and development have a lower level of CASE usage.

This negative relation between experience and CASE usage is a classic example of change resistance.

IS professionals, who invested a considerable amount of time and effort in earning and perfecting their experience in traditional systems development methods, tend to resist any change that they perceive as a threat to their hard-earned experience. Several previous studies have concluded that the considerable changes brought by CASE are always met with resistance from more experienced potential users who are trying to protect their points of strength (Chau, 1996, Iivari, 1996, Ramiller, 1993).

The IS professionals "often show as much resistance to change as their users did when IS brought them new systems" (Keen, 1991). This is one of CASE's unique effects where the change agents are themselves required to change.

The fifth hypothesise suggested a direct positive relation between perceived ease of use and perceived usefulness. Analysis of the data supported a significant positive relation between the two independent variables, which translates into an indirect positive[5] path between perceived ease of use and usage.

---

[5] Since the two paths making the indirect path are positive.

This indirect path between found between perceived ease of use and CASE usage through usefulness supports the idea that CASE's ease of use is a secondary concern.

IS professionals seem to judge CASE by how useful they think it is. The mediation of perceived usefulness between perceived ease of use and usage suggest that user perceptions of CASE as easy to use affects their perception of CASE as useful, which through a direct path affects usage.

This indirect positive effect of perceived ease of use on usage is very well supported by Chau (1996), where he concluded that "the individuals. assessment of the usefulness of the technology is influenced by the technology's ease of use". Even Davis (1989) suggested that perceived ease of use "may actually be a causal antecedent to perceived usefulness, as opposed to a parallel, direct determinant of usage".

Findings of this research suggest that perceived ease of use of CASE technology is a secondary factor that does eventually affect CASE usage.

Conclusion

The final model concluded in this study identified four factors that affect CASE usage by systems analysts and developers. Two of these factors, perceived usefulness of CASE and management support to CASE, were found to have a direct positive effect on the technology usage. On the other hand, the third factor, that is the users' level of non-CASE experience, had a negative direct effect on CASE usage.

However, the hypothesised path between perceived ease of use of CASE and the level of CASE usage, which suggested a direct positive relation, was not supported by the findings. Nevertheless, although a direct path was not significant, a positive indirect relation was concluded between perceived ease of use and usage, with perceived usefulness as a mediating factor.

Results of this study were in harmony with findings of previous studies. The strong direct path between perceived usefulness of a certain technology and its usage is widely supported (Dishaw and Strong, 1998, Chau, 1996, Davis, 1989). System experts' pursuit of their own professional development lead them to build a significant portion of their decision on their perception of how useful CASE is to their job. The more they perceive the technology as an addition to their set of skills the more they move toward using it.

The IS professionals' emphasis on how easy CASE technology is followed a different path. CASE users seemed to treat this aspect of the technology as a secondary issue. The indirect path through which perceived ease of use influences usage indicated that if a

CASE tool is perceived as easy to use, this is not sufficient enough to increase its usage. The role of perceived usefulness as a mediating factor suggests that perceptions of a CASE tool as easy to use will translate into higher perceptions of the usefulness of that tool, thus, leading to an increase in that tools' usage.

The effect of management support to CASE simply follows common sense. The more interest management shows in CASE technology the more receptive the users are. Management support to CASE helps create a CASE-friendly work environment that promotes the adoption of the technology by its potential users.

The negative relation between non-CASE experience and usage is a classic example of resistance to change. IS professionals with more traditional systems development experience view CASE as a threat. Hence, they react in a defensive way and avoid using the technology.

This low usage of CASE by analysts and developers with more traditional experience could be a result of their concerns that adopting the new technology will bring them back to square one, where their hard-earned non-CASE experience is of much less value.

Applying the findings of this study in organisations that are trying to promote CASE usage by its employees could be achieved by providing employees with more knowledge about CASE, its capabilities, and the advantages of using it. At the same time, providing them with professional CASE help through a CASE consulting firm for example, will help break the learning barriers of the technology.

More focus on young and relatively less experienced generation of IS professionals could yield better results since such generation is more receptive and welcoming to change and display less resistance due to their lower levels of experience that could be rendered inapplicable by CASE adoption.

Limitations and Recommendations

Findings of this study were based on data collected from 107 IS professionals. Such a small sample size imposes some limitations on the ability to generalise the findings to a larger population.

Although findings of this study were in mesh with previous studies, the fact that such previous studies were conducted in a different geographical context, like England or Finland for example, limits our ability to attribute conclusions reached in this study to the Canadian IS population. Similar studies are needed within Canada so that a better view is provided.

The voluntary CASE usage policy adopted by the organisations studied in this research also imposes limitations on applying the findings to organisations with different CASE usage policies for their employees.

The focus of this study, as stated in the beginning, was on CASE as a technology and not on a specific tool in particular. Since different CASE tools have different characteristics, findings of this study should be interpreted carefully when dealing with a specific CASE tool.

This research explored the effect of four factors suggested by the literature to have a significant effect on how IS professionals build their decision regarding CASE usage. It is very apparent, from both the literature review and the 0.732 R^2 obtained, that other factors exist. The fact that such other factors did not fit the scope of this study, and therefore were not included, does not mean that they don't exist. A deeper look at factors like the technical characteristics of CASE and the diffusion of the technology within the educational system would provide some insightful findings.

This research studied the effect of traditional (non-CASE) systems development experience on CASE adoption by IS professionals. However, due to scope limitation, the level of CASE experience possessed by the respondents was not taken into consideration. Further investigation on the combined effect of both CASE and traditional methods on IS professionals CASE adoption is required. Including the levels of CASE experience possessed by the respondents in the model will certainly provide a better understanding of IS professionals CASE adoption behaviour.

The methodology used in this study depended on IS professionals response to an advertisement in CIPS newsletter asking them to participate. As a consequence, the type of respondents studied was IS professionals that are somehow interested in CASE technology. Not necessarily using it, but at least interested enough to participate in a study about it.

The use of a more direct and personal approach, as in e-mails sent to a target population requesting their input is highly recommended for future research. On one hand, a more direct approach makes it more practical to follow up on late responses, thus, increasing the overall response rate. On the other hand, it could provide a study sample that is more normally distributed with regards to their interest in CASE technology.

REFERENCES

Aaen, I. Siltanen, A. Sorensen, C. and Tahvanainen, V. A tale of two countries. IFIP Transactions A-8, North Holland, Amsterdam, 1992.

Adams, D. Nelson, R. Todd, P. Perceived usefulness, ease of use, and usage of information technology: a replication, MIS Quarterly, Vol. 16, Iss. 2, 1992, p. 227-248.

Agarwal, R. The role of innovation characteristics and perceived voluntariness in the acceptance of information technology. Decision Sciences, Atlanta; Vol. 28, Iss. 3, Summer 1997, p. 557-583.

Ambler, S. Forever in search of the perfect CASE tool. Computing Canada, Vol. 23, Iss. 4, Feb. 17, 1997, p. 25-26.

Banker, R.D. and Kauffman, R. Reuse and productivity in integrated computer-aided software engineering. MIS Quarterly, Vol. 15, Iss. 3, Sep 1991, pg. 375-402.

Banker, R. Davis, G. Slaughter, S. Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study. Management Science, Providence; Vol. 44, Iss. 4 Apr 1998, p. 433-451.

Bartholomew, D. CASE closed? Andersen Consulting is quietly closing the door on users of its software; Informationweek, Manhasset; Oct 16, 1995, pg. 14-15.

Beck, L. Perkins, T. A Survey of Software Engineering Practice: Tools, Methods, and Results. IEEE Transactions on Software Engineering SE-9, No. 5, Sep. 1988, p. 541-561.

Becker, S. Gibson, R. An information abstraction model for end user CASE support. Journal of End User Computing; Harrisburg, Vol. 9, Iss. 1, Winter 1997, pg. 28-35.

Biggs, M. CASE tool make for quick methodical development. InfoWorld, Framingham; Vol. 18, Iss. 50: Dec 9, 1996, p. 116-117.

Borchardt, D. CASE tools as catalysts for reengineering: Partial tool support is better than none. Information Systems Management, Boston; Vol. 12, Iss. 4, Fall 1995, p. 20-26.

Bulc,V, A study of impacts of CASE tools on the working process of system developing crew. Elektrotechniski, Jan-Mar 1992.

Burkhard, D. Implementing CASE tools. Journal of System Management, Vol. 40, Iss. 5, 1989, p. 20-6-26.

Card, D. McGarry. F. Page. G. Evaluating software engineering technologies. IEEE Transactions on Software Engineering. SE-13, 7 Jul 1987.

Case, A. Computer-Aided Software engineering: Technology for Improving Software Development Productivity. Data Base 17, No. 1, Fall 1985, p. 24-27.

Chau, Patrick Y K. An empirical investigation on factors affecting the acceptance of CASE by systems developers. Information & Management, Amsterdam, Vol. 30, Iss. 6, Sep 1996, p. 269-281.

Chau, P. An empirical assessment of a modified technology acceptance model. Journal of management information systems. Armonk, Vol. 13, Iss. 2, Fall 1996, p. 185-205.

Chen, M. Nunamaker, J. Weber, E. Computer-aided software engineering: present status and future directions. DATA BASE, Spring 1989.

Chikofsky, J. Rubenstein, B. CASE: reliability engineering for information systems. IEEE software, Vol. 5, Iss. 2, 1989.

Coolidge, R. Making Software: the case for CASE. Inform. Vol. 2, Iss. 2, Feb 1988, p. 19-22.

Coupe, R. Onodu, N. Evaluating the impact of CASE: an empirical comparison of retrospective and cross-sectional survey approaches. European Journal of Information Systems, Vol. 6, Iss. 1, Mar 1997, p. 15-25.

Darrow. B. Borland prepares object exchange technology. Computer Reseller News, Mar 1992.

Davis, F.D, Perceived Usefulness, Ease of Use, and User Acceptance of Information Technology, MIS Quarterly, Vol. 13 Iss. 1, Mar 1989, p. 319-339.

Davis, F. Bagozzi, R. and Warshaw, P. A comparison of two theoretical models. Management Science. Providence, Vol. 35, Iss. 8, Aug 1989, p. 982-1004.

DeMarco, T. Lister, T. Peopleware. Dorest House, New York, 1987.

Dickey, S. A hot warehouse solution at Southwest frozen food. Midrange Systems, Spring House; Vol. 11, Iss. 6, May 1998, p.59-60.

Dishaw, M. Strong, D. Supporting software maintenance with software engineering tools: A Computed task-technology fit analysis. The Journal of Systems and Software, Vol. 44, Iss. 2, Dec 1998, p. 107-121.

Edelman. F. Managers. Computer systems and productivity. MIS Quarterly, September, 1981.

Emory, C. Cooper, D. Business Research Methods. 4th ed. Homewood, Illinois: Irwin Publishing Inc. 1991.

Eriksen. L. Stage, J. A qualitative empirical study of Case tool support to method learning. Information and Software Technology. Guildford, Vol. 40, Iss. 5,6, Jul 1998, p. 339-346.

Feuche, M. Implementing CASE? Learn from the users. MIS Week, Sep. 18, 1989.

Finlay, P. Mitchell, A. Perceptions of the benefits from the introduction of CASE: A. MIS Quarterly. Minneapolis, Vol. 18, Iss. 4, Dec 1994, p. 353-371.

Fishbein, M. Azjen, I. Belief, Attitude, Intention, and Behavior, Reading, MA: Addison-Wesley, 1975.

Flynn, D. Arce, E. A Case tool to support critical success factors analysis in IT planning and requirements determination. Information and Software Technology. Guildford, Vol. 39, Iss. 5, May 1997, p. 311-322.

