# TOWARDS A SYSTEM REDESIGN FOR BETTER PERFORMANCE AND CUSTOMER SATISFACTION: A CASE STUDY OF THE ICTS HELPDESK AT THE UNIVERSITY OF CAPE TOWN

A MINI DISSERTATION SUBMITTED IN PARTIAL
FULFILLMENT FOR THE AWARD OF A
MASTER OF PHILOSOPHY IN INFORMATION TECHNOLOGY,
UNIVERSITY OF CAPE TOWN, SOUTH AFRICA



BY
BALIKUDDEMBE JOSEPH KIBOMBO

SUPERVISED
BY
DR. ANET POTGIETER

# ACKNOWLEDGMENTS

I would like to thank the numerous people who have been very supportive during my research. My utmost gratitude goes to my supervisor Dr. Anet Potgieter for her constant criticism, which led to a better research product. I would also like to thank my colleagues at work, Shaan and Werner, for their support in this research. My sincere thanks go to the ICTS management for their cooperation and for granting me the opportunity to work at the helpdesk. This helped me to understand the problem domain better. I also want to thank all the people who participated in the testing of my prototype.

It has been a pleasure to work with you all.

- Joseph Balikuddembe Kibombo

# ABSTRACT

This paper presents the findings from a study, which was carried out to investigate how the design of knowledge management systems could be improved for enhanced performance and greater customer satisfaction. The ICTS Department's helpdesk at the University of Cape Town, South Africa, was the venue for this case study. The study set out to meet the following objectives:

- undertaking a knowledge acquisition strategy by carrying out a systems evaluation and analysis of the existing web-based user support system,
- suggesting a knowledge representation model for an adaptive web-based user support system, and
- developing and testing an online troubleshooter prototype for an improved knowledge use support system.

To achieve the objectives of the study, knowledge engineering techniques were deployed on top of a qualitative research design. Questionnaires, which were supplemented by interview guides and observations, were the research tools used in gathering the data. In addition to this, a representative sample of the ICTS clientele and management was interviewed.

It was discovered that poorly designed knowledge management systems cause frustration among the clientele who interact with the system. Specifically, it was found that the language used for knowledge representation plays a vital role in determining how best users can interpret knowledge items in a given knowledge domain. In other words, knowledge modelling and representation can improve knowledge representation if knowledge engineering techniques are appropriately followed in designing knowledge based systems.

It was concluded that knowledge representation can be improved significantly if, firstly, the ontology technique is embraced as a mechanism of knowledge representation. Secondly, using hierarchies and taxonomies improves navigability in the knowledge structure. Thirdly, visual knowledge representation that supplements textual knowledge adds more meaning to the user, and is such a major and important technique that it can even cater for novice users.

# LIST OF ACRONYMS

| | |
|---|---|
| **CIS**: | Content Intelligent Service |
| **GUI**: | Graphical User Interface |
| **HEAT**: | User support system for ICTS Department |
| **HTML**: | Hyper Text Markup Language |
| **ICTS**: | Information Communication Technology Services |
| **IDL**: | Repository Interface |
| **IT**: | Information Technology |
| **JAD**: | Joint Application Development |
| **KA**: | Knowledge Acquisition |
| **KBS**: | Knowledge Based Systems |
| **KE:** | Knowledge Engineering |
| **KM**: | Knowledge Management |
| **KR**: | Knowledge Representation |
| **LPP**: | Legitimate Peripheral Participation |
| **MS-DOS**: | Microsoft Disk Operating System |
| **OCL**: | Constraint Language |
| **PHP**: | Hypertext Preprocessor |
| **PCS:** | Personal Computing Support group |
| **UML**: | Unified Modelling Language |
| **WINMINE**: | Windows-based Data mining tool |
| **XMI**: | Interchange Format |
| **XML**: | Extensible Markup Language |

# LIST OF FIGURES

# TABLE OF CONTENTS

## CHAPTER FIVE
## SURVEY DESIGN AND METHODOLOGY



## CHAPTER SIX
## CASE STUDY: ICTS KNOWLEDGE BASE REDESIGNING

# CHAPTER ONE
# INTRODUCTION TO THE STUDY

## 1.1 Introduction and Background to the Problem

The recent interest in knowledge management by business communities, consisting of researchers and industry practitioners, has identified organizational knowledge as a competitive asset. Processes and knowledge-based environments are thus being developed to support the knowledge worker. Company knowledge is known to exist primarily in the minds of people, and in business processes, policies and strategies of the organization, as well as in supporting document management systems and information technology systems.

A so-called 'White Paper' drafted by Attar Software [2] suggests that organizations today are making every effort to improve their products and services while controlling costs. Performing complex tasks consequently requires the know-how of the organization's experts and specialists. Unfortunately, there is usually a shortage of such people and their knowledge in most cases is not shared with others, so to speak, rather than being contained in written documents or implemented systems. As a result, if they leave the organization, they take their knowledge with them. Which means that the organisation loses out, whereas the new organisation benefits. In order to reduce such risks of human capital erosion, new business process have to be developed to tap such resources.

The successful implementation of any new business process requires foresight and planning. It also requires perseverance on the part of both employees and managers, as well as an understanding of how the new process will affect corporate culture. And most importantly, the success of the project is always more attainable if there is already a tried-and-tested set of procedures to follow.

Information Technology (IT) organizations continue to face tight budgets and headcount restrictions that threaten to reduce the quality of the services they offer. As a result, some of these companies are turning to use of helpdesk centres in an attempt to assist their users.

The credibility of an IT helpdesk is based on the following key metrics that are often affected by the lack of adequate resources [35]: first call resolution, call handle time, escalation rates and average time in the queue. These metrics are used as gauges to determine the end-user's level of satisfaction and the support agent's success.

Institutions or companies engaged in the delivery of computing services, like the University of Cape Town's Information Communication Technology Services (ICTS), find it challenging to satisfy their users with timely response in instances of problem solving. As a result, online help features have been put in place to ensure that such timely response are indeed given to the customer whenever the need arises. The helpdesk needs to be adequately staffed by trained and highly skilled individuals. Although they can provide workable solutions to the customer, this is not usually the case. In the past, customers have expressed dissatisfaction with responses, e.g. when a helpdesk consultant, while telephonically troubleshooting a given computer problem, uses technical computer jargon not necessarily minding whether the client is an expert or a novice user.



Most solutions in computing today are achieved through working around the problem, rather than solving it. This could largely be associated with the inherent software or hardware limitations. This kind of approach provides the technician with an opportunity to apply the hard knowledge obtained from books or manuals, while combining it with personal experience to arrive at a workable solution. The general steps taken and how the solution is arrived at constitute what can be called un-harvested "soft knowledge", which means that the technician uses the general work experience gained in solving such tasks to solve given computer problems. It is this kind of knowledge that needs to be integrated into a usable system to facilitate adequate knowledge use.

Lave and Wenger [17] suggest that soft knowledge is created, sustained and shared through Communities of Practice. Nickols [20] defines Communities of Practice (CoPs) as groups of people in organizations that form to share what they know, to learn from one another regarding some aspects of their work and to provide a social context for that work. In this regard, it can be the ICTS` technicians and consultant's Communities of Practice used to share knowledge pertaining computer problems and solutions.

In order to ensure improved efficiency and productivity, it is necessary to create a knowledge domain. As an example, such a knowledge domain details possible steps to take in the event of solving or diagnosing a particularly tricky computer problem. This knowledge needs to be tailored to work practices specific to the community. Sacket DL et al. [24] define work practices (best practices) as examples of how *not* to do something. The knowledge that the community constructs about the competencies of its members needs to be incorporated into work practices.

## 1.2 Statement of the problem

The growing number of users and service agreements between ICTS and its clients requires timely and efficient responses to user needs and queries. The telephonic centre at ICTS cannot solve all user problems within a day. It takes considerable resources in terms of time and finances to train new staff on how best to accomplish particular tasks within the current system. Data mining, which is one of the new technologies in assisting knowledge representation, is not yet being utilized in this environment to adapt information to user needs. Substandard information presentation is instead used to present solutions to clients. The solutions do not work, or if they do, they do not work effectively and require follow-up. This is time-consuming and a waste of resources. In other words, it is not only the substandard information that is the problem, but the fact that this information cannot actually solve the problem. This results in further overload on the helpdesk.

## 1.3 Aim of the Study

The aim was to investigate how best the existing web-based user support system could be improved to enhance the efficiency of the helpdesk in the ICTS department at the University of Cape Town.

## 1.4 Objectives of the Study

- To use a knowledge engineering approach to carry out a systems evaluation and analysis of the existing web-based user support system;
- To suggest a knowledge representation model for an adaptive web-based user support system;
- To develop and test an online troubleshooter prototype for an improved knowledge use support system.

## 1.5 Scope of the Study

This study deals with the development and evaluation of the user support software to be used at the University of Cape Town. The physical parameters of this research included the ICTS Department as well as a representative sample from the top ten departments of the University that frequently utilize the helpdesk. The virtual parameters concentrated on daily tasks undertaken within the helpdesk section. For example, it studied, among others, how the consultants talk the users through troubleshooting problems and the kinds of references that they use while answering user queries.

## 1.6 Significance of the Study

The primary aim of the study was to redesign the web-based knowledge base of ICTS. The increasing volume of phone calls to the helpdesk by users is likely to become unsustainable in the future. The ultimate goal is that users will be able to solve their computer related problems by using the web-based knowledge base. However, this knowledge base needs to be redesigned to accommodate all types of users and to improve its performance.

Consequently, it is necessary to ascertain and study the various complex interrelationships between symptoms and solutions to given computer related problems, so that the optimal solution to a given set of observed symptoms can be presented. A given computer problem presents a set of evidence to which a solution can be given. The underlying cause of a problem can only be identified based on given symptoms to which a solution needs to be given. Ontologies provide the means to represent relationships. For this reason, ontologies were used as the main modelling technique to establish the relationships between the concepts in order arrive at a better and more complete representation of knowledge.

With this kind of technology, even novice users may be able to solve their own computer related problems. Not surprisingly, it is a knowledge representation technology that has attracted much research recently.

# CHAPTER TWO

# KNOWLEDGE MODELLING AND MANAGEMENT

Webster's New Collegiate Dictionary [37] defines knowledge as "the fact or condition of knowing something with familiarity gained through experience or association". Knowledge is the next natural and logical progression after information, because it adds understanding and retention to information. In this way, having "knowledge" requires information to exist in conjunction with patterns between data, information and other knowledge, coupled with understanding and cognition [18].

In this research, then, there was a need to determine how knowledge could be captured, interpreted, modelled, filtered and managed so that optimal use by the end user can be maximised. This chapter thus identifies and elaborates on some of the general concepts in the knowledge management field that were considered most important for this study. In order to do so, the relevant literature contained in journals, white papers and articles was reviewed. This review is classified according to knowledge engineering, knowledge acquisition and knowledge modelling and management.



## 2.1 Knowledge Engineering

Knowledge Engineering is associated with the process of developing Knowledge Based Systems (KBS). This differs from conventional software engineering in that it is mainly used during the early stages of the life cycle when user requirements and functional methods (or knowledge) are still being ascertained. Although the particular tools for implementation, user interface design, and testing, maintaining and updating systems may differ, the same principles govern all software systems.