Flynn, D. Vagner, J. Dal Vecchio, O. Is CASE technology improving quality and productivity in software development. Logistics Information Management, Vol. 8, Iss. 2, Feb 1995, p. 8-22.

Fort, G. Norman, R. A self-assessment by the software engineering community. Communications of the ACM. New York, Vol. 35, Iss. 4, Apr 1992, p. 28-47.

Fuller, M. and Swanson, E. Information centers as organisational innovation: exploring the correlates of implementation success. Journal of Management Information Systems, Armonk, Vol. 9, Iss. 1, Summer 1992, p. 47-68.

Gibson, M. Snyder. C. Rainer, M. CASE: clarifying common misconceptions. Journal of Systems Management, Vol. 40, Iss. 5, May 1989, p. 12-20.

Goodhue, D. Thompson, R. Task-Technology Fit and Individual Performance. MIS Quarterly, Vol. 19, Iss. 2, Jun 1995, p. 213-237.

Guinan, P. The effective use of automated application development tools. IBM Systems Journal, Armonk; Vol. 36, Iss. 1; 1997, p. 124-140.

Haley, K. Lyman, H. thaine, M. The realities of CASE. Journal of Information Systems Management. Vol. 7, Iss. 3, Summer 1990, p . 18-24.

Henderson. A. Glorfeld. K. Cronan. T. On the repeated test-retest reliability of the end-user computing satisfaction instrument: a comment. Decision Sciences. Vol. 25, Iss. 4, Jul/Aug 1994, p. 655-668.

Henderson, J. Cooprider, J. Dimensions of IS planning and design aids: A functional model of CASE technology. Information Systems, Vol. 1, Iss. 3, Mar 1990. p.36-45.

Huff, C. Elements of realistic CASE tool adoption budget. Communications of the ACM. New York, Vol. 35, Iss. 4, Apr 1992, p. 45-55.

Iivari, J. Factors affecting perceptions of CASE effectiveness. European Journal of Information Systems, Vol. 4, Iss. 3, Mar 1995, p. 143-159.

Iivari, J. Why are CASE tools not used, ACM. Vol. 39, Iss. 10, Oct 1996, p. 94-104.

Isoda, S. Yamamoto, S. Kuroki, H.Oka, A. Evaluation and introduction of the structured methodology and a CASE tool. The Journal of Systems and Software. New York, Vol. 28, Iss. 1, Jan 1995, p. 49-59.

Jankowsi, D. CASE tool selection. Journal of Systems Management. Cleveland, Vol. 46, Iss. 4, Jul/Aug 1995, p. 20-27.

Jarzabek,S. The case for user-centered CASE tools. Association for Computing Machinery. Communications of the ACM. New York, Vol. 41, Iss. 8, Aug. 1998, p. 93-100.

Jones, M. Arnett, K. CASE use is growing but in surprising ways. Datamation. Vol. 38, Iss. 10, May 1992, p. 108-110.

Keen, P. Shaping the Future: Business Design through Information Technology, Harvard Business School Press, Boston, MA., 1991.

Kemerer, C. How the learning curve affects CASE tool adoption. IEEE Software, March 1992.

Kim, C., Westin, S. Software maintainability: Perceptions of EDP professionals. MIS Quarterly. Minneapolis, Vol. 12, Iss. 2, Jun 1988, p. 167-186.

Kusters, R. and Wijers, G. On the practical use of CASE tools, CASE'93, IEEE Computer society press, 1993.

Lempp, P. Lauber, R. What productivity increases to expect from a CASE environment: Results of a user survey. In Productivity, Progress, Prospects, and Payoff, Proceedings of the 27th Annual Technical Symposium, Gaithersburg, Md. June 9, 1988.

Leonard-Barton, D. and Deschamps, I. Managerial influence in the implementation of new technology. Management Science. Providence, Vol. 34, Iss. 10; Oct 1988, p. 1252-1266.

Lewis, T. The Big Software Chill. Computer. New York, Vol. 29, Iss. 3, Mar 1996, p. 12-14.

Loh, M. Nelson, R. Reaping CASE Harvests. Datamation. Barrington, Vol. 35, Iss. 13. Jul 1989, p. 31-34.

Low, G. Jeffery, D. Software development productivity and back-end CASE tools. Information and Software Technology. Amsterdam, Vol. 33, Iss. 9, Nov 1991, p. 616-622.

Mair, P. The Strategic Procurement of CASE Tools. NCC Blackwell, Oxford, 1993.

Martin, J. Previewing the application development revolution. PC Week. New York, Vol. 6, Iss. 34, Aug 1989, p. 70.

Martin, M. The case against CASE. Journal of Systems Management. Cleveland, Vol. 46, Iss. 1, Jan/Feb 1995, p. 54-58.

McClure, C. CASE is Software Automation. Prentice Hall, New Jersey, 1989.

McManus, J. A methodology for assessing BPR CASE tools. Management Services. Enfield, Vol. 41, Iss. 6, Jun 1997, p. 16-17.

Melone, N. Bayer, J. The effect of communication channel, level of sponsorship and timing on implementation. Working paper no. 88-89-32, GSIA, Carnegie-Mellon University, 1990.

Moore, G. Benbasat, I. Development of an instrument to measure the perceptions of adopting an information technology innovation. Information Systems Research. Vol. 2, Iss. 3, Mar 1991.

Necco, C. Tsai, N. Holgeson, K. Current usage of CASE software. Journal of Systems Management. Cleveland, Vol. 40, Iss. 5, May 1989, p. 6-12.

Nelson, A. Rottman, J. Before and after CASE adoption. Information and Management. Amsterdam, Vol. 31, Iss. 4, Dec 1996, p. 193-203.

Neter, J. Kunter, M. Wasserman, W. Nachtsheim, C. Applied Linear Statistical Models. Fourth Edition. McGraw-Hill 1996.

Norman, R. and Nunamaker, J. CASE productivity perceptions of software engineering professionals. Communications of the ACM. New York, Vol. 32, Iss. 9, Sep 1989, pg. 1102-1130.

Oinas-Kukkonen, H. Towards greater flexibility in software design systems through hypermedia functionality. Information and Software Technology. Guildford, Vol. 39, Iss. 6, Jun 1997, p. 391-398.

Orlikowski, W. CASE tools as organisational change: investigation incremental and radical changes in systems development. MIS Quarterly. Minneapolis, Vol. 17, Iss. 3, Sep 1993, p. 309-341.

Pearson, J. Hagmann, C. Status report on quality assurance methods. Information Systems Management. Boston, Vol. 13, Iss. 1, Winter 1996, p. 52-59.

Philipson, G. Implementing CASE Technology. Computer Technology Research Corporation, South California, 1990.

Poo. Danny, C. Chung, M. CASE and software maintenance practices in Singapore. The Journal of Systems and Software. New York, Vol. 44, Iss. 2, Dec 1998, p. 97-106.

Post, G. Kegan, A. Keim. Robert, T. A Structural Equation Evaluation of CASE Tools Attributes. Journal of Management Information Systems. Armonk, Vol. 15, Iss. 4, Spring 1999, p. 215-235.

Post, G. Kegan, A. Keim. Robert, T. A comparative evaluation of CASE tools. The Journal of Systems and Software. New York, Vol. 44, Iss. 2, Dec 1998, p. 87-97.

Rai, A. Howard, G. Propagating CASE usage for software development: an empirical investigationof key organisational correlates. OMEGA: International Journal of Management Science. Oxford, Vol. 22, Iss. 2, Mar 1994, p. 133-148.

Rai, A. and Patnayakuni, R. A Structual Model for CASE Adoption Behavior. Journal of Management Information Systems. Vol 13, No.2, Fall 1996, p. 205-235.

Ramiller, N. Computer-aided software engineering: perceptions of technology, work, and management among IS personnel, Proceedings of the 1993 ACM SIGCPR Conference, St.Louis, MO., 1993.

Ramsey, B. CASE tools simplify complex embedding. Electronic Engineering Times. Manhasset, Iss. 919, Sep. 16, 1996, p. 73-75.

Raymond, L. Bergeron, F. Rivard, S. Determinants of business process reengineering success in small and large enterprises: An empirical study in the Canadian context. Journal of Small Business Management. Milwaukee, Vol. 36, Iss. 1, Jan 1998, p. 72-86.

Robillard, P. The role of knowledge in software development. Association for Computing Machinery. Communications of the ACM. New York, Vol. 42, Iss. 1, Jan 1999, p. 87-93.

Rogers, E. Diffusion of innovations, Fourth Edition. The Free Press, New York, 1995.

Rothstein, M. Rosner, B. Senatore, M. Mulligan, D. Structured Analysis and Design for the CASE User. McGraw-Hill, New York, 1993.

Russell, F. The case for CASE. Software Engineering: A European Perspective. Eds. IEEE Computer Society Press. Los Alamitos, California, 1993.

Schneindwind, N. The state of software maintenance. IEEE Trans. Software Eng. SE-13, Vol. 3, 1987.

Scott, A. Forever in search of the perfect CASE tool. Computing Canada. Willowdale, Vol. 23, Iss. 4, Feb 1997, p. 25-26.

Scott, M. What's a CASE tool? Black Enterprise. New York, Vol. 25, Iss. 12, Jul 1995, p. 52-53.

Segars, A. Grover, V. Re-examining perceived ease of use and usefulness: a confirmatory factor analysis. MIS Quarterly. Vol. 17, Iss. 4, Dec 1993, p. 517-226.

Seiter, C. Making a case for CASE tools. MACWORLD. San Francisco, Vol. 14, Iss. 9, Sep 1997, p. 84-85.

Selamat, M. Rahim, M. Evaluation of computer-aided software engineering tools: Experience from Malaysian organisations. International Journal of Information Management. Guildford, Vol. 16, Iss. 4, Aug 1996, p. 299-314.

Senn, J. The other side of case implementation: Best practices for success. Information Systems Management. Boston. Vol. 12, Iss. 4, Fall 1995, p. 7-15.

Sharma, S. Rai, A. CASE Deployment in IS Organisations. Communications of the ACM. Vol. 34, No. 1, Jan 2000, p. 80-89.

Simons, L. Making CASE pay means more than a purchase. Computing Canada. Willowdale, Vol. 18, Iss. 13, Jun 1992, p. 31-32.

Smith, T. CASE technology and modeling make a comeback. Computer Reseller News. Manhasset, Iss. 633, Jun 1995, p. 144-145.

Solomon, H. Modelling tools help managers. Computer Dealer News. Willowdal, Vol. 15, Iss. 24, Jun 1999, p. 18.

Srinarayan, S. Case deployment in IS organisations. Association for Computing Machinery. Communications of the ACM. New York, Vol. 43, Iss. 1, Jan 2000, p. 80-89.

Stobart, S. Reeken, A. Low, G. An empirical evaluation of the use of CASE tools. Proceedings of the 16th International Workshop on Computer Aided Software Engineering, Singapore, July 1993.

Stobart, S. Thompson, J. Smith, P. software engineering tools for systems development. Computer-aided engineering Journal. Cleveland, Vol. 10, Iss. 6, Jun 1991, p. 72-80.

Subramanian. G. An examination of some software development effort and productivity determinants in ICASE tool projects. Journal of Management Information Systems. Armonk, Vol. 12, Iss. 4, Spring 1996, p. 143-161.

Swanson, E., Beath, C. Maintaining Information Systems in Organisations. Wiley, New York, 1989.

Swanson, J. McComb, D. Smith, J. McCubbrey, D. The application software factory: applying total quality technique to systems development. MIS Quarterly. Minneapolis, Vol. 15, Iss. 4, Dec 1991, p. 566-580.

Szajna, B. Empirical evaluation of the revised technology acceptance model. Management Science. Providence,Vol. 42, No 1, Jan 1996, p. 85-93.

Tabachnick, B. Fidell, L. Using Multivariate Statistics. 2nd Edition. New York: Harper and Row Publishers. 1989.

Tornatzky, L. Klein, K. Innovation characteristics and innovation adoption implementation: a meta analysis of findings. IEEE Transactions on Engineering Management. Vol. 29, Iss. 1, Jan 1982.

Urwiler, R. Ramarapu, N. Wilkes, R.Frolick, M. Computer-aided software engineering: The determinants of an effective implementation strategy. Information & Management. Amsterdam, Vol. 29, Iss. 4, Oct 1995, P. 215-226.

Van de Ven, A. H. Central problems in the management of innovation. Management Science. Providence, Vol. 32, Iss. 5, May 1986, p. 590-608.

Venkatraman, N. The Concept of Fit in Strategy Research: Toward Verbal and Statistical Correspondence. Academy of Management Review, Mississippi State, Vol. 14, Iss. 3, Jul 1989, p. 423-445.

Vessey, S. Jarvenpaa, and Tractinski, CASE tools as methodology companions. Communications of the ACM, Vol. 35, Apr 1992, p. 90-109.

Vessey, I. Sravanapudi, A. CASE tools as collaborative support technologies. Communications of the ACM. New York, Vol. 38, Iss. 1, Jan 1995, p. 83-96.

White, M. and Nelson, D. Organisational and individual differences in learning perspectives during the technological innovation process. Walter de Gruyter, Berlin, New York. January 1990.

Whitten, J. Bentley, L. Barlow, V. System Analysis and Design Methods, 3rd edn., Burr Ridge, Illinois. Irwin, 1994.

Windsor, J. Are automated tools changing systems analysis and design? Journal of Systems Management. Cleveland, Vol. 37, Iss. 11, Nov 1986, p. 28-33.

Wynekoop, J. An innovation study of implementation of computer-aided software engineering tools. Unpublished doctoral dissertation. Georgia State University, 1991.

Yang, H. Adoption and implementation of CASE tools in Taiwan. Information & Management. Amsterdam, Vol. 69, Iss. 1, Feb 1999, p. 89-113.

Yellen, R. System analysts performance using CASE versus manual methods. Proceedings of the Annual Hawaii International Conference on System Sciences, 1990.

Young, T. The lonely Micro. Datamation. Vol. 30, Iss. 4, Apr 1984, p. 100-108.

Zmud, R. An Examination of "Push-Pull" theory applied to process innovation and knowledge work. Management Science. Vol. 30, Iss. 6, Jun 1984, p. 727-739.

Appendix 1

Davis 1989 original questionnaire

Perceived Usefulness

Using CHART-MASTER in my job would enable me to accomplish tasks more quickly.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Using CHART-MASTER would improve my job performance.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Using CHART-MASTER in my job would increase my productivity.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Using CHART-MASTER would enhance my effectiveness on the job.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Using CHART-MASTER would make it easier to do my job.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

I would find CHART-MASTER useful in my job.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Perceived Ease of Use

Learning to operate CHART-MASTER would be easy for me.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

I would find it easy to get CHART-MASTER to do what I want it to do.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

My interaction with CHART-MASTER would be clear and understandable.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

I would find CHART-MASTER to be flexible to interact with.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

It would be easy for me to become skillful at using CHART-MASTER.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

I would find CHART-MASTER easy to use.

likely |_______|_______|_______|_______|_______|_______|_______| unlikely

extremely quite slightly neither slightly quite extremely

Appendix 2a

Confidentiality statement

Appendix 2b

Data Collection Instrument

The information you provide will be kept strictly confidential.

Thank you for your cooperation.

1. Using CASE in my job enables me to accomplish tasks more quickly

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

2. Learning to operate CASE is easy for me.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

3.Management provides me with the training I need in order to use CASE effectively.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

4. Years of experience in system development

Less than two	2-4 years	5-7 years	8-10 years	More than 10
○	○	○	○	○

5. Number of system development projects worked on

None	1-2 projects	3-4 projects	5-6 projects	More than 6
○	○	○	○	○

6.I feel I have a solid "Network of support" in the form of knowledgeable colleagues, internal support personnel, and/or outside consultants, to help me better understand and use CASE.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

7. I find it easy to get CASE to do what I want it to do.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

8. Using CASE improves my job performance.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

9. Using CASE increases my productivity

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

10. My interaction with CASE is clear and understandable

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree
○	○	○	○	○	○	○

11. Using CASE in my organization is:

11. Using CASE in my organization is:

Voluntary	Obligatory

12. Years of experience in **non-CASE** systems development.

Less than two	2-4 years	5-7 years	8-10 years	More than 10

13. Number of **non-CASE** systems development projects worked on.

None	1-2 projects	3-4 projects	5-6 projects	More than 6

14. Management supports CASE usage.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

15. I find CASE to be flexible to interact with.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

16. Using CASE enhances my effectiveness on the job.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

17. If level 1 represented the entry position at your department and level 5 represented the Head of the MIS department, at what level would you be.

Level 1	Level 2	Level 3	Level 4	Level 5

18. Using CASE makes it easier to do my job.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

19. It is easy for me to become skillful at using CASE.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

20. I find CASE easy to use.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

21 I find CASE useful in my job.

Strongly Agree	Quite Agree	Slightly Agree	Neither	Slightly Disagree	Quite Disagree	Strongly Disagree

22. Of all the functions provided by CASE, I use:

All of them	Most of them	Some of them	Few of them	None of them

22. Of all the functions provided by CASE, I use:

All of them	Most of them	Some of them	Few of them	None of them
○	○	○	○	○

23. To perform a task where either CASE or traditional methods could be utilized, I use:

CASE Always	CASE Most of the time	CASE Slightly more	Either	Traditional methods Slightly more	Traditional methods Most of the time	Traditional methods Always
○	○	○	○	○	○	○

The following questions are for statistical analysis purposes.
They will be kept extremely confidential as will all other information you provided so far.

24 What type of industry does your organization belong to?

Manufacturing	Financial services	Information Technology	Health care	Merchandizing	Public sector	Other
○	○	○	○	○	○	○

25. How many employees are there in your organization?

Less than 50	51-100	101-500	501-1000	1001-2000	2001-4000	More than 4000
○	○	○	○	○	○	○

26. Please indicate the highest education level you have achieved.

High school or less	CEGEP or equivalent	University certificate	Bachelors degree	Masters degree	Ph.D. degree
○	○	○	○	○	○

27. Age:

Less than 20	20-25	26-30	31-35	36-40	41-45	More than 45
○	○	○	○	○	○	○

Submit Questionaire

Appendix 3

Methodology verification

Date: Tue, 21 Nov 2000 16:32:16 -0500
Subject: Re: Electronic data collection
To: elkhairy@yahoo.com
From: "Linda Dyer" <ldyer@mercato.concordia.ca> |

Hi Jawad,

It was good to see you yesterday, and I'm happy that you are making progress on your thesis research.

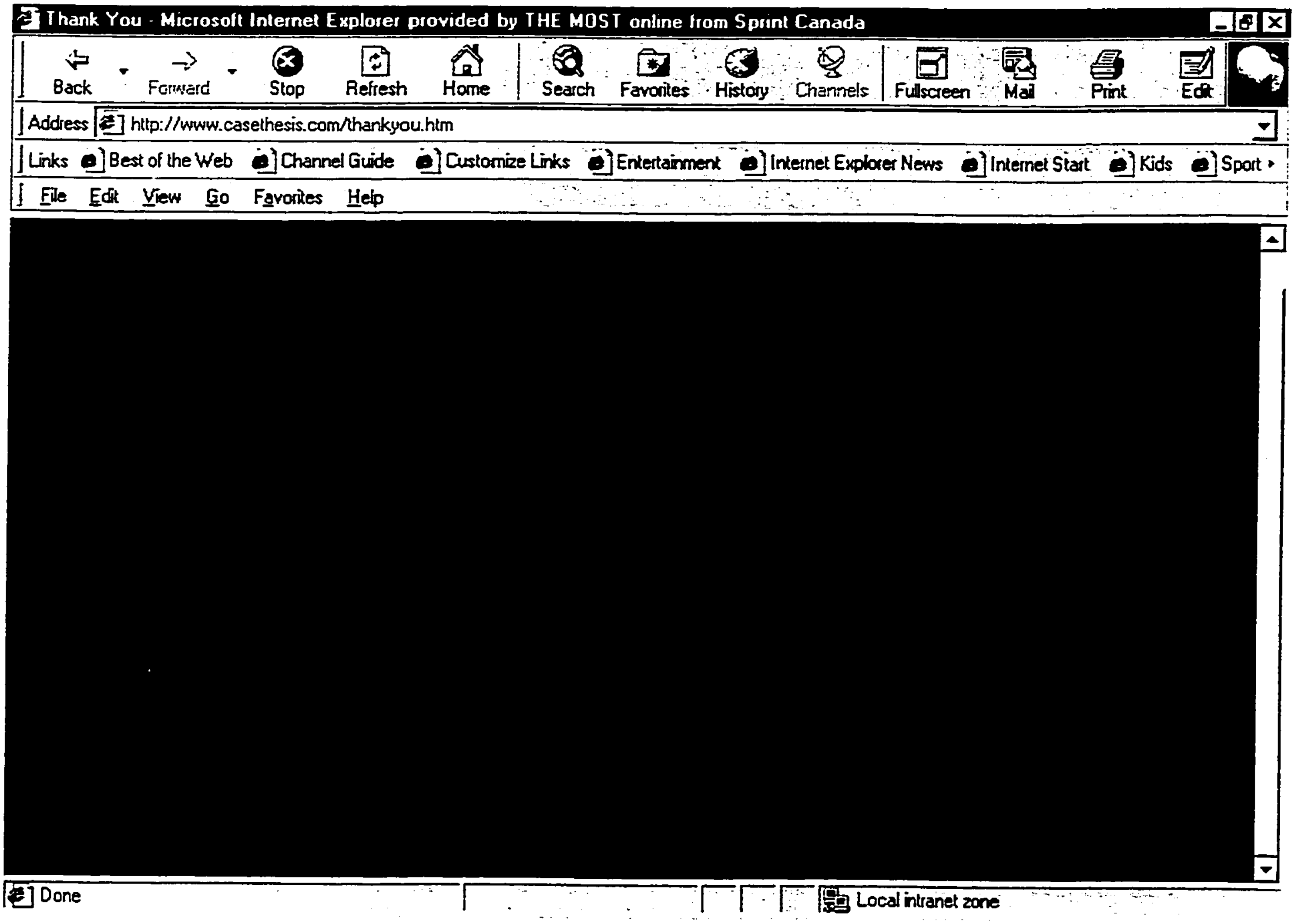
The only caveat regarding electronic data collection is that people need
to be assured that their responses really are anonymous--or if responses
are NOT anonymous, people need to be assured that they will be treated confidentially. I think a major advantage is that it is inexpensive to collect data this way, and people are likely to respond more quickly than
they would to a mail survey. I'm not aware of any general bias in this method (assuming, of course, that your entire population is computer literate).

Good luck,

Linda Dyer

Appendix 4

Confirmation Message



Appendix 5

Telephone Script

Good afternoon,

My name is Jawad Al-Khairy, I am a graduate student at Concordia University in Montreal. I'm conducting a research regarding CASE tools Acceptance by System Developers. The research involves contacting System Developers in order to provide them with a questionnaire. I'm calling to ask if you can provide me with a list of subscribers' e-mail addresses for that matter. Any information you provide will be dealt with in a professional way. Participation is voluntary. Any information they provide will be kept strictly confidential and for this research purposes only. Once the data are collected, the list will be destroyed.