Preece et al. [24] contend that the aim of knowledge engineering methods is to extract knowledge from an application domain and to make it suitable for computer processing. The aim of software engineering methods, by comparison, is to control and optimize the design process of software systems, using relevant quality criteria and requirements management at various abstraction levels.

Consequently, although these two methods are closely related, what distinguishes knowledge engineering from software engineering is the unique range of techniques of the former for knowledge elicitation and modelling, as well as offering a collection of formalisms for representing knowledge.

In this research, early stages of knowledge acquisition involved knowledge engineering techniques and domain expertise. Later stages involved the implementation and integration of software engineering techniques. Since the expert's knowledge must be represented at the knowledge level rather than at lower levels during knowledge acquisition, the computational representation at high levels normally persists throughout a KBS life cycle.

In this research, it was necessary to exploit core knowledge engineering processes that include: requirement analysis, conceptual modelling, knowledge base construction, operationalization and validation, as well as refinement and maintenance. The advantage of this was that it allowed users and experts to describe requirements and methods to perform the required functions at a high level close to the level in which they thought about the task: i.e. at the knowledge level.



## 2.2 Knowledge Acquisition

Knowledge acquisition has been broadly defined as the process of eliciting, analyzing and interpreting the knowledge that a human expert uses when solving problems [16]. This whole process is broadly summarised as requirement analysis.

ICTS realized the need to structure and manage knowledge online through a knowledge base in order to enhance its wide area distribution and access of such resources to its clients. This was because it had been determined that the ICTS clientele needed to access such knowledge resources at the time that was most suitable and convenient for them, rather than having to phone the helpdesk during limited working hours.

GodBout and GodBout [10] define a knowledge base as a collection of facts, relationships and rules, which embody current expertise in a particular area.
ServiceWare [28] reckons that if you do not offer your customers or employees web-based self-service, then you have already fallen behind the times. As a result, there was a need interpret the acquired knowledge into a more representative knowledge structure so that a customized redesigned system for better performance could be achieved.

In knowledge acquisition, the relevant knowledge that is pertinent to a specific knowledge domain needs to be elicited from wherever it might be (i.e. human experts, records). Knowledge representation consists of various ways in which the acquired knowledge is expressed and structured so that it can be dealt with in a meaningful way by the machine. The knowledge acquired and maintained by an organization is, after all, one of its most important assets. How organizations obtain and use their knowledge provides the foundation of knowledge management.
In this research, requirement analysis was used to capture organizational knowledge The Unified Model Language (UML) used in modelling software projects was used for knowledge modelling and representation, and thereby to help the client to understand the captured knowledge graphically. Zhu [38] holds that well-structured representation of system requirements can dramatically improve the communication among analysts, designers, users and programmers.

The design and implementation of better and more effective knowledge acquisition practices require the designer, the client and the end user all to be involved, so that optimal system design is maximized. The most recommended methodology for achieving this is known as Joint Application Development (JAD), a method designed by Chuck and Crawford [6]

JAD involves the client or end user in the design and development of an application, through a succession of collaborative workshops called JAD sessions [6]. Joint Application Design (JAD) and prototyping were used in this study because they are two modern methodologies of determining requirements, which are based on and developed from previous traditional methods.

It is generally agreed that most knowledge resides in the heads of experts, and that they have vast amounts of knowledge, most of which is not written down in any consistent format. In this study, the experts included the ITS telephonic consultants as well as the PCS consultants. Interviews were also conducted with ICTS knowledge experts in order to explore the scope of the existing knowledge repository.

Preece et al. [24] maintain that at the knowledge acquisition stage, where stakeholders in the system are interviewed, scope of the knowledge-based system is identified; typically in terms of its expected competency (for example, the kind of queries it will be able to answer).

The benefit of using such a knowledge engineering technique was that, once the requirements analysis has been successfully completed and knowledge has been captured and recorded, a knowledge base could then be utilized by automated systems whose aims are the following:

- To enhance human decision making by offering advice;
- To free experts from repetitive routine decisions for more productive and rewarding work;
- To ensure that decisions are made in a consistent way and as speedily as possible; and
- To retain the organization's expertise in a readily maintainable form.

AttarSoftware [2] contends that knowledge acquisition is an iterative process, best performed with interactive tools engineered specifically to capture knowledge. In this study therefore, this iterative process was realised with utilisation of interactive tools like SELECT Enterprise. This whole process gave rise to knowledge modelling and subsequently knowledge management.

## 2.3 Knowledge Modelling and Management

VNU Business Media [36] defines Knowledge Management (KM) as the process of capturing, storing, transforming, and disseminating information within an organization, with the goal of, at least, promoting efficiency and, at best, facilitating innovation and competitive advantage. KM was specifically used in this research because of the benefits it can offer. Hariharan [12] perceives that sustained strategic commitment and a corporate culture that is conducive to knowledge-performance are vital for success in Knowledge Management. Technicians and consultants of ICTS have to be committed for knowledge management to succeed. This has to be grounded in the departmental culture to which ICTS management must initiate.

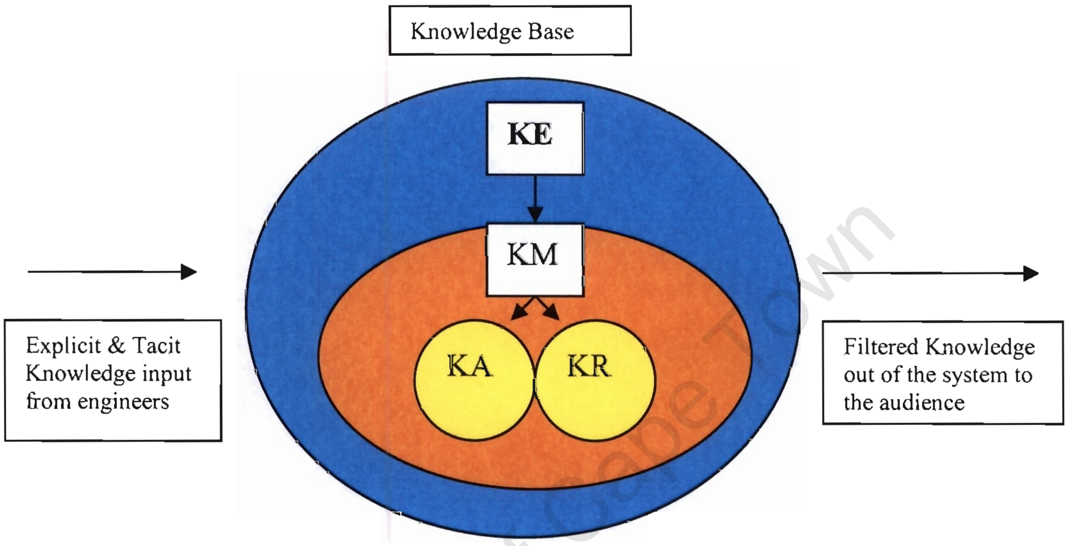
Smith [29] observes that knowledge management is widely accepted as valuable means for organizations to enhance their intellectual capital, to encourage innovation and to optimize performance. Benjamins [3] urges that organizational processes and cultural issues are important. When managed correctly, knowledge can be turned into a strategic asset of any knowledge-intensive organization. Tobin [35] maintains that in today's knowledge economy it is very likely that people are the most important assets of an organisation, and that this is largely due to their knowledge. Essentially, when they leave the organization, they leave with the knowledge that they have acquired. By embracing an improved knowledge management approach, ICTS can manage its knowledge assets effectively so that innovation is supported.



An investment in Knowledge Management (KM) is not a quick solution, as it may take several iterations of real input and measurable output and subsequent updates, before a good KM system is in place. Accordingly, continuous refinement and maintenance are vital. Preece et al. [24] assert that after a system has been developed and put in place, it will continue to evolve as knowledge changes. Knowledge management is thus often embedded in organizational structures that enable organizational learning [22]. Knowledge in the ICTS knowledge base needs to be refined regularly so that adequate maintenance is enhanced.

Knowledge engineering in this study therefore, formed the basis for the whole system development. It includes knowledge management, which is further subdivided into knowledge acquisition and knowledge representation as illustrated in figure 1.

**Figure 1: Knowledge Base Interdependence**






In figure 1 above, knowledge has to flow into the system, where it is processed and filtered. It is then presented to the audience (in this case the clients), using a given technology or methodology. The underlying architecture that leads to the establishment of a knowledge base is embedded in knowledge engineering, which in turn incorporates knowledge management. This, then, has the sub-components of knowledge acquisition (KA) and knowledge representation (KR).

With a proven set of best practices, organizations can successfully integrate a KM system into their workplace and ensure its continued viability and growth for years to come [15].

# CHAPTER THREE

# KNOWLEDGE REPRESENTATION USING ONTOLOGIES

## 3.1 Knowledge Representation and Use

Knowledge representation (KR) is the study of how knowledge about the world can be represented and what kinds of reasoning can be done with that knowledge [9]. A system under development will need to be transformed into layman's language so that the client(s) can properly understand the system overview before it is implemented. One of the best practices today in system modelling is the use of the Unified Modelling Language (UML). Object Management Group (OMG) [23] defines UML as a language for specifying, visualizing, constructing, and documenting the artifacts of software systems. It can also be used for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

UML provides a set of notational conventions that developers of information technology systems can use to describe an application. This description is usually done in a precise abstract fashion where a specification of relationships between all elements in a metamodel is given [14]. In this study, the UML metamodel description was used in order to capture and visualise system requirements in an abstract fashion so that the clients could understand the system under development much better.



## 3.2. What is Knowledge?

Storey[32] views knowledge in four dimensions as:

- A computable model of some domain for some purpose
- A competence-like notion, being a potential for generating action
- A justified true belief (philosophical definition)
- The specification of what a symbol structure should be able to do.

In this study, knowledge was considered to be a an expertise skill, being a potential for generation of action. For example computer problems determine the kind of solution to be given to a user.

Staab et al. [30] maintain that an IT-supported knowledge management solution is built around an organizational structure that integrates tacit and explicit knowledge to facilitate its access, sharing and reuse. Tacit knowledge can be referred to as the "**know how**" that belongs to someone, while explicit knowledge can be referred to as the **"know that"**, which is public and can be shared [4]. In the present context, where knowledge has to be modelled, structured and interlinked, ontologies can help to formalize the knowledge shared by a group of people. Knowledge in the web-based knowledge system can thus be systematically modelled and represented by using ontologies.

### 3.3 Utilizing Ontologies

Swartout and Tale [33] hold that an ontology is the basic structures or armature around which a knowledge base can be built. However, an ontology is not a programmatic representation; it addresses domain conceptualizations [8].

Gruber [11] similarly, defines an ontology as ''the specification of conceptualizations, used to help programs and humans share knowledge'.

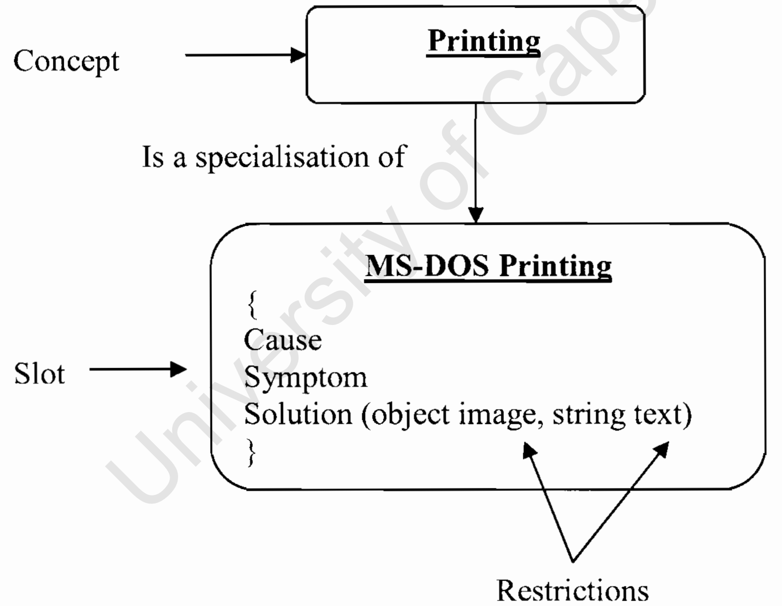
Stevens [31] contends that **conceptualization** is the formulation of knowledge about the world in terms of entities (i.e. things, the relationships that link them and the constraints that exist between them). **Specification** refers to the representation of this conceptualization in a concrete form.

On the whole, an ontology characterizes itself with respect to mechanism and content. It could consist of a conceptual model, a thesaurus and a set of expanded attributes and axioms, with the main concern of appropriate representation of content.

Musen and Gruber [19];[11] emphasize that the common goal in developing ontologies is to further share a common understanding of the structure of information among people and parts of software. From that perspective, a troubleshooting ontology specific to ICTS was developed in this research. This consisted of a formal explicit description of concepts in the knowledge domain of discourse (classes/concepts), the properties of each concept describing various features and attributes of the concepts (slots/roles/properties), and the restrictions on the slots (facets/role restrictions).

- Classes describe **concepts** in the domain. For example, UML was used in Figure 2. A class 'Printing' represents all problems related to printing in this troubleshooting ontology. Specific printing problems are a specialization of printing problems; for example, MS-DOS printing is a specialization of printing problems.
- **Slots** describe properties of classes and instances. For example, the slot describing properties for the instance concept 'MS-DOS printing' includes its associated symptom, cause and solution. The restrictions on solution as a property for the instance class MS-DOS printing could be diagrammatic or textual as illustrated in figure 15. Restrictions on the property are used to present the solution to the user.

**Figure 2: Ontology Description**




Properties of MS-DOS Printing slot are the cause, symptom and the solutions. Restrictions on the solutions property are the image and text.

A knowledge domain can be defined as an area of knowledge that the community agrees to learn about and advance. It is negotiated among the participants, especially among the community's experts [1]. The focus for this research was the accessibility of and knowledge representation in the ICTS Knowledge base.

Ontologies were generally considered useful for this study because they offer a number of advantages to knowledge base development. They are the glue that binds knowledge subprocesses together. They allow us to move from a document-oriented view of knowledge management to a content-oriented view, where knowledge items are interlinked, combined and used. In ICTS for example, attachments are added to resolutions where more explanation is need. The following fundamental rules [21] in ontology design were used for this research.

1. There is no one correct way to model a knowledge structure in a given knowledge domain. There are always viable alternatives. The best solution usually depends on the application that you have in mind and the extensions that you anticipate.
2. It is always necessary to know that an ontology development is an iterative process.
3. Concepts in the ontology should be close to the objects (physical or logical) and relationships in your domain of interest. These are most likely to be nouns (objects) or verbs (relationships) in sentences that describe your domain.

### 3.4 Taxonomies Aiding Knowledge Discovery

Delphi Group [7] defines a taxonomy as a hierarchical or polyhierarchical listing of topics or subject categories. It may not include definition of the topics but only the hierarchical relationship of topics to one and another. A taxonomy can incorporate content from both a thesaurus and an ontology. In this research, a taxonomy was used to provide a structural navigational path through a content collection of the knowledge base.

Relationships between concepts are fundamental to building an ontology. They describe the interrelations between concepts or a concept's properties. *Taxonomies* organize concepts into sub- and super-concept tree structures. They could be specialization relationships commonly known as the ''is a kind of'' relationship or partitive relationships, which describe concepts that are part of other concepts [7]; [19].

One of the major challenges in structuring content in a knowledge base is to know how best its users can access the content. In overcoming such problems, taxonomies can offer the following advantages:

- **Intelligent knowledge processing**: It is always difficult to mine answers from unstructured content. This is because people can distinguish between the specific meanings of a word or concept, based on the context within which the word is used, but computers cannot. Other methods must therefore be used to help computers to provide intelligent knowledge processing. This can be achieved with the use of taxonomies.
- **Context**: People explore concepts, and primary searching thus occurs for key words. The relevance of the concepts or key words is subjective to the individual who performs the search. Only each individual can judge how relevant a particular bit of information is to what they are attempting to discover. The document may be too technical, out of date or too general to your needs. If you have categories and hierarchical structures of knowledge, you will however be able to narrow the search field and to find the relevant information faster.
- **Browsing vs. Searching:** Effective search of content without benefit of a taxonomy requires that you have to know the words and phrases to use before you can see what is in the collection of the knowledge base. Keyword searching assumes that you know what you are looking for, which is usually an erroneous assumption. Browsing, on the other hand, encourages associative thought and it guides one through the knowledge discovery process. Browsing is complementary to searching but not superior to it. Visual, hierarchical arrangements of subject categorization trigger associations and relationships that may not be obvious when initially searching in most instances.

Whether navigation is used for discovery, research or education or whether searching is used as targeted retrieval of information, business users turn to searching and taxonomies to find relevant knowledge quickly and intuitively, and to support better informed and hence more effective decisions and actions.

Figure 3 illustrates an example of a hierarchical knowledge representation of categorisation approaches using taxonomies. Categorisation can be done manually, automatically or using a combination of both. The manual approach has both advantages and disadvantages. The automatic approach could either use a rule based approach or training by example.

**Figure 3: Taxonomy example**

**Categorization approaches**

- Manual
  - Disadvantages
    - Does not scale well
      - Does not keep with rapid change of info
    - Inconsistency When Classifying
    - Classifiers May Find Work Boring
    - Classifiers are Expensive Assets
  - Advantages
    - Scales Well...
- Automatic
  - Rule Based
  - Training by Example
  - Semantic Analysis
  - Clustering Analysis
- Combination



There are a couple of companies worldwide engaged in developing commercial taxonomies for representing knowledge. Examples include:

- **Documentum** (http://www.documentum.com)

  Documentum develops content applications that power the Documentum platform to determine distinct content management challenges by using taxonomies. Content Intelligence services (CIS), a Documentum product, automates and controls the attribution and organization of enterprise content based on powerful information extraction, conceptual classification, business analysis, and taxonomy and metadata management capabilities. This enables advanced searching, easier navigation, and more effective personalization to facilitate content reuse and enrich customer applications.

- **Autonomy** (http://www.autonomy.com)

  Autonomy's taxonomy generation feature can automatically and consistently understand and create deep hierarchical contextual taxonomies of information based on conceptual understanding. The resulting taxonomy can be used to provide insight into specific areas of the information, an overall 'information landscape', or training material for the automatic categorization that then allows information to be placed into a formally dictated, controlled and named category hierarchy.

- **Inxight** (www.inxight.com)

  Inxight's technologies and applications can transform an organization's reservoir of unstructured content into a well-organized repository of knowledge. Organizations can apply categorization, extraction and visualization technologies in several ways to increase operational effectiveness using this technology.

With such tested technology and knowledge representation techniques mentioned above, major areas addressed by such techniques were adopted in this study. For example, the use of Bayesian Networks. This technology will be explained in the next section.

# CHAPTER FOUR
# THE POWER OF BAYESIAN NETWORKS

Knowledge management systems, whether web-based or not, need to utilize technologies that minimize redundancy in knowledge representation. This can be done by using data mining techniques, where uncertain patterns in knowledge are observed and learned, and relationships are drawn to optimize knowledge use.

Thearling [34] asserts that data mining technologies and techniques for recognizing and tracking patterns within data help businesses to sift through layers of seemingly unrelated data for meaningful relationships, allowing them to anticipate, rather than simply react to customers' needs.

Knowledge in a web-based knowledge system can be systematically modelled and represented by means of Bayesian Network technology. Bayesian Networks provide a powerful capability for probabilistic reasoning and inference that was applied to the uncertainty in the ICTS environment. Bayesian inference and learning allowed us to explain hidden patterns between concepts.

According to Heckerman [13], a Bayesian network can offer several advantages for data modelling. This is because the model encodes dependencies among all variables; it thus readily handles situations where some data entries are missing. It can also be used to learn causal relationships, and hence can be used to gain an understanding about a problem domain and to predict the consequences of intervention among others. With such a powerful knowledge management engine (Bayesian Network) in place, customer support and knowledge sharing and representation can be enhanced.

## 4.1 Overview of the Current System

The current web-based system in place at ICTS is based on an automatically generated output of the HEAT System. It confines users to only accessing the solutions that have been entered into the system by subjects. Solutions are authored after technicians realize persistent problems arising over a period of time, say two weeks. These solutions are then entered into the HEAT system, which automatically generates it to the Web-interface. Some of these solutions are too technical though, and it becomes very hard for a novice user to locate the information needed quickly. The search facility created on the web system does not work with or indicate close relationships between solutions.

## 4.2 Knowledge Engineering and Modelling

As a solution to this problem, I re-engineered the knowledge in the HEAT system for customer adaptability and efficiency. I looked at this in the following five dimensions: problem, evidence, symptoms, cause and resolution

- **Problems** are symptoms that are significant enough to cause a user to access the web-based system. In this case, he or she will access the online troubleshooter. For example, *"My printer doesn't print."*
- **Evidence** refers to the observable phenomena in this system. It may or may not indicate a problem or cause. For example, "*I am getting a 404 error when I try to go to open a certain webpage."*
- **Symptoms** compromised groups of evidence that were parallel. These may contain two pieces of evidence: a test succeeds, or it fails. It can contain more, for example, *"I am using Windows NT."*
- **Causes** were considered as the underlying reasons for why the user was having a problem in the first place. However, the user does not have to know the cause in order to fix it. In future, it will be the author of the solution who defines the cause, so that the knowledge engineer can model it more effectively. For example, "*The printer cable is faulty.*"
- **Resolutions** are the instructions for fixing a problem. They are usually associated with a single cause, but not always. For example, removing and reinstalling MS Word may fix many causes, such as, among others, a corrupted system file or an incorrect registry entry. Likewise, more than one resolution may fix a single cause, for example, deleting or renaming the Normal.dot file may also fix the cause of a corrupted Normal.dot file, because Word will create a new one.