If you consider this a viable idea, I will proceed by sending you a formal letter from Concordia University signed by my self and my supervisor, Prof. Dennis Kira, an Associate Professor in the Msc. In Administration department at the University.

Thank you and have a good day.

Appendix 6

E-mail to CIPS

Dear Alex

As per our phone conversation on Friday 17, I would like to thank you for your cooperation regarding including my questionnaire in your Newsletter. I really appreciate your help in this matter.

I want to reassure you that any information the subscribers provide will be dealt with in a professional manner and with extreme confidentiality. In fact, the method of data collection used grantees that respondents as well as their organizations will remain anonymous as mentioned on the cover letter.

The cover letter of the questionnaire will also indicate that the results of the study will be made available for those respondents that are interested. Findings of the study will also be made available for you and the Newsletter should you so desire.

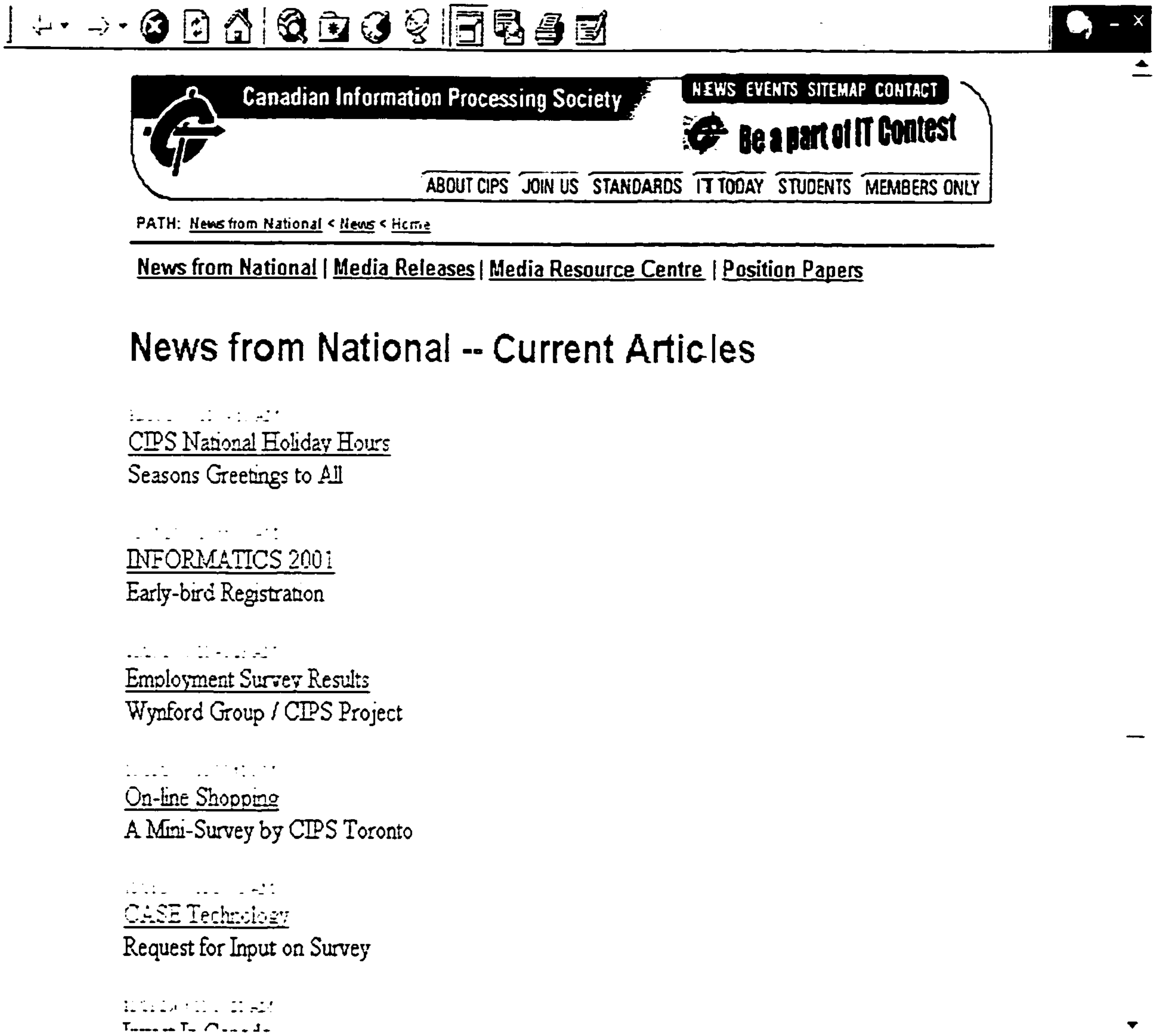
Once again I would like to thank you for your support. A statement thanking CIPS for its cooperation will be included in the early pages of my written thesis.

Sincerely,

Jawad Al-Khairy, M.Sc. student
Department of Decision Sciences and Management Information
Concordia University
Montreal, Quebec.

Appendix 7a

CIPS web-site


Canadian Information Processing Society

NEWS EVENTS SITEMAP CONTACT

Be a part of IT Contest

ABOUT CIPS | JOIN US | STANDARDS | IT TODAY | STUDENTS | MEMBERS ONLY

PATH: News from National < News < Home

News from National | Media Releases | Media Resource Centre | Position Papers

News from National -- Current Articles

CIPS National Holiday Hours
Seasons Greetings to All

INFORMATICS 2001
Early-bird Registration

Employment Survey Results
Wynford Group / CIPS Project

On-line Shopping
A Mini-Survey by CIPS Toronto

CASE Technology
Request for Input on Survey

Appendix 7b

Research outline on CIPS web-site

ABOUT CIPS | JOIN US | STANDARDS | IT TODAY | STUDENTS | MEMBERS ONLY

PATH: News from National < News < Home

News from National | Media Releases | Media Resource Centre | Position Papers

News from National -- Current Articles

12/21/2000 11:28:02 AM

CASE Technology

Request for Input on Survey

CASE (Computer-Aided Software Engineering) technology received contradicting evaluations throughout the past three decades. While some empirical studies applauded the technology on the improvements it introduced to IT, others regarded the technology as a total failure. Yet, generally speaking, most CASE research indicated that the final word should be that of the IS professionals that the technology is meant to serve.

A Masters student at Concordia University in Montreal, Jawad Al-Khairy, is conducting a research aimed at identifying the factors that affect CASE usage by system developers. The data collected from IS professionals on their attitude regarding CASE technology will be analyzed and incorporated in building a theoretical model that explains the relationships between CASE usage and the factors that affect it.

As an IS professional, you can participate in this research by answering the questionnaire attached to this link http://www.casethesis.com. Any information that you provide will be dealt with in a professional manner and with extreme confidentiality. Should you so desire, findings of the study will be made available to you.

News from National | Archives | Search

Appendix 8

Construct Measurement variables

Construct	Questionnaire item	Source	Total # of variables
Perceived Usefulness	1, 8, 9, 16, 18, 21	Davis, 1989	6
Perceived Ease of Use	2, 7, 10, 15, 19, 20	Davis, 1989	6
Management Support	3, 6, 14	Chau, 1996	3
Experience	12, 13 17	Iivari, 1995 Orlikowski, 1993	3
Usage	22, 23	Davis, 1989 Kim and Lee, 1986	2

Appendix 9

Questionnaire coding directory

Construct	Item #	Item code	Value						
			Agree			Neither	Disagree		
			Strongly	Quite	Slightly	Neither	Slightly	Quite	Strongly
Perceived Usefulness	1	Useful1	1	2	3	4	5	6	7
	8	Useful2	1	2	3	4	5	6	7
	9	Useful3	1	2	3	4	5	6	7
	16	Useful4	1	2	3	4	5	6	7
	18	Useful5	1	2	3	4	5	6	7
	21	Useful6	1	2	3	4	5	6	7
Ease of Use	2	Ease1	1	2	3	4	5	6	7
	7	Ease2	1	2	3	4	5	6	7
	10	Ease3	1	2	3	4	5	6	7
	15	Ease4	1	2	3	4	5	6	7
	19	Ease5	1	2	3	4	5	6	7
	20	Ease6	1	2	3	4	5	6	7
Management Support	3	Mgtspt1	1	2	3	4	5	6	7
	6	Mgtspt2	1	2	3	4	5	6	7
	14	Mgtspt3	1	2	3	4	5	6	7
			# of Years						
Experience	12	Exp1	**< 2**	**2-4**	**5-7**	**8-10**	**> 10**		
			5	4	3	2	1		
			# of Projects						
	13	Exp2	**0**	**1-2**	**3-4**	**5-6**	**> 6**		
			5	4	3	2	1		
			Job Level						
	17	Exp3	**Entry**	**2**	**3**	**4**	**Director**		
			5	4	3	2	1		
			CASE features						
Usage	22	Usag1	**All**	**Most**	**Some**	**Few**	**Non**		
			1	2.5	4	5.5	7		

Construct	Item #	Item code	Value						
			CASE				Traditional		
			Always	Most of the time	Slightly more	Either	Slightly more	Most of the time	Always
Usage	23	Usag2	1	2	3	4	5	6	7

Sample Characteristics

Field	Item #	Item code	Value						
Type of Industry	24	Indsry	Manufacturing	Financial Services	Information Technology	Health Care	Merchandising	Public Sector	Other
			1	2	3	4	5	6	7
# of Employees	25	Eplyes	< 50	51-100	101-500	501-1000	1001-2000	2001-4000	> 4000
			1	2	3	4	5	6	7
Education Level	26	Edction	High School or less	CEGEP or equivalent	University certificate	Bachelors degree	Masters degree	Ph.D degree	
			1	2	3	4	5	6	
Age	27	Age	< 20	20 - 25	26 - 30	31 - 35	36 - 40	41 - 45	> 45
			1	2	3	4	5	6	7

Appendix 10

Correlation matrices

Factor: Perceived Usefulness

R E L I A B I L I T Y   A N A L Y S I S
S C A L E   (A L P H A)

Correlation Matrix

	USEFUL1	USEFUL2	USEFUL3	USEFUL4	USEFUL5
USEFUL1	1.0000				
USEFUL2	.9044	1.0000			
USEFUL3	.8826	.8997	1.0000		
USEFUL4	.8474	.8703	.8669	1.0000	
USEFUL5	.7643	.7958	.8459	.8133	1.0000
USEFUL6	.8641	.8733	.9069	.8709	.8747

	USEFUL6
USEFUL6	1.0000

N of Cases = 107.0

Reliability Coefficients 6 items

Alpha = .9717 Standardised item alpha = .9733

Factor: Perceived Ease of Use

R E L I A B I L I T Y   A N A L Y S I S
S C A L E   (A L P H A)

Correlation Matrix

	EASE1	EASE2	EASE3	EASE4	EASE5
EASE1	1.0000				
EASE2	.7830	1.0000			
EASE3	.8559	.8250	1.0000		
EASE4	.7787	.9237	.8261	1.0000	
EASE5	.8820	.7242	.8430	.7364	1.0000
EASE6	.8043	.8666	.8145	.8626	.7884

	EASE6
EASE6	1.0000

N of Cases = 107.0

Reliability Coefficients 6 items

Alpha = .9648 Standardised item alpha = .9649

Factor: Management Support

R E L I A B I L I T Y   A N A L Y S I S
S C A L E   (A L P H A)