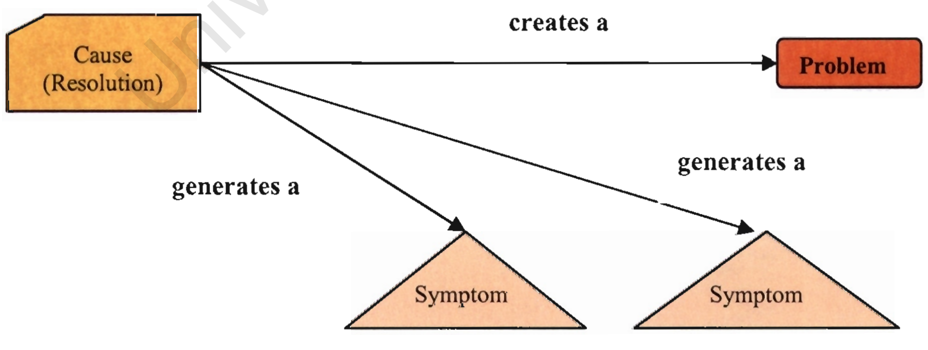
### 4.3 Bayesian Network Construction

The modelling and construction of the online troubleshooter was based on the following types of nodes:

- Problem nodes,
- Informational nodes,
- Fixable/Observable nodes,
- Fixable/Unobservable nodes and
- Unfixable nodes.

A **problem node** is a node that represents a possible primary symptom experienced by a user. Instantiating a problem node begins a session. Problem nodes contain information from problems provided by the authors. An **informational node** contains a single symptom, which in turn is a collection of related, exclusive pieces of evidence. A **fixable/observable node** represents a cause that has both a symptom and a resolution. The symptom should specifically identify this cause, and the resolution should be the most effective one for the cause. A **fixable/unobservable node** contains only a resolution. Generally, this type of node is of use if the only way to detect a cause is to resolve it, or if there are multiple resolutions for a single cause. **Unfixable nodes** contain problems that require a user to log a call so that his/her machine is worked on at the ICTS workshop.

**Figure 4: Web-based troubleshooter model**




The web-based troubleshooter model is illustrated in figure 4. A model works from the cause to the problem, while a session progresses from the problem to the resolution.

**Figure 5: Web-based troubleshooter session**

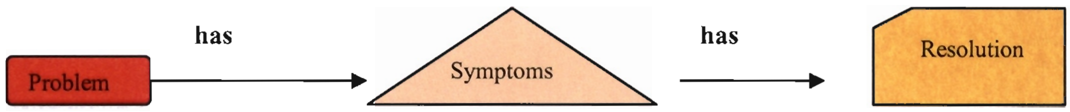



Figure 5 illustrates a web-based troubleshooter session. The web-based troubleshooter solves a user's problem in a linear fashion, starting with the problem, and inferring with the symptoms, to arrive at a resolution. In this linear fashion, a problem has symptoms. Given set of symptoms have resolutions that can be used to solve a given problem.

### 4.4 Causal dependence and Inference

**Figure 6: Inference**

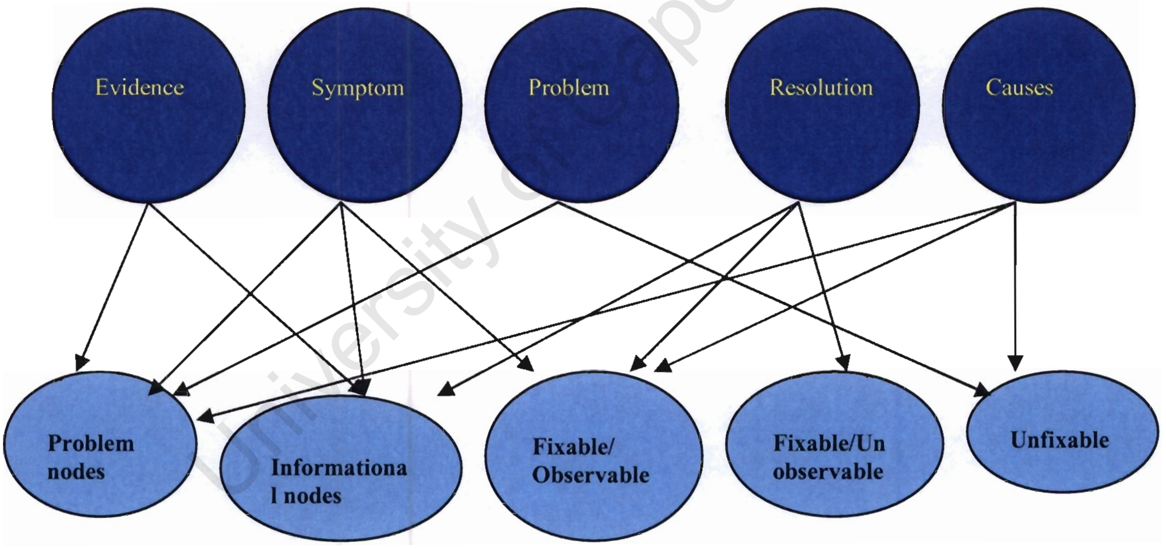





Figure 6 illustrates relationships between knowledge items. The **problem nodes** depend on the evidence around the problem, the symptoms given, the problems themselves, as well as the possible causes. The **informational nodes** depend on the resolutions, given the evidence of a collection of symptoms. In this model, inference is made on the various child nodes so that the web-based troubleshooter executes the right algorithm to generate relevant solution based on the user's problem.

# CHAPTER FIVE
# SURVEY DESIGN AND METHODOLOGY

This chapter discusses how the research was carried out and addresses the following areas: research design, area and population of the study, sampling strategies, data collection methods and instruments, data quality controls, and data analysis and presentation.

## 5.1 Survey Design

In order to gain a clear picture on the current state of user support software used in the ICTS department, an analytical qualitative study was undertaken. This type of approach was chosen due to inventions in the study, which called for descriptive research methods and analysis of documents and peoples' views and opinions. This research provided the system specifications in accordance with which a prototype was developed. This prototype in turn created a framework for an effective and efficient user support system in the ICTS department at the University of Cape Town.

## 5.2 Description of the Study Area

The study was carried out in the ICTS Department at the University of Cape Town, South Africa. The study area incorporated all the various sub-departments within ICTS that proved helpful in the study, as well as all the other University departments that frequently use the helpdesk services.

## 5.3 Description of the Population of the Study

The study population considered in this research included the administrators of the information technology helpdesk, the staff at the ICTS helpdesk, and other identified stakeholders in the system development and evaluation.

## 5.4 Sampling Strategies

In an effort to establish a framework of how best to adapt the present system so that it can be used by all kinds of users, a non-probabilistic sampling approach was implemented. In such an approach a representative group is observed for purposes of illustration or explanation. This sampling method was vital in collecting information from a category of knowledgeable respondents. This kind of sample was chosen because of their potential role in both managing and interacting with the current system.

## 5.5 Data Collection Methods and Instruments

In undertaking a systems analysis and evaluation of the current system, a combination of various instruments for data collection was used, including questionnaires, interviews, observation and document (content) analysis.

- ***Unstructured Questionnaires:*** The aim of the questionnaires was to obtain opinions and subjective information on the prototype being developed. Closed-ended questions were included in the questionnaire to capture respondents' attitudes towards some of the issues. This created a basis for analysing users' requirements.
- ***Interview Guide*:** In addition to the questionnaire, an interview guide was prepared for the management staff of the helpdesk section. The interviews usually began with a self-introduction of the researcher. Selected respondents were interviewed to supplement the information that had been obtained from the questionnaires. The interviews were both formal and informal and, given the busy schedule of the respondents, did not last more than thirty minutes. The formal interviews were administered in accordance with the interview guide. The informal interviews, which are intended to deduce and acquire information that is otherwise not given in formal interviews, did not use the guide. The confidentiality of the respondents and their responses was always assured. During the course of the interviews, it often became necessary to reassure respondents of the confidentiality of their responses to avert doubt or hesitations in them expressing their views.
- ***Observation*:** The researcher frequently visited the department and other targeted areas to examine how users were interacting with the current system, and the difficulties they were facing. In addition, the researcher shadowed at the helpdesk as a telephonic consultant, helping users to solve their computer related problems telephonically, whilst making reference to

the web-based system. This helped the researcher to become conversant with the current situation of the user support systems in the ICTS department as well as scoping the domain of study.

- ***Document Sampling and Analysis***: This formed part of data collection. The purpose of this was to study the corporate vision of the ICTS department, the standards required as well as other issues pertaining the study, such as departmental standards with regard to information presentation.

Casual discussions supplemented the above mentioned data collection methods. This was possible in instances when officers dealing with various issues were encountered in their offices or any other convenient places.

### 5.6 Prototype development

Java, C#, XML, and HTML were used as development languages for the online troubleshooter prototype. Knowledge acquired through analysis and design was the basis for development. Development took into account Human Computer Interaction standards such as usability as a quality objective. Rational Unified Process was used as a guiding tool in the systems analysis and design [25]. Protégé [26] was used as an ontology modelling tool for knowledge representation in the knowledge domain. The front-end user interface was connected to the backend database that was developed in MySQL. This database catered for security of information, database expansion as well optimization for information retrieval.

### 5.7 Data Quality Control

Data quality control refers to whether the data is up-to-date and accurate. Given the abilities and knowledge of the respondents in this study, it is fair to assume that the data collected was indeed valid. The targeted population, after all, possessed the knowledge to respond to the questions on the current system. The researcher's supervisor moreover cross-examined the data collection instruments. All the questions were checked to verify whether they had been properly answered, and inconsistencies were eliminated.

## 5.8 Data Analysis

The analysis was done according to content, in other words, selected records, documents, published and unpublished information were subjected to content analysis. Data was recorded thematically according to the objectives of the study. For the purposes of analysis, the structured data in the questionnaires was matched according to the category of respondents. Data was recorded thematically, interpreted and presented with the help of direct quotations with regard to the extent of the problem in comparison to the available data or information. The frequencies of the various responses were tallied for analysis purposes, and averages and rankings of discrete responses were made. Descriptive research analysis was used for closed-ended questions that demand discrete responses.

## 5.9 Data Presentation

After the data had been collected, it was sorted, analyzed and presented according to each objective. Before the data was presented, however, it was inspected and edited to avoid distortion of the message from the respondents. Anonymous quotations were noted down from questionnaires, actual interview, casual discussions and open-ended questions, and observational descriptions were made.

## 5.10 Limitations of the Study

There were several limitations that were faced in this study. These included:

- **Access to Information:** One of the limitations of this study was the lack of access to information that was vital in the study. Some of the information was regarded as sensitive according to the various department policies governing information. Some respondents were moreover hesitant to give information while others wanted to protect their own interests.
- **Time**: This being a qualitative study, it demanded much of researcher's time and personal involvement given the time limit in which the degree had to be completed.

# CHAPTER SIX

# CASE STUDY: REDESIGNING THE ICTS KNOWLEDGE BASE

This chapter highlights the findings from the study carried out at ICTS to redesign its online knowledge base. It is structured according to the knowledge engineering principles that were used, that is, requirement analysis, conceptual modelling and knowledge base construction, operationalization and validation. Thereafter, two tests were conducted on basis of on usability of the prototype and the other centred on the comparison between the prototype and the existing system.

## 6.1 Requirement Analysis

In developing the online troubleshooter, knowledge capturing was undertaken in terms of systems analysis and evaluation. This involved capturing the tacit knowledge and incorporating this into the system to be improved. The underlying objective was **to redesign the Web-based knowledge base into more usable support systems**.

A series of interviews were conducted with management and the repository managers in order to explore the scope of the existing knowledge repository. The domain expertise used considered mechanisms of listening to both novices and members on the boundary in the sample in order to explain and reinforce the knowledge domain. The main area of focus in this regard was the deep structure that would be necessary to organize the domain knowledge and to work towards solving problems in this domain.