Correlation Matrix

	MGTSPT1	MGTSPT2	MGTSPT3
MGTSPT1	1.0000		
MGTSPT2	.6833	1.0000	
MGTSPT3	.6127	.6609	1.0000

N of Cases = 107.0

Reliability Coefficients 3 items

Alpha = .8477 Standardised item alpha = .8491

Factor: Experience

R E L I A B I L I T Y   A N A L Y S I S
S C A L E   (A L P H A)

Correlation Matrix

	EXP1	EXP2	EXP3
EXP1	1.0000		
EXP2	.6470	1.0000	
EXP3	.6216	.6294	1.0000

N of Cases = 107.0

Reliability Coefficients 3 items

Alpha = .8159 Standardised item alpha = .8378

Factor: CASE Usage

R E L I A B I L I T Y   A N A L Y S I S
S C A L E   (A L P H A)

Correlation Matrix

	USAG1	USAG2
USAG1	1.0000	
USAG2	.8905	1.0000

N of Cases = 107.0

Reliability Coefficients 2 items

Alpha = .8543 Standardised item alpha = .9421

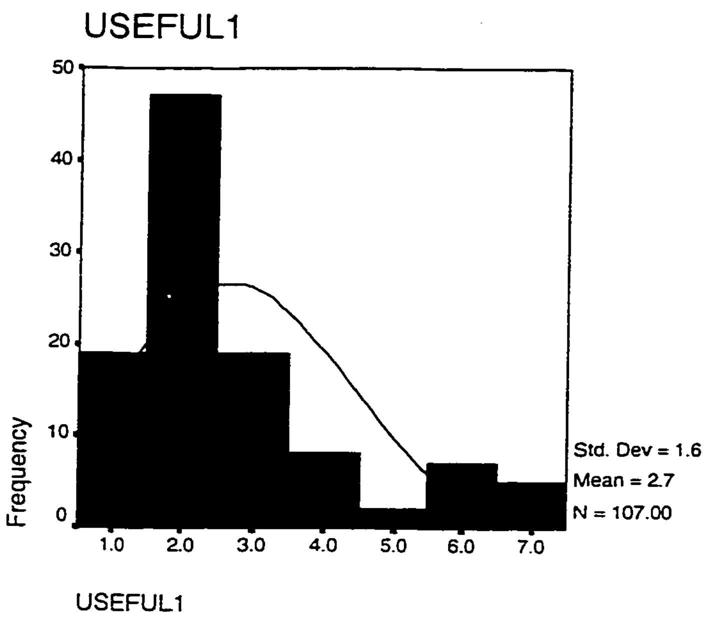
Appendix 11

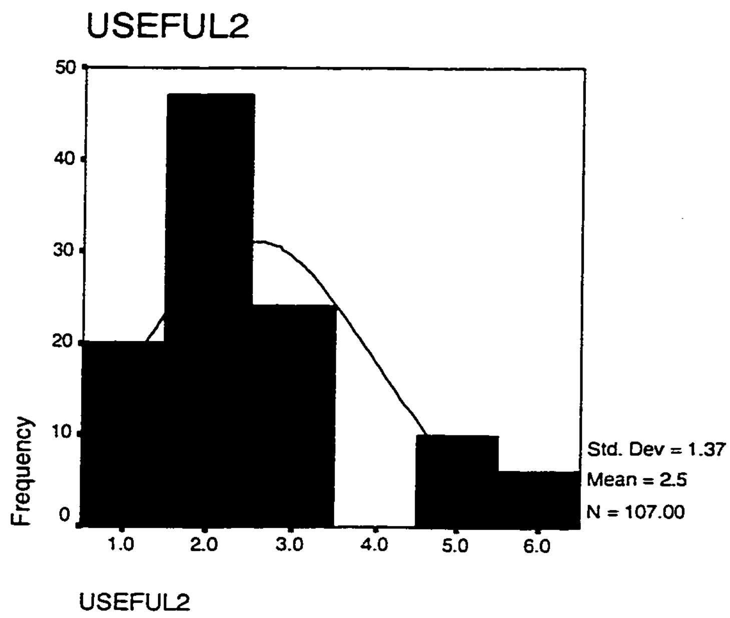
Correlation Matrix

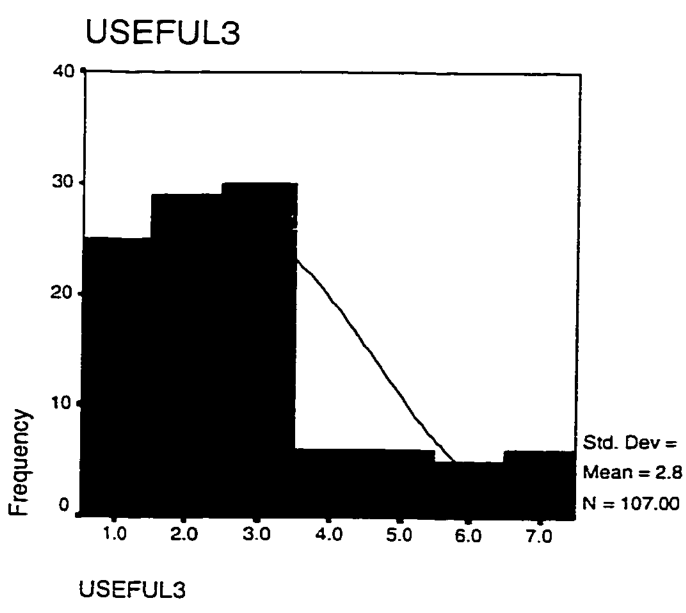
Item	USEFUL1	USEFUL2	USEFUL3	USEFUL4	USEFUL5	USEFUL6	EASE1	EASE2	EASE3	EASE4	EASE5	EASE6	MGTST1	MGTST2	MGTST3	EXP1	EXP2	EXP3	USAG1	USAG2
USEFUL1	1	.904	.883	.847	.764	.864	.574	.625	.658	.655	.673	.660	.241	.264	.337	-.335	-.281	-.499	.695	.771
USEFUL2	.904	1	.900	.870	.796	.873	.582	.666	.691	.660	.691	.656	.209	.306	.337	-.342	-.310	-.463	.656	.722
USEFUL3	.883	.900	1	.867	.846	.907	.561	.678	.707	.688	.638	.661	.234	.354	.332	-.336	-.267	-.395	.687	.724
USEFUL4	.847	.870	.867	1	.813	.871	.549	.688	.660	.716	.640	.614	.100	.155	.232	-.307	-.224	-.392	.565	.564
USEFUL5	.764	.796	.846	.813	1	.875	.646	.695	.701	.709	.731	.712	.194	.180	.173	-.236	-.334	-.353	.498	.551
USEFUL6	.864	.873	.907	.871	.875	1	.612	.648	.694	.684	.743	.670	.168	.247	.253	-.271	-.316	-.379	.589	.672
EASE1	.574	.582	.561	.549	.646	.612	1	.783	.856	.779	.882	.804	.376	.020	.059	-.258	-.179	-.267	.281	.415
EASE2	.625	.666	.678	.688	.695	.648	.783	1	.825	.924	.724	.867	.244	.080	.047	-.260	-.179	-.199	.362	.335
EASE3	.658	.691	.707	.660	.701	.694	.856	.825	1	.826	.843	.815	.486	.247	.207	-.266	-.196	-.296	.449	.509
EASE4	.655	.660	.688	.716	.709	.684	.779	.924	.826	1	.736	.863	.249	.025	.096	-.242	-.155	-.252	.402	.383
EASE5	.673	.691	.638	.640	.731	.743	.882	.724	.843	.736	1	.788	.396	.099	.157	-.229	-.188	-.269	.379	.515
EASE6	.660	.656	.661	.614	.712	.670	.804	.867	.815	.863	.788	1	.252	.024	.069	-.226	-.243	-.235	.361	.421
MGTSPT1	.241	.209	.234	.100	.194	.168	.376	.244	.486	.249	.396	.252	1	.683	.613	-.248	-.089	-.303	.461	.437
MGTSPT2	.264	.306	.354	.155	.180	.247	.020	.080	.247	.025	.099	.024	.683	1	.661	-.135	-.116	-.213	.552	.527
MGTSPT3	.337	.337	.332	.232	.173	.253	.059	.047	.207	.096	.157	.069	.613	.661	1	-.261	-.137	-.290	.664	.613
EXP1	-.335	-.342	-.336	-.307	-.236	-.271	-.258	-.260	-.266	-.242	-.229	-.226	-.248	-.135	-.261	1	.647	.622	-.488	-.341
EXP2	-.281	-.310	-.267	-.224	-.334	-.316	-.179	-.179	-.196	-.155	-.188	-.243	-.089	-.116	-.137	.647	1	.629	-.376	-.250
EXP3	-.499	-.463	-.395	-.392	-.353	-.379	-.267	-.199	-.296	-.252	-.269	-.235	-.303	-.213	-.290	.622	.629	1	-.658	-.543
USAG1	.695	.656	.687	.565	.498	.589	.281	.362	.449	.402	.379	.361	.461	.552	.664	-.488	-.376	-.658	1	.890
USAG2	.771	.722	.724	.564	.551	.672	.415	.335	.509	.383	.515	.421	.437	.527	.613	-.341	-.250	-.543	.890	1

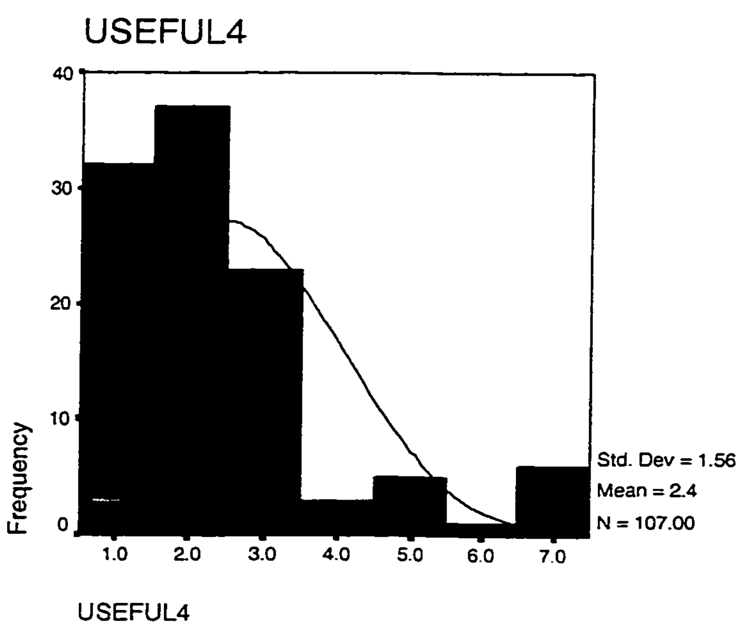
Appendix 12

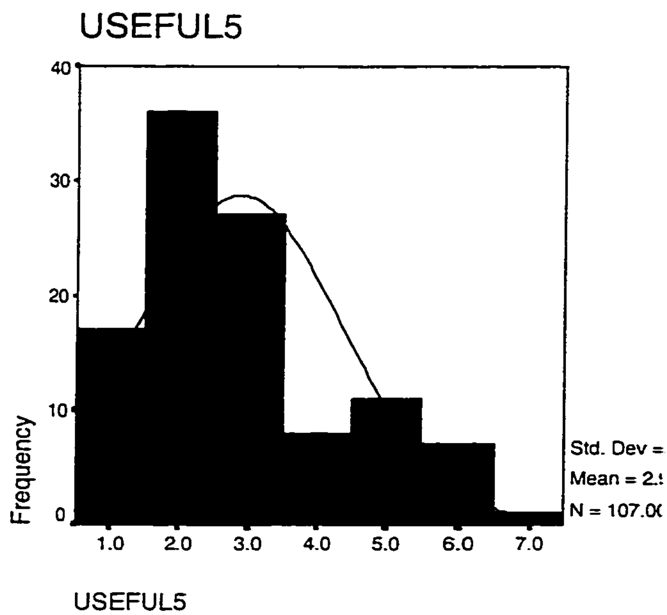
Variable distribution histograms and calculations