The key finding was that the system ought to be very user-friendly to cater for all categories of users, because the knowledge base serves many people at different levels of computer literacy, and because its content is constantly evolving. The knowledge given also ought to be current and relevant to the clientele's needs. As a result, the redesigned system would most likely have to cope with the following kinds of change:

- The metadata (*information used principally in aiding users to access knowledge objects of interest* used needs to be extensible, as further concepts and relationships will be incorporated in the future.
- The taxonomy developed needs to be understandable to everyone, from experts to novice users.
- New knowledge will be added to the knowledge base by the knowledge engineers every fortnight, thus this proportionally frequent knowledge growth needs to be catered for.
- The taxonomy structure needs to be able to change, especially when a knowledge audit is done to decommission outdated knowledge.
- The system needs to be refined and maintained by the knowledge manager in order to prevent system overload and to grant rights to the various knowledge engineers.

### 6.2 Conceptual Modelling and Knowledge Base Construction

Based on the requirements gathered during a series of interviews with the target population, within the sample for this research, an initial glossary of terms, a blueprint document and a project plan were developed. In an attempt to derive a set of concepts, sample data was obtained from the current knowledge base. WinMine [5] was used as a data-mining tool in discovering patterns and relationships between concepts from the sample data. The WinMine Toolkit is a set of tools that can run on Windows 2000/NT/XP platforms. It allows a user to build statistical models from data. The majority of these tools are command-line executables that can be run in scripts. WinMine [5] discovered about 15% of the concept relationships whereas the remaining 85% were added manually. These derived relationships were the basis for the construction of a simple Bayesian Network for a cause-effect relationship. For example, concepts associated with printing, including broad definition of printing problems, were derived. This led to development of the troubleshooting ontology.

**Figure 7: Concept extraction using WinMine**

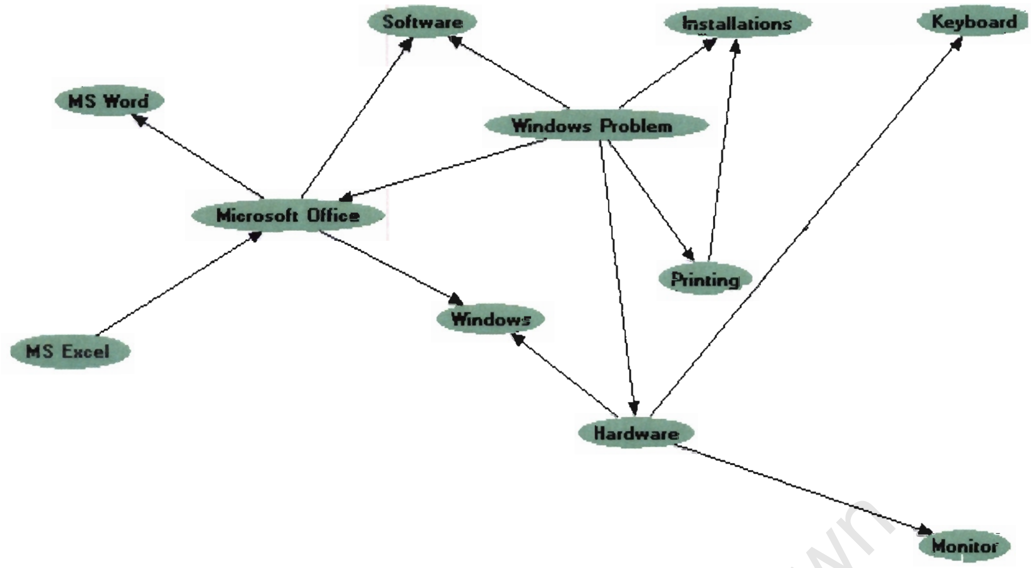



Figure 7 illustrates concepts extracted by WinMine from sample data obtained from ICTS. From the diagram, windows problems affect software, installations, printing, hardware as well as Microsoft office. Microsoft office has a dependency on MS Word, MS Excel, Software and windows. Hardware is associated with the monitor, keyboard and windows.

The fundamental steps taken at this level included determining the domain and scope of the ontology, enumerating important terms in it, defining the classes and the class hierarchy, defining the properties of classes, defining the facets of the slots as well as creating instances of classes. Representing problems, symptoms and solutions in a linear fashion was the domain of the ontology, and thus facilitated the development of an online troubleshooter, which suggests the right solution to a computer related problem. Concepts describing different types of printing problems, their solutions as well as good combination of two or more solution for the given task figured in this ontology. It also considered the level of detail of the given information in the knowledge base. Important terms in the ontology were enumerated so that statements in further explanation could be made or given to the user. The properties for these terms were given, for example, causes, symptoms and solutions.

In formalizing the concepts in a software environment, Protégé was used as the ontology management tool for modelling concepts derived from the sample data. Protégé can be used as a tool allowing users to construct a domain ontology. It can also be used as a platform that can be extended with graphical widgets for tables, diagrams, animation components to access other embedded applications in knowledge-based systems. It can also be used as a library, which other applications can use to access and display knowledge bases. With this tool, therefore, slots of concepts as well as role restrictions were determined. Taxonomies that organize concepts into sub- and super-concept tree structures were used to evaluate the relationships between concepts.

A top-down approach for defining classes and the class hierarchy was used. In this approach, the development process started with the definition of the most general concepts in the domain, and continued with subsequent specialization of the concepts. Classes were defined and their internal structure was ascertained. Slots and slot values were defined as well.

**Figure 8: Troubleshooting ontology package**

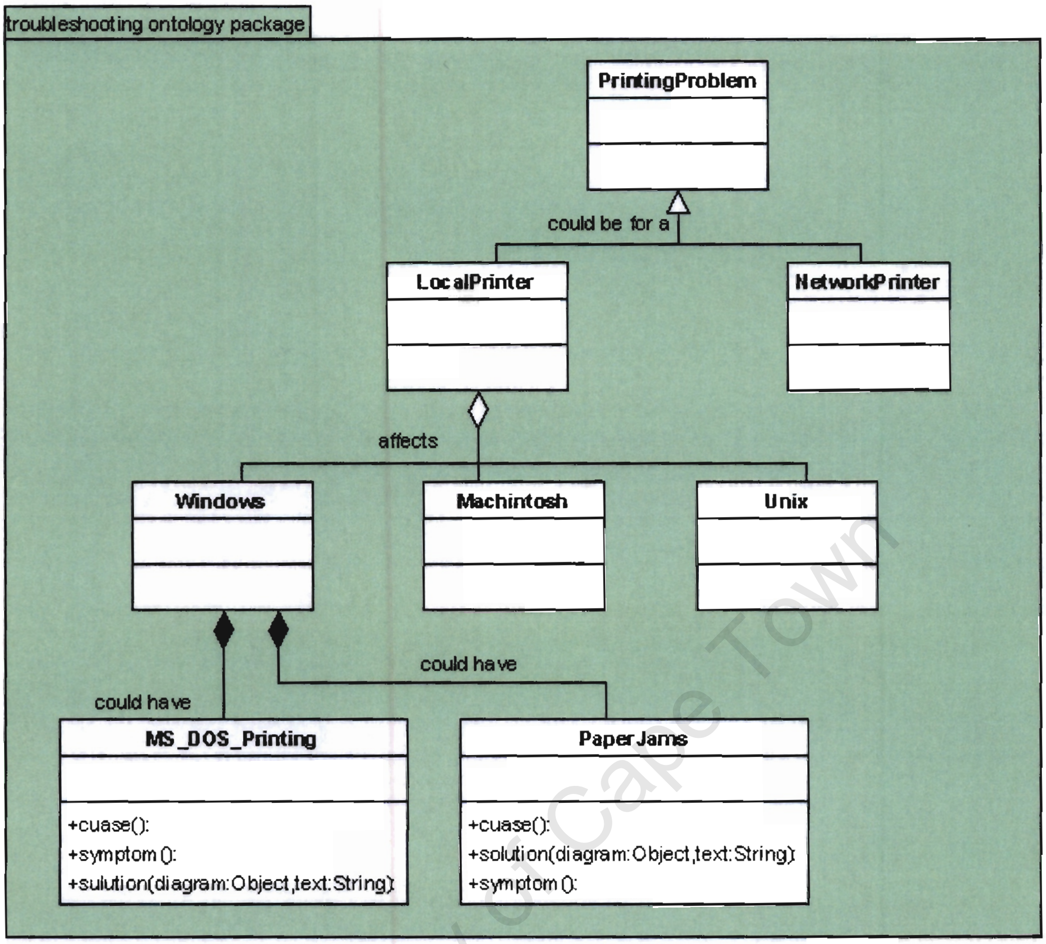



Figure 8 illustrates concept development in the online troubleshooter ontology using UML. The model, for example, represents PrintingProblem as the main concept. Printing problems could occur for either a local printer or a network printer. A local printer problem affects Windows, Macintosh and UNIX operating systems respectively. Windows could have MS_DOS_Printing problems, PaperJams etc. PaperJams comprises the slots of cause, symptom and solution. Solution has a restriction of either a diagram or a string.

**Figure 9: Protégé Ontology Representation**

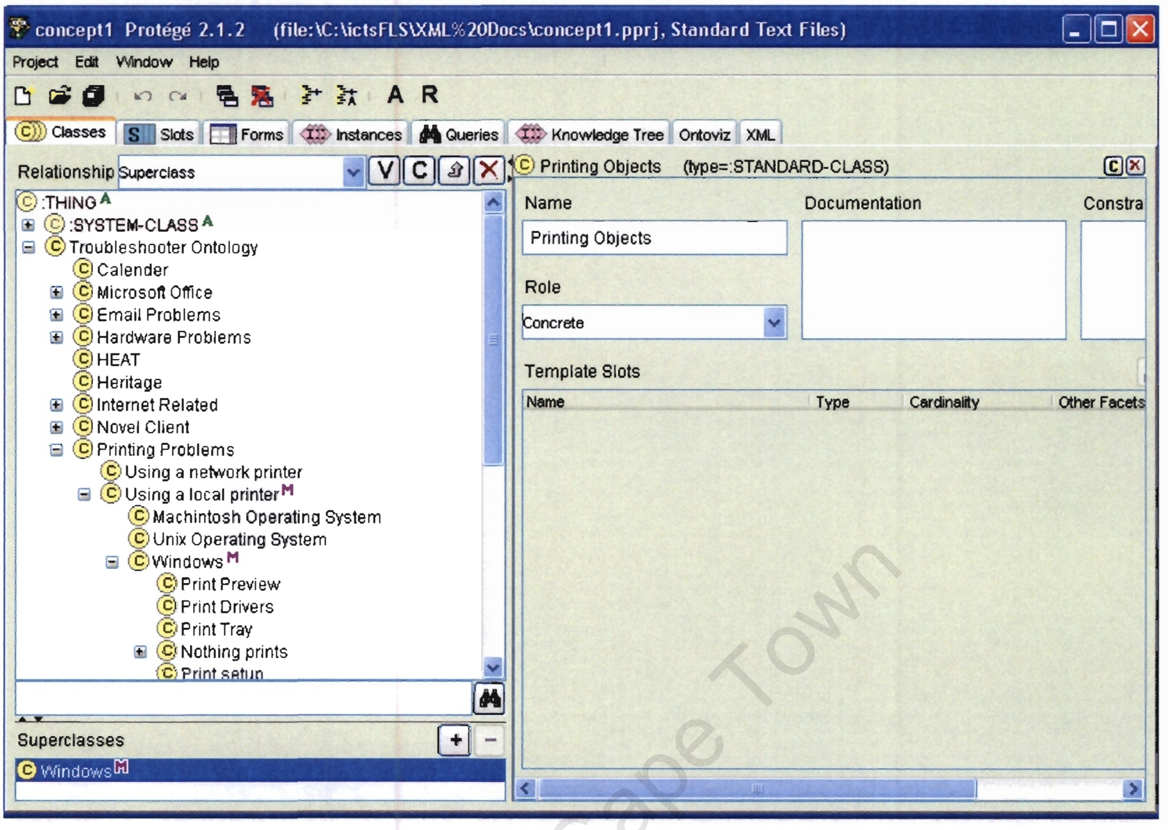
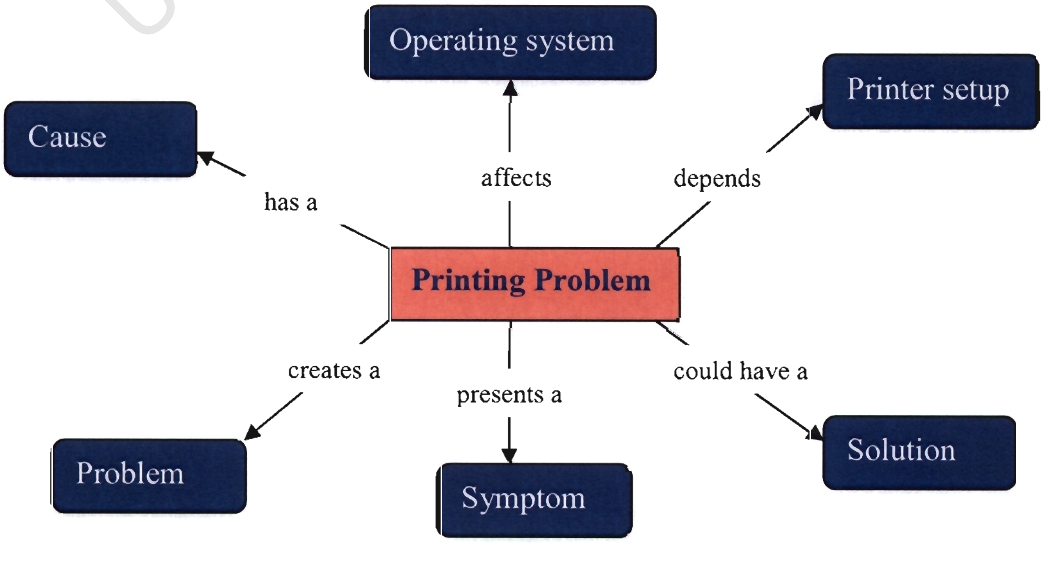



Figure 9 above illustrates the hierarchical knowledge representation in the ontology using Protégé.

**Figure 10: Graphical Frame Visualization**



Knowledge was visualized as a two-dimensional slice using the graphical frame visualization technique as illustrated in Figure 10. This was adopted to ease navigation within the taxonomy structure. The central concept or focal point chosen is the printing problem. The related concepts and their explicitly named relations are shown radiating outward from the centre. Thus improving knowledge visualisation in the model. The graphical frame was used to represent one concept out of many in the troubleshooter ontology. The whole knowledge structure in the knowledge base, however, formed a complex three-dimensional knowledge structure, which is illustrated in Figure 11. Ontoviz, a plug-in of Protégé, of the Graphvis visualization tool was used here. This was done to present a meaningful visualization of the underlying knowledge structure, while providing an intuitive understanding, eliminating visual clutter and maintaining aesthetic appeal. Concepts in the troubleshooting ontology are connected, regardless of how far away they may be from each other. For instance, a faulty printer cable, which is an instance of a hardware concept, can cause printing problems. It has multiple inheritances of symptoms in the ontology. Reinstalling Microsoft Office 2000, which is one solution often suggested by technicians at the ICTS helpdesk, solves many problems, including printing problems from applications. All of the above explain the need for the intertwined knowledge structure of the concepts within the ontology.

**Figure 11: Complex Three-dimensional Knowledge Structure**

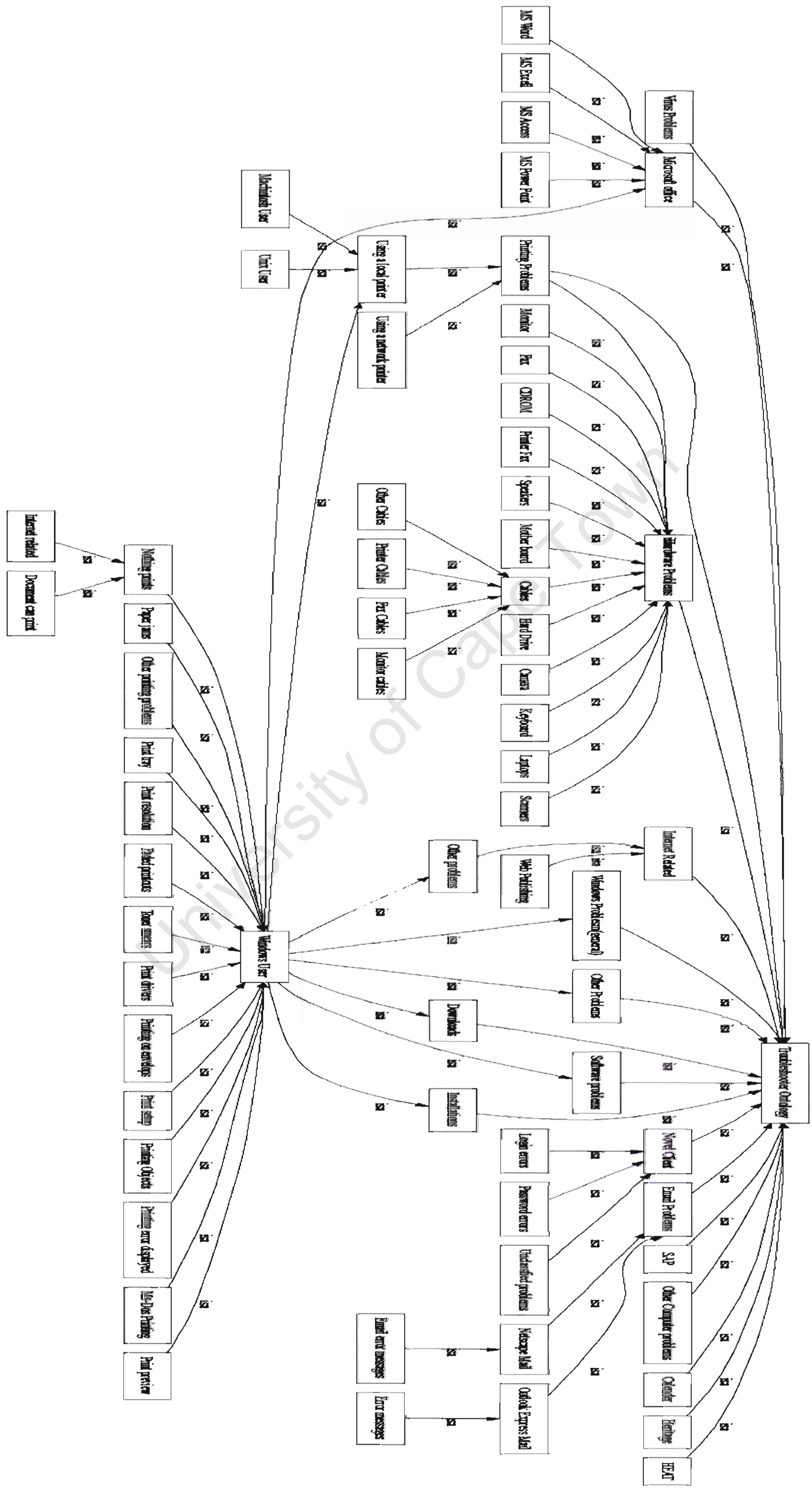


**Figure 12: Knowledge Modelling using Ontoviz**

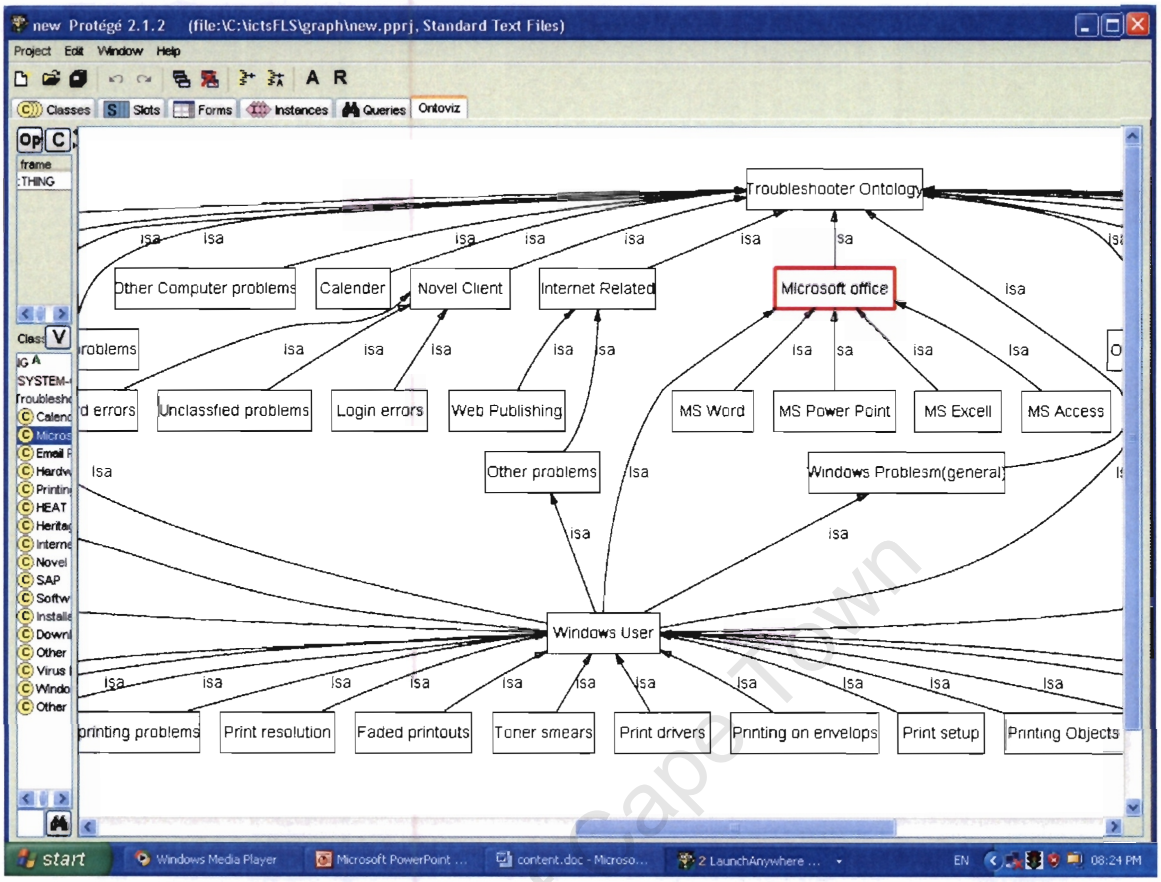



Figure 12 above is a closer look at the same graph shown in Figure 10, in Ontoviz a Protégé plug-in. This kind of visualization is important and useful to knowledge auditors in the system. It clearly and graphically illustrates the interrelationships of knowledge items in the knowledge base. Deleting a single knowledge item from the knowledge base may have cascading effects in that it could create knowledge islands that are no longer referenced by other items, which may wreak havoc in the system. However, this point exactly might be very important to the knowledge base users. In such circumstances, other nodes related to the main root node could also be deleted.