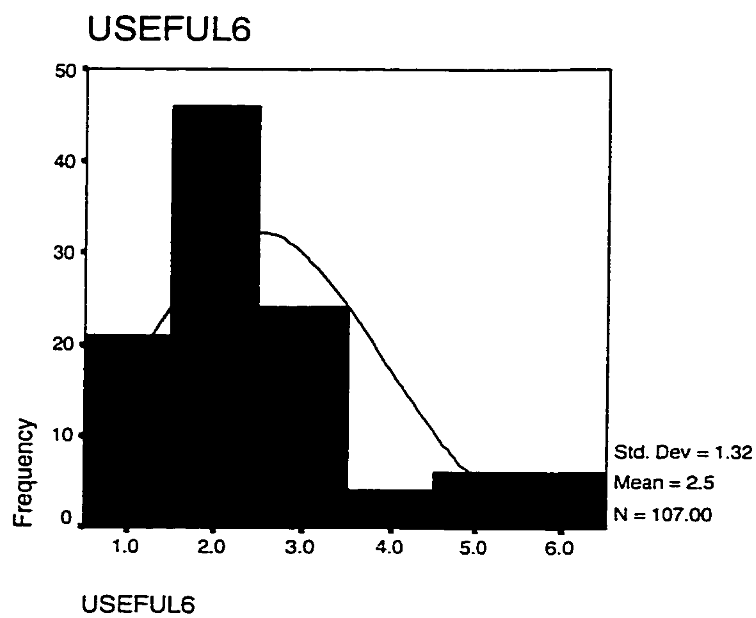






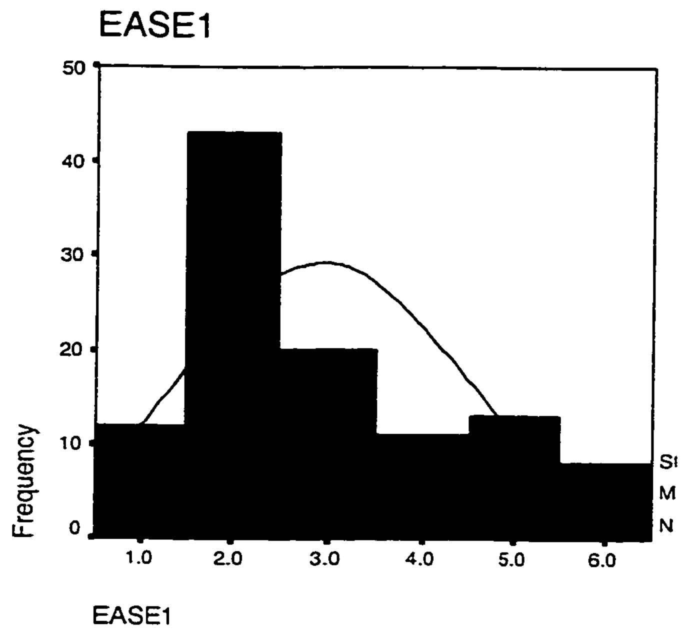


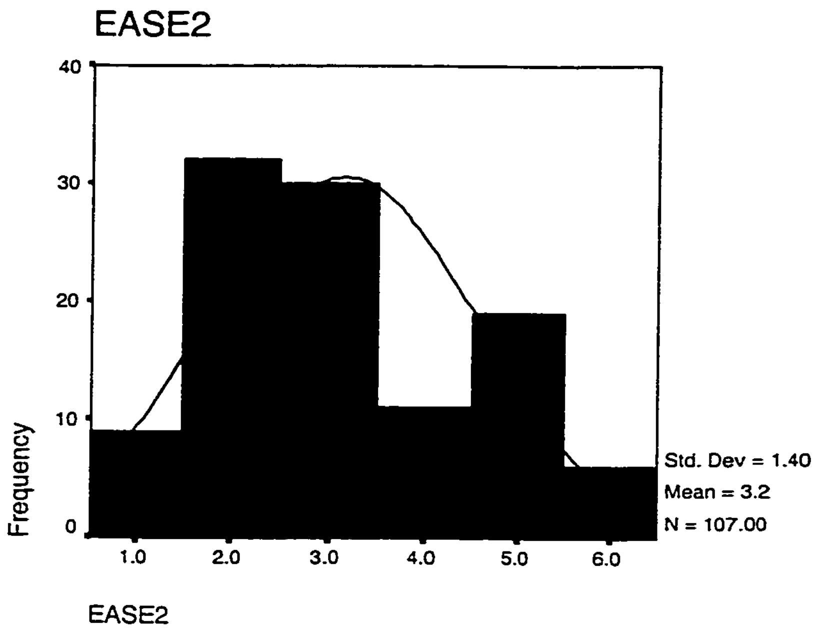


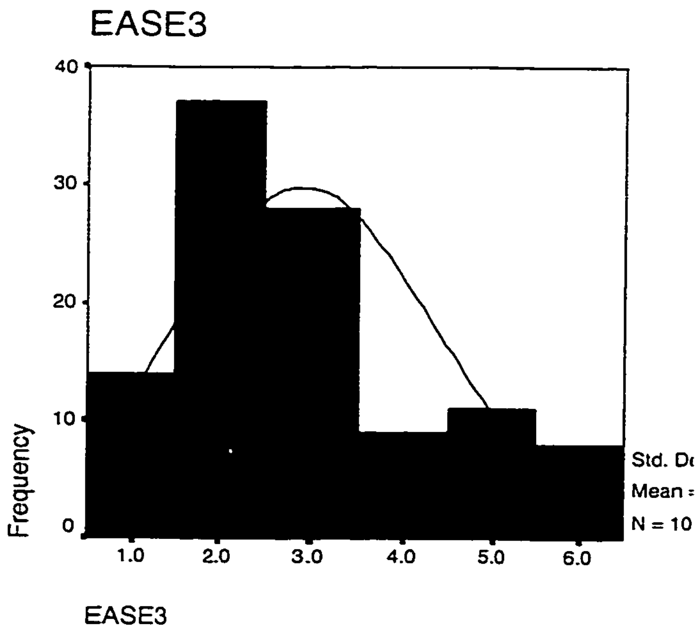


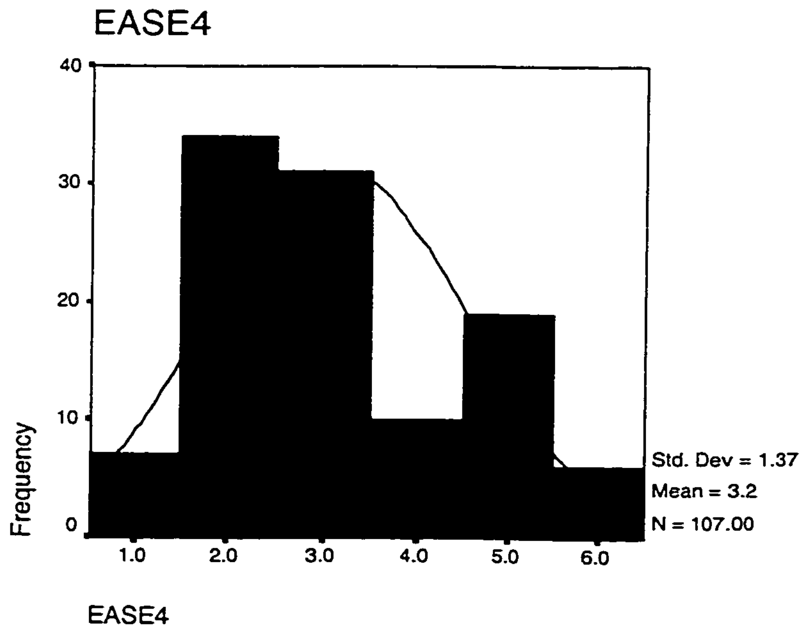
Descriptive Statistics

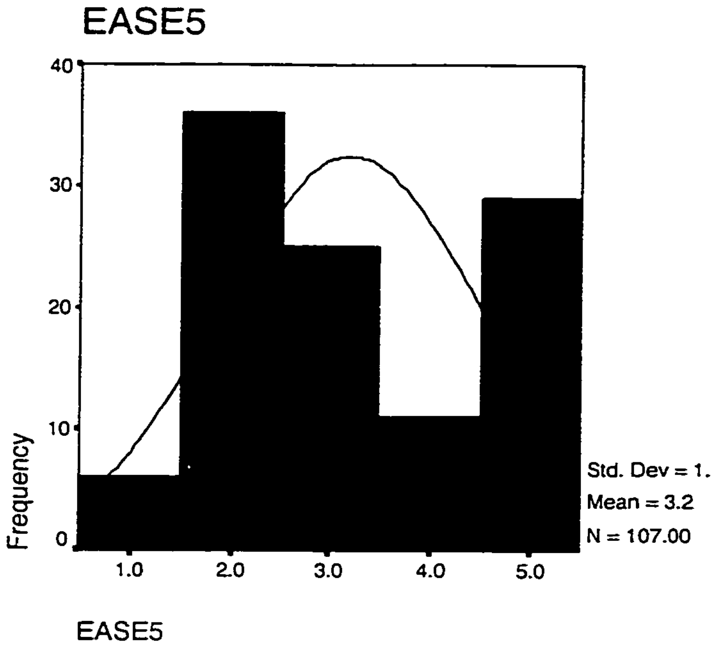
	N	Minimum	Maximum	Mean	Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
USEFUL1	107	1.00	7.00	2.7009	1.60315	1.356	.234	1.124	.463
USEFUL2	107	1.00	6.00	2.5421	1.36865	1.193	.234	.729	.463
USEFUL3	107	1.00	7.00	2.7944	1.66959	1.089	.234	.550	.463
USEFUL4	107	1.00	7.00	2.4299	1.56079	1.574	.234	2.218	.463
USEFUL5	107	1.00	7.00	2.8598	1.48228	.812	.234	-.105	.463
USEFUL6	107	1.00	6.00	2.4953	1.32019	1.226	.234	1.091	.463
Valid N (listwise)	107								

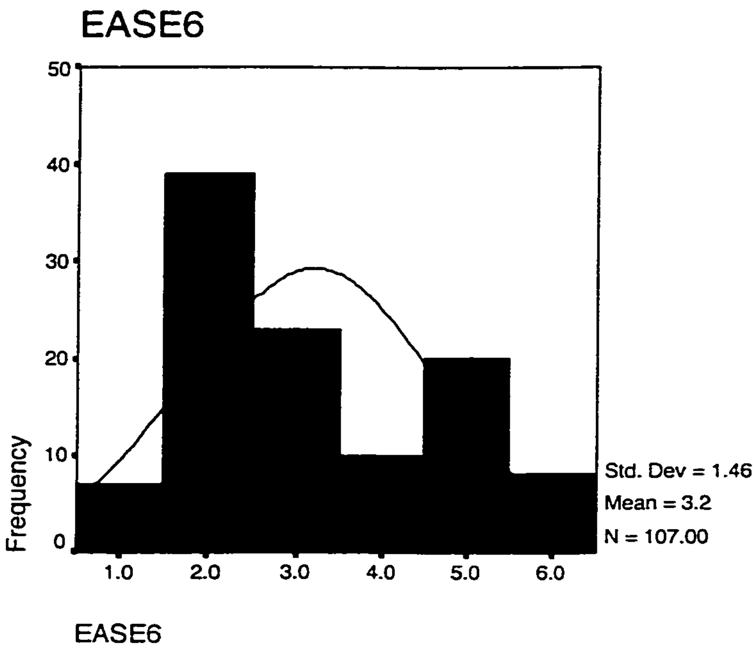

EASE1
Frequency
EASE1


EASE2
Frequency
Std. Dev = 1.40
Mean = 3.2
N = 107.00
EASE2


EASE3
Frequency
EASE3

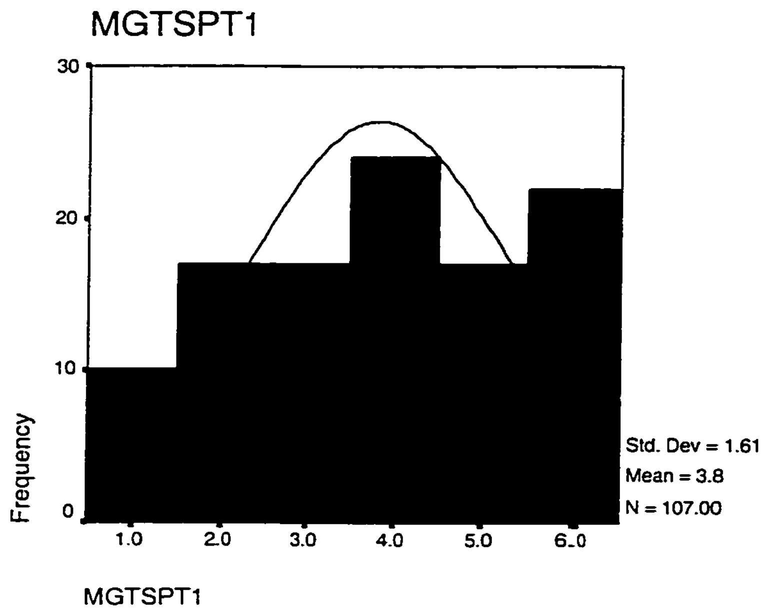

EASE4
Frequency
Std. Dev = 1.37
Mean = 3.2
N = 107.00
EASE4

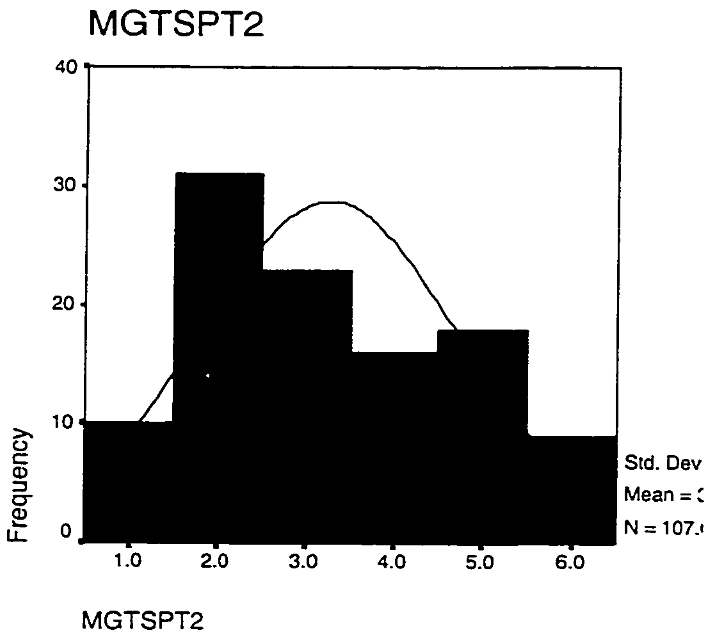


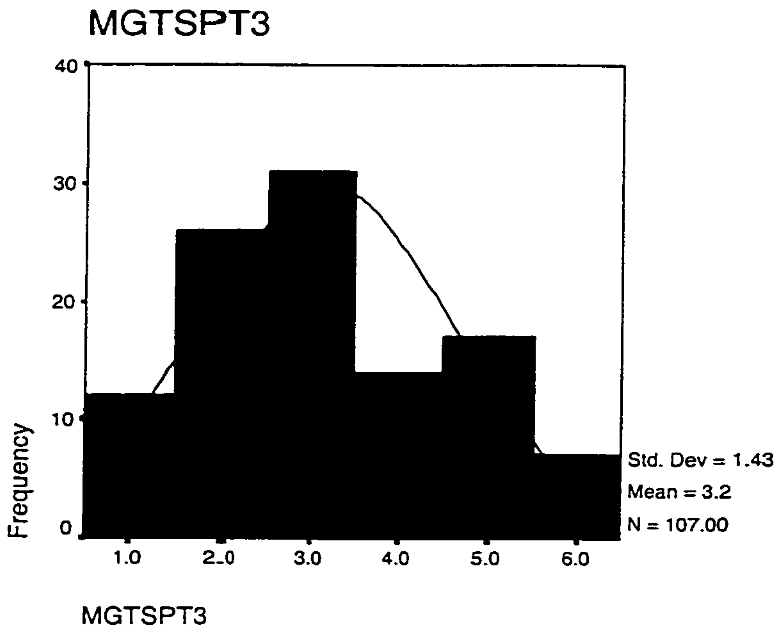


Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
EASE1	107	1.00	6.00	2.9439	1.45908	.712	.234	-.555	.463
EASE2	107	1.00	6.00	3.1589	1.39507	.433	.234	-.799	.463
EASE3	107	1.00	6.00	2.9065	1.43101	.738	.234	-.309	.463
EASE4	107	1.00	6.00	3.1682	1.37007	.497	.234	-.756	.463
EASE5	107	1.00	5.00	3.1963	1.31349	.214	.234	-1.326	.463
EASE6	107	1.00	6.00	3.1963	1.45654	.491	.234	-.973	.463
Valid N (listwise)	107								

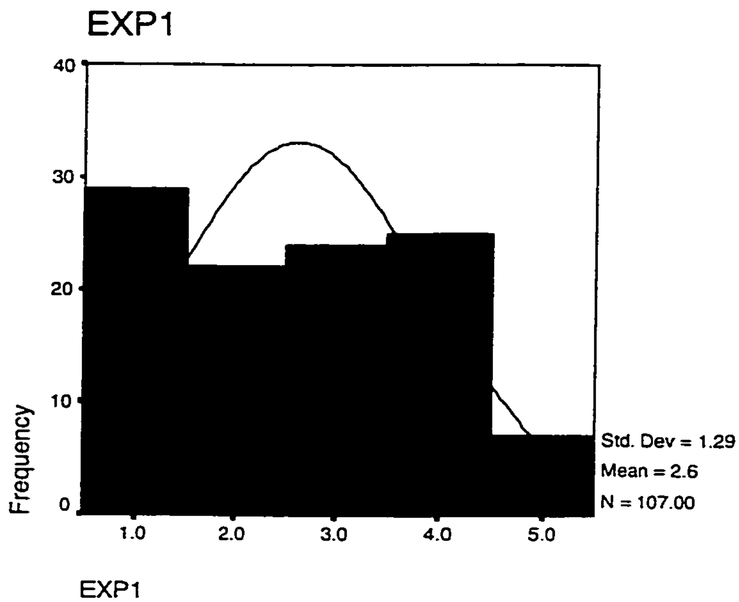


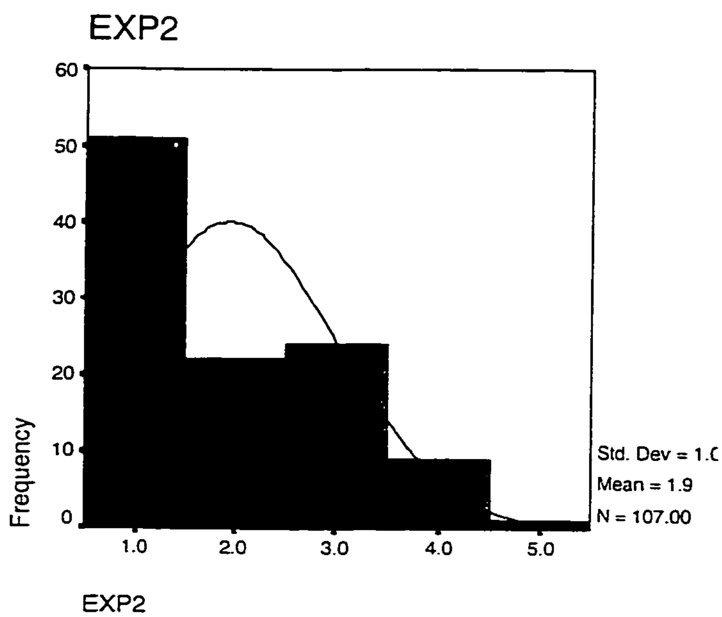


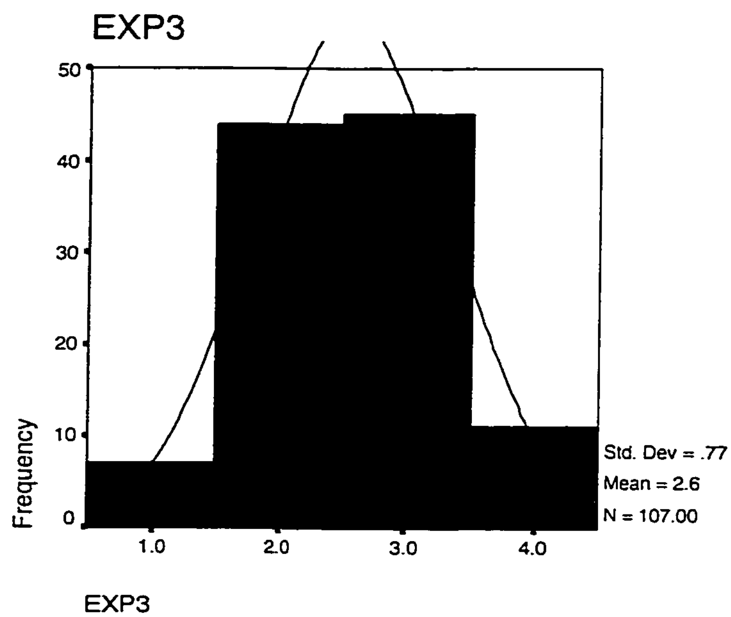


Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
MGTSPT1	107	1.00	6.00	3.8131	1.61439	-.170	.234	-1.096	.463
MGTSPT2	107	1.00	6.00	3.2617	1.48156	.318	.234	-.983	.463
MGTSPT3	107	1.00	6.00	3.1776	1.42626	.336	.234	-.776	.463
Valid N (listwise)	107								