## 6.3 Operationalization and Validation

Operationalization in the knowledge base was done using UML, WinMine and Protégé. Other online ontology libraries were referenced for indirect validation. Sample documents of information presentation standards for ICTS were referenced as well for direct validation purposes.

**Figure 13: Troubleshooter Components**

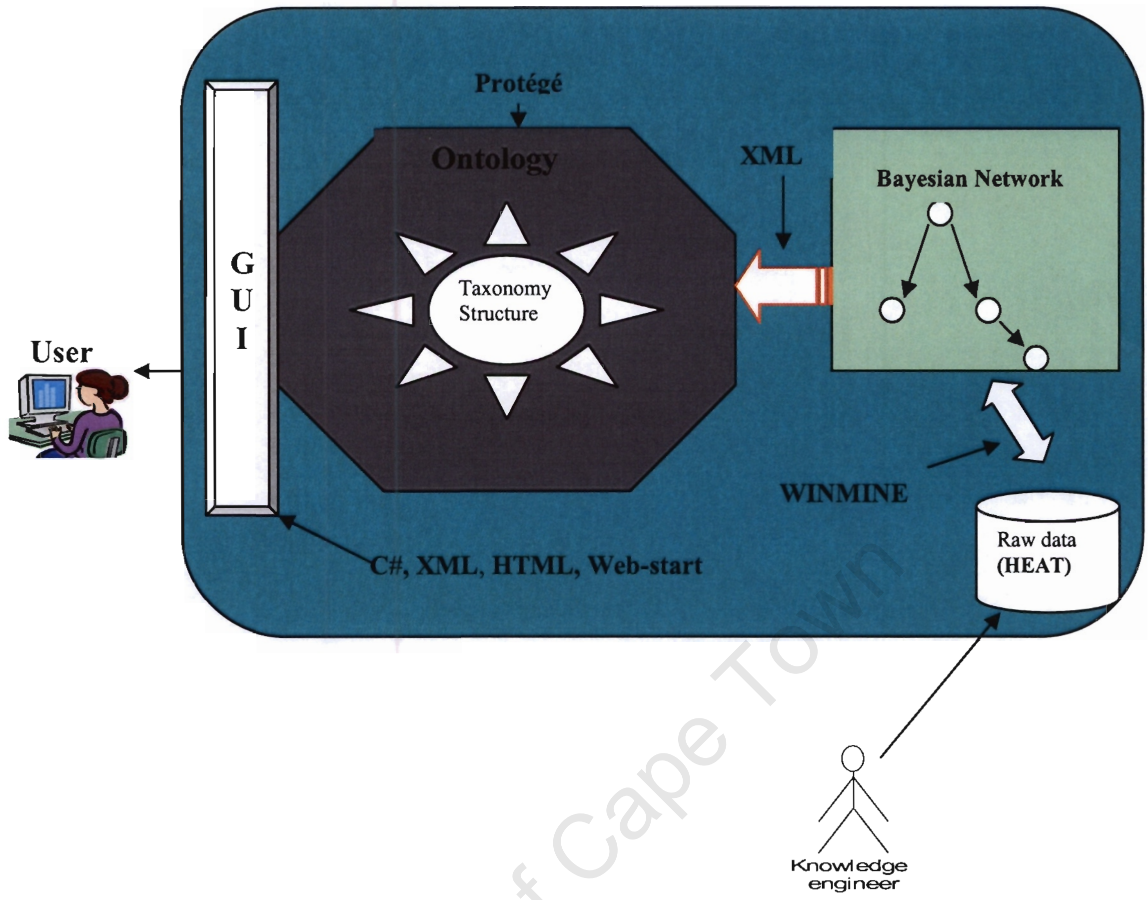
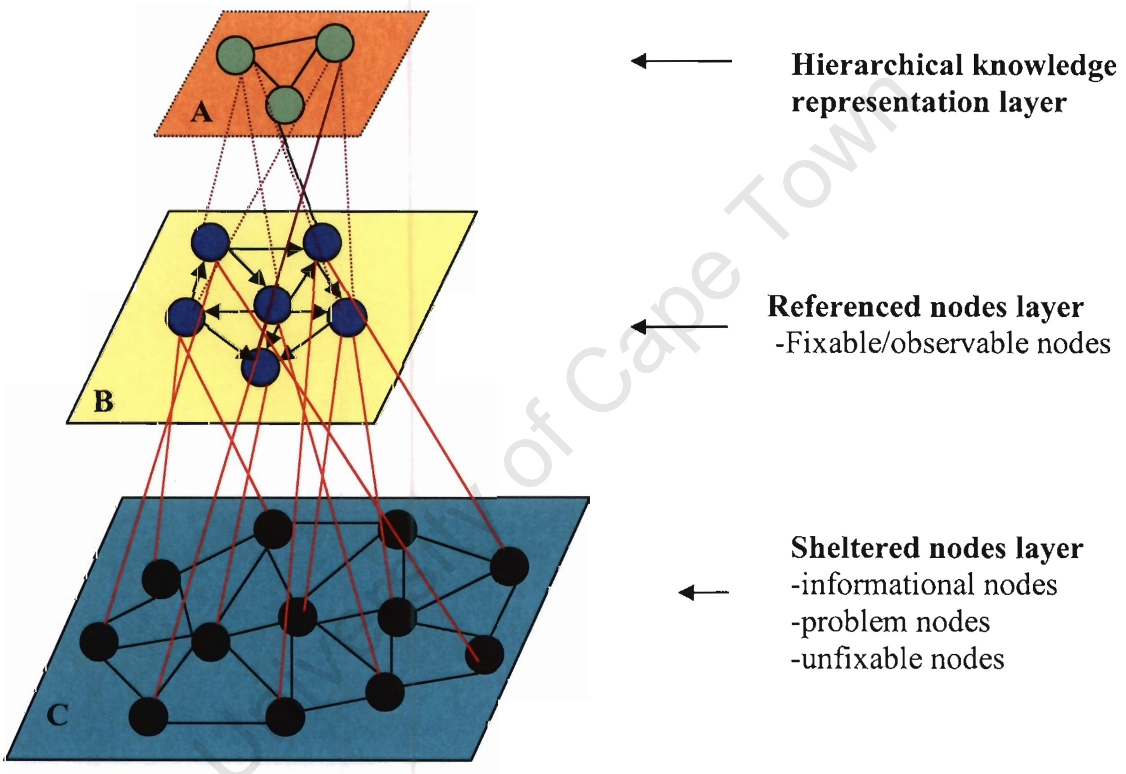



Figure 13 illustrates how the troubleshooter components are interrelated. The knowledge engineer interacts with the HEAT system by adding resources and auditing the knowledge. WinMine is used to mine patterns and relationships between the data. XML is used as an export language to transfer the relationships into Protégé, whose role is ontology management. Knowledge in the ontology is structured using taxonomies for easier navigation within the knowledge base. A graphical frame is implicitly used to present concepts to the user through the Graphical User Interface (GUI). The troubleshooter works in a linear sequence when solving computer related problems. It starts off with problems, and then finds the symptoms followed by the solutions.

Nodes containing common problems that users face while using computers are presented in a taxonomical structure. Concepts, representing these nodes, are arranged according to broad classification standards for ICTS. These nodes reference the symptoms that are associated with particular problems. Based on the symptom selected, a solution node that matches the selection is displayed to the user as illustrated in Figure 14.

A solution is presented in two formats, text and image. The image field is added to aid novice users who cannot follow text solutions only. The visual solution of what step to take in solving a problem is displayed. This enables the user to troubleshoot using both the image and the textual solution simultaneously.

**Figure 14: Knowledge layering in the prototype**






Layer A in Figure 14 above presents the hierarchical knowledge item nodes in the troubleshooting ontology. This is what the users see in the form of a navigation tree structure. Fixable/observable nodes (comprising the referenced nodes layer), which contain symptoms of given computer problems, are presented in the middle layer (B). Based on the kind of knowledge item selected in the first layer, corresponding symptom nodes are displayed to the user in the GUI. These symptom nodes reference the information nodes that contain appropriate resolutions. Layer C is abstracted away from the user. This layer contains the informational nodes, which have resolutions; it has the problem nodes, as well as the unfixable nodes. However, the user only needs to see the symptoms and the resolution, and does not need to see the cause for such a problem unless they know their way around computers and would like to sort out the cause so it doesn't happen again. Such nodes are only useful to the technicians. When interacting with the system, it is not easy for the user to differentiate between the various layers of knowledge representation; consequently, knowledge management is important for leading to knowledge filtering for better and relevant representation.