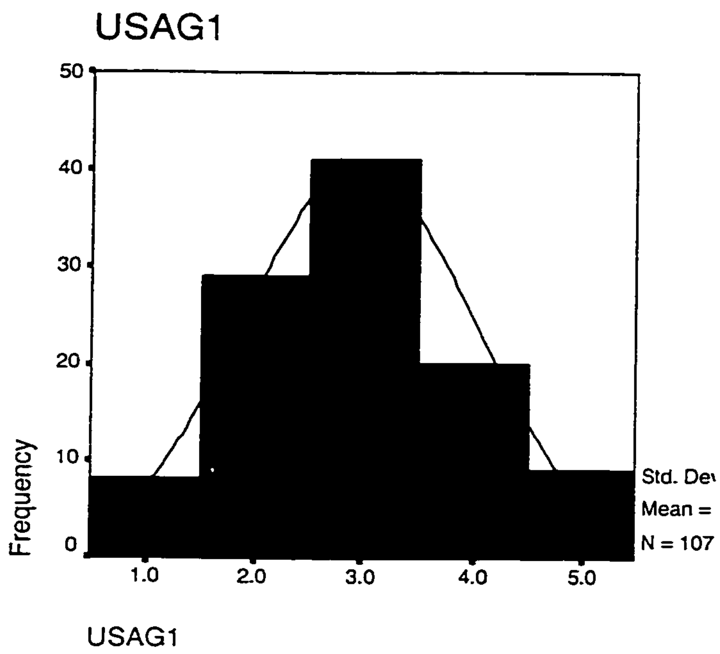


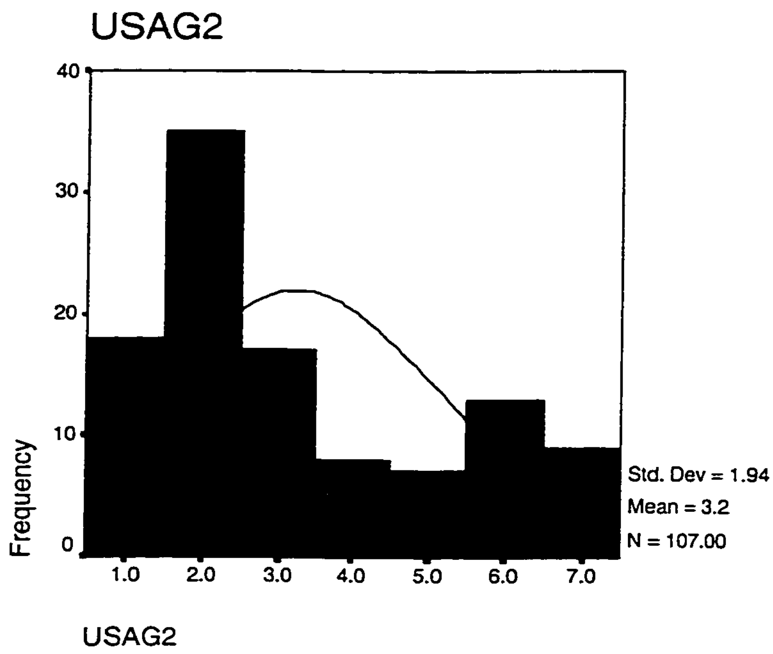




Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
EXP1	107	1.00	5.00	2.6168	1.28595	.156	.234	-1.191	.463
EXP2	107	1.00	5.00	1.9439	1.06250	.739	.234	-.592	.463
EXP3	107	1.00	4.00	2.5607	.76698	.048	.234	-.353	.463
Valid N (listwise)	107								





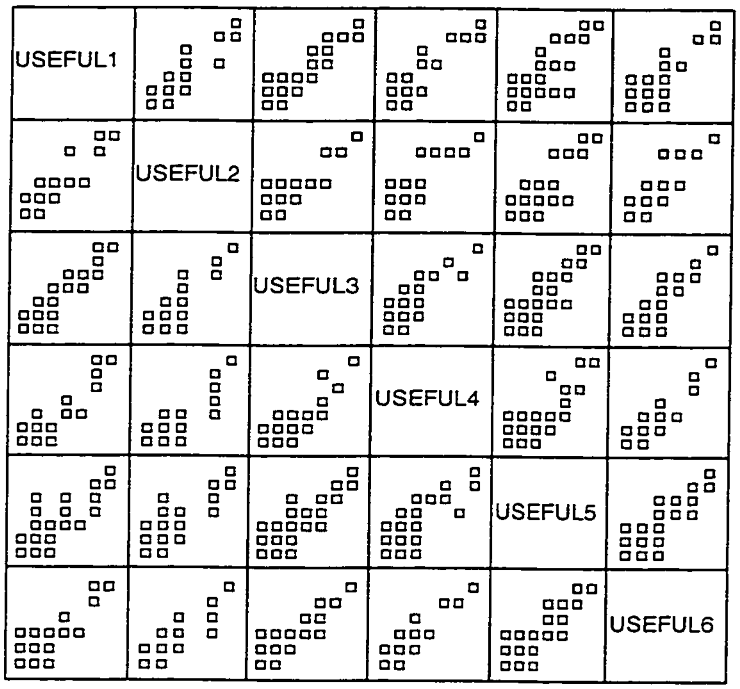
Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
USAG1	107	1.00	5.00	2.9346	1.04855	.183	.234	-.413	.463
USAG2	107	1.00	7.00	3.2430	1.93694	.689	.234	-.851	.463
Valid N (listwise)	107								

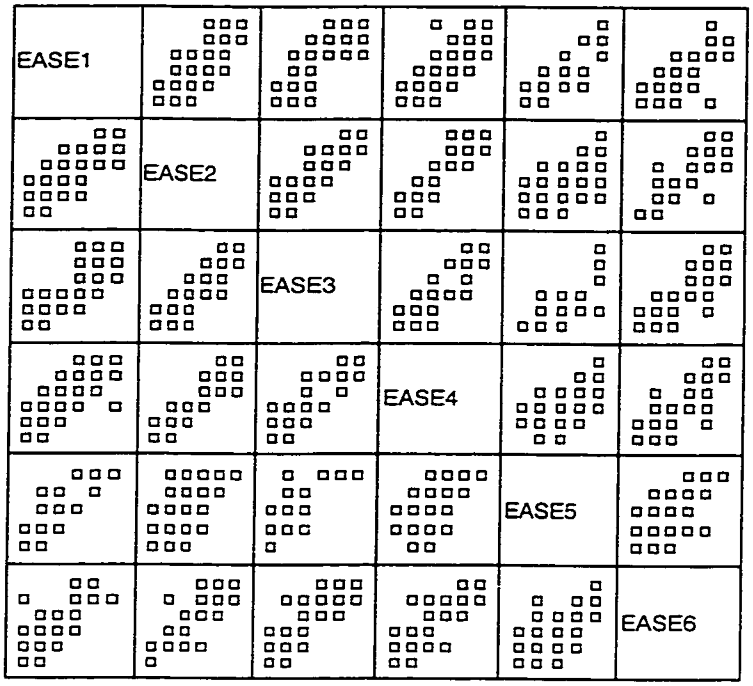
Appendix 13

Variable Scatter plots

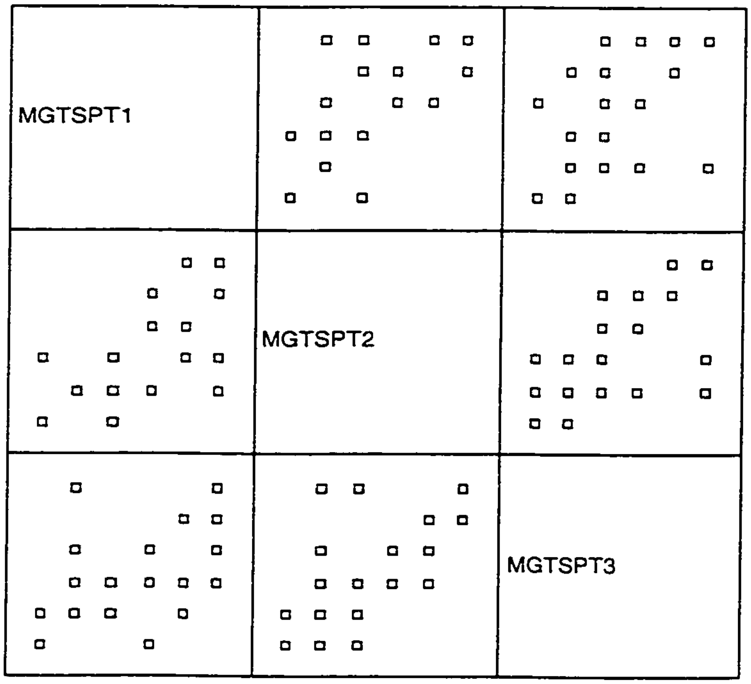
Factor: Perceived Usefulness



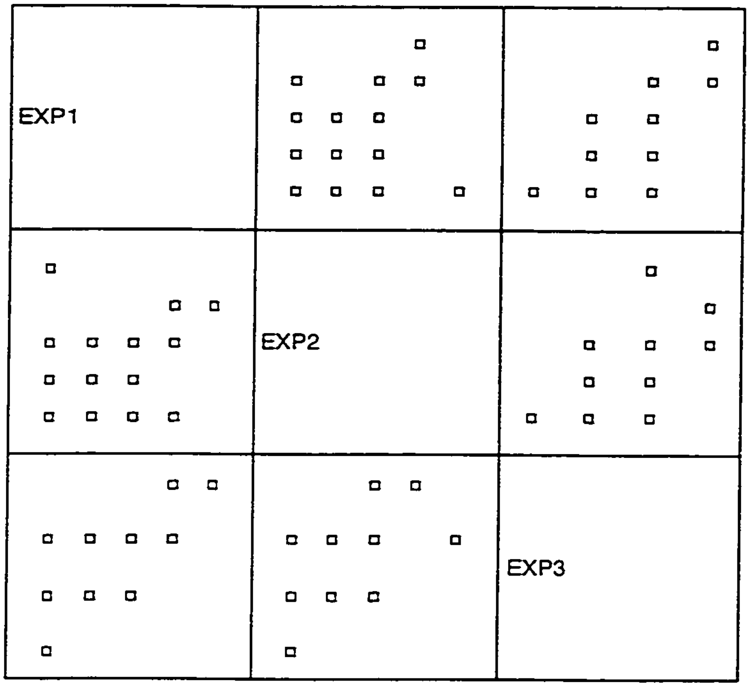
Factor: Perceived Ease of Use



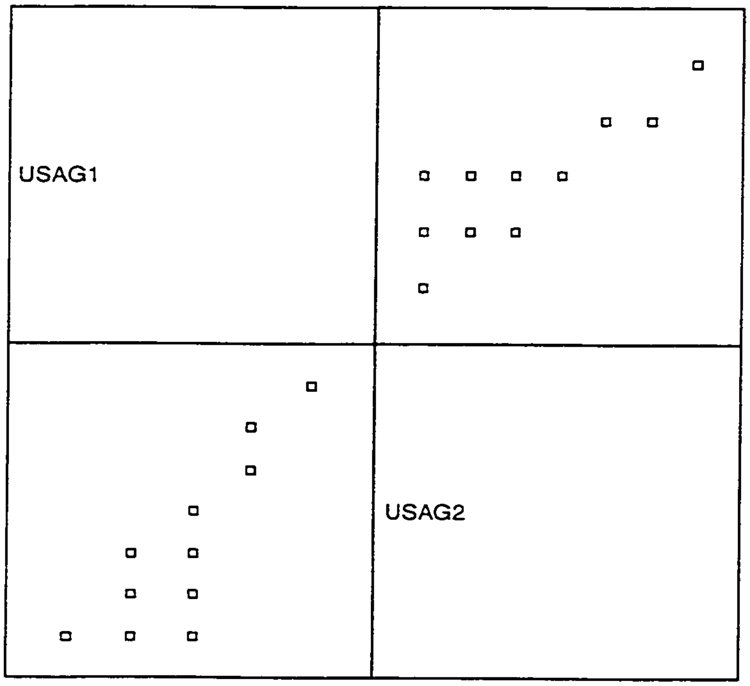
Factor: Management Support



Factor: Experience



Factor: CASE Usage



Appendix 14

Multicollinearity diagnosis

	Collinearity Statistics	
	Tolerance	VIF
USEFUL	.368	2.718
EASE	.415	2.409
MGTSPT	.891	1.122
EXP	.807	1.239

a Dependent Variable: USAGE

Mean VIF = (2.718 + 2.409 + 1.122 + 1.239) / 4

= 1.872

Appendix 15

Scatter plot matrix of standardised average of variables