**Figure 15: Online Troubleshooter Prototype**

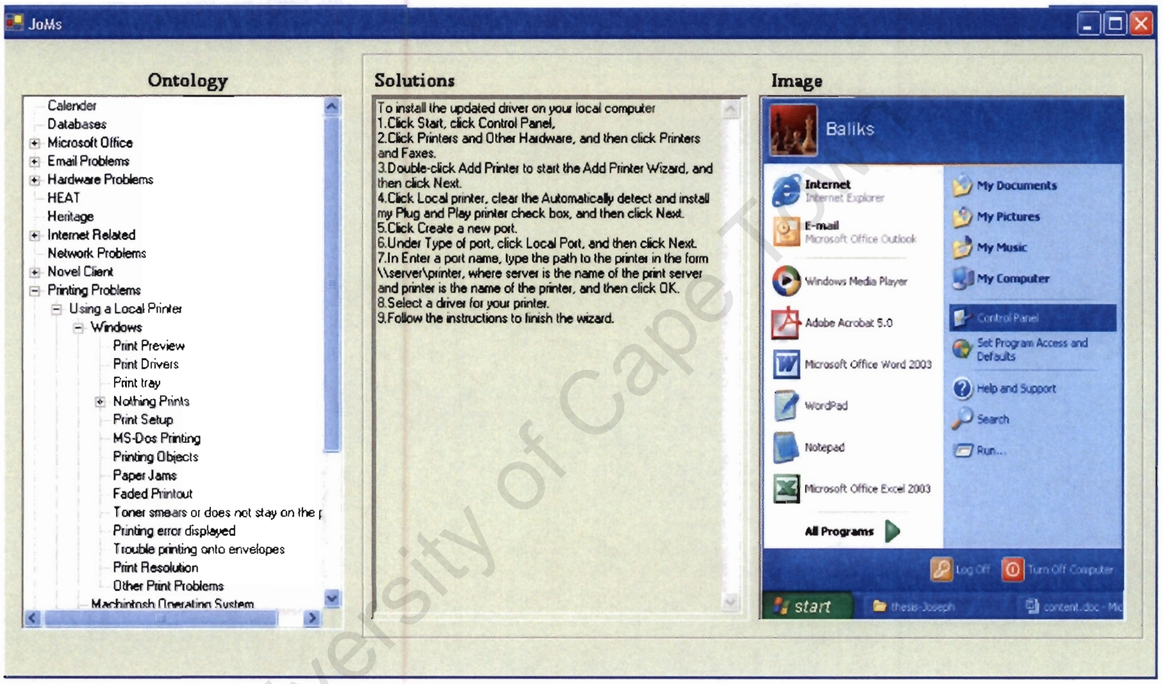


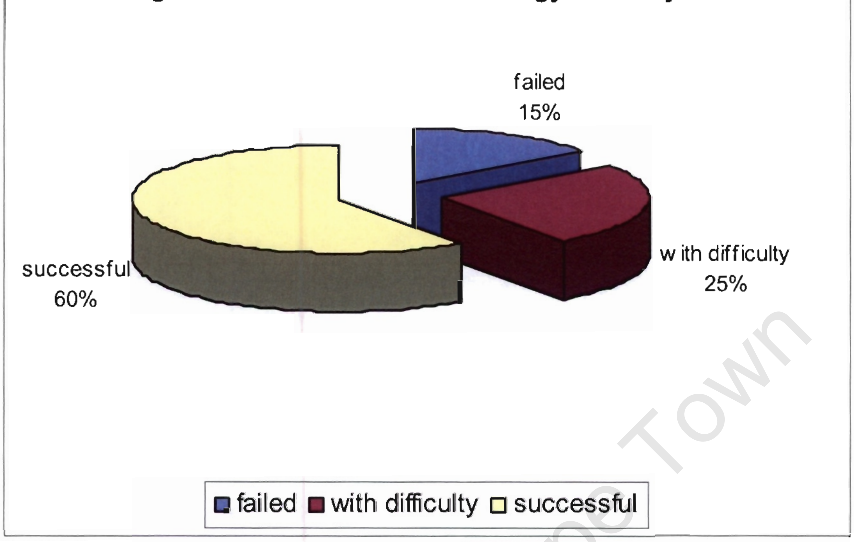



Figure 15 above illustrates the online troubleshooter prototype. It depicts a section of the troubleshooting ontology where knowledge is represented in a taxonomical structure. Based on the kind of computer problem identified in the taxonomy, key symptoms associated with that problem are displayed in the middle panel of the prototype. The user is given the choice to select a particular symptom associated with his/her problem. Sequentially, the system displays the solution both textually and diagrammatically. The images are meant to be dynamic, showing each particular step that a user will be going through while troubleshooting. For example, the given image in the screen shot, describes the first instruction to the user to click start and control panel.

## 6.4 Usability Test Results

Twenty respondents were selected to test the usability feature of the prototype. The sampling was random and non-probabilistic from the ICTS clientele, as illustrated in Figure 16.

**Figure 16:Troubleshooter ontology usability test**




The results revealed that 15% of the respondents failed to execute the tasks assigned to them in the experiment; 25% were able to execute tasks assigned to them but with great difficulty; but a clear majority of 60% were able to execute the tasks assigned to them in the experiment without any help. For the purposes of this study, those who failed felt that the software was not working. These were regarded as novice users of computer systems in this study. Those who had difficulties but nonetheless succeeded were considered semi-expert users, whereas the remainder, those who were successful, were considered expert computer users.

Novice users and semi-expert users would therefore require a query tool on the ontology in order to be able to locate information from the taxonomy structure. In improving visualization and adding meaning to solutions presented, dynamic images solutions were rated very helpful by both the novice and semi-expert users; one respondent observed: *"Images/pictures explain better than written text."*

In order to improve the system, therefore, more dynamic image solutions need to accompany the text solutions in the troubleshooter ontology. This will cater for users who have difficulty in locating information and understanding the computer language used. It would be much easier for such a user to follow steps of trouble shooting if a visual aid was available.

## 6.5 Comparison Test between Applications

In addition to the usability test conducted on the prototype system, a comparison experiment was done between the existing system and the prototype. This experiment targeted the clientele that was already using the ICTS knowledge base. Tests centred on three main aspects, that is, clarity of language used, navigability and user-friendliness.

**Figure 17: Comparison test between applications**

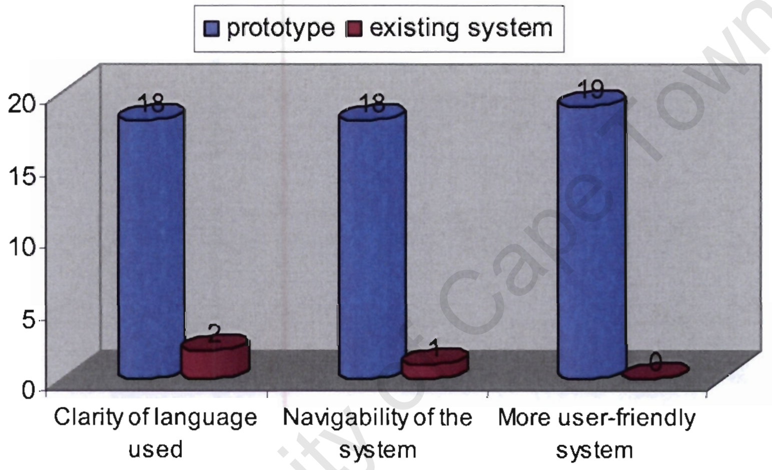





Figure 17 illustrates the findings from these three areas. Twenty people were sampled in this experiment, and the same kinds of task were given for both applications.

### 6.5.1 Clarity of Language Used

In this comparison test, 90 % (18) of the respondents found the language used in the prototype to be non-technical, clear and easy to understand. Although the language used in the existing system was more technical, 10% (2) of the respondents felt that the language used portrayed the same meaning. To these respondents, there was no clear-cut distinction between the two systems as far as the technicality of the language was concerned.

### 6.5.2 Navigability of the System

95 % (19) of the respondents found the prototype to be more navigable than the existing system. The hierarchical knowledge representation used in the prototype made the ontology navigable, and thus respondents commented.

- *"Easier to navigate and the image panel helps for clarity."*
- *"The prototype gives more information and the steps are well presented and the composition of the ontology, solution and the windows interface. The ontology is a bit complex, but provides adequate information. The tree structure is better as used in the prototype than just using a long list on the other system. The representation of all the interfaces in one window reduces navigability problems. In this regard, the prototype is well represented and easier to use."*

Only 5 % (1) of the respondents considered the navigation structure to be the same across both applications.

### 6.5.3 User-friendliness of the System

95% (19) of the respondents in the study considered the prototype to be more user-friendly than the existing system. They said that the image accompanying the textual resolution of a given computer problem added meaning and clarity. Various comments were made by respondents with regard to usability of both systems:

- *"The prototype is user-friendly. It gives the solution that you can understand."*
- *"The prototype is compact, all together-3 screen option, drop menus ensures a logical process, and modernization is successful. ICTS knowledge base: I don't like the color scheme used and the print font. It is old fashioned and not as user friendly."*
- *"The prototype gave the image and was easier to follow."*
- *"The notion of the "tree" is better in the prototype. There is an image supporting the textual description."*
- *"The prototype is better because it gives a pictorial view of the solution so that the user can see the text."*

Only 5 % (1) of the respondents were indifferent with regard to the user-friendliness of the system, as can be observed in the comment made.

- *"Both are unfriendly but the prototype is marginally better."*

Considering the fact that 95% percentage of the respondents considered the system to be usable, the prototype is worth developing into a full-fledged system for general usage by ICTS. Enhancing the prototype with a query tool and improved knowledge visualization will render the system user-friendly to cater for all categories of users, from novice to expert level. Although dynamic images increase overload on the system, thorough database tuning and optimization will resolve this challenge.

## 6.6 Conclusions

Based on the findings from the study, the following conclusions were drawn.

- Knowledge modelling and representation can improve knowledge representation if knowledge engineering techniques are appropriately and carefully followed in designing knowledge base systems.
- The language used for knowledge representation plays a vital role in determining how best users can interpret knowledge items in a given knowledge domain.
- Poorly designed knowledge management systems cause frustration among the clientele that interacts with the system.
- Consequently, embracing the ontology technique can play a vital role in improving knowledge representation.
- The hierarchical knowledge representation of knowledge using taxonomies improves the navigability of the knowledge structure.
- Visual knowledge representation as a supplement to textual knowledge adds meaning for the user. This technique is so important and useful because it assists especially the novice users.



## 6.7 Recommendations

- **Bayesian Networks and Ontology Techniques:** There is a need to utilize data mining techniques so that patterns and relationships between concepts can be discovered. The new system needs to interface directly with the HEAT system. A Bayesian Network can be developed based on such discovered patterns and relationships. By directly translating the output from the inference engine into the ontology structure, knowledge representation can be improved.

- **Domain Expert:** There is a need to employ a domain expert to take final responsibility for knowledge auditing on the system. It will thus be his or her duty to control and check each new concept and the relationships ascertained in the knowledge base in the future, thus controlling the extensibility of the metadata used. This will also necessitate controlling knowledge representation as well.

## 6.8 Areas of Further Research

There is a need for further research on adequate knowledge filtering techniques using knowledge engineering mechanisms to improve knowledge representation in knowledge bases. It would also be interesting to do further research on Topic Maps and hypergraphs as excellent ways of knowledge representation where knowledge structures and their relationships are visualized.

In order for knowledge to be meaningful to all users in any knowledge management system, it has to be filtered and presented well for easy and better interpretation. Otherwise, it may not serve its purpose if its intended audience does not understand it easily. The technique of knowledge modelling and representation therefore, plays a vital role in enhancing knowledge management.

## REFERENCES


[1] Angros, Richard Jr, W.Lewis Johnspn, Jeff Rickel, Andrew Scholer. Learning domain knowledge for teaching procedural skills. *In Proceedings of the first international joint conference on Autonomous agents and multiagent systems*: part 3. (Bologna Italy), ACM Press, New York, NY, 2002, 1372-1378

[2] Attar Software. Capturing and using knowledge: *A white paper*. 2002; http://www.intellicrafters.com/capture.pdf

[3] Benjamins, R.V, Lopez M.J, Contreras J, Casillas J et al. Skills management in knowledge-intensive organizations. *In Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web.* Springer-Verlag London, UK 2002, 80-95

[4] Brézillon P. and Pomerol J.-Ch. Is context a kind of collective tacit knowledge? *European CSCW 2001 Workshop on Managing Tacit Knowledge*. (Bonn, Germany). M. Jacovi and A. Ribak (Eds.), 2001,23-29

[5] Chickering David Maxwell. *The WinMine Toolkit*, Microsoft, MSR-TR-2002-103, 2002, Redmond, WA.

[6] Dennis, Alan R., Hayes, Glenda S., Daniels, Robert M. Jr. Business process modeling with group support systems. *Journal of Management Information Systems*. 1999; Spring: 115-142.

*[7]* Delphi Group. *Taxonomy & content classification: Market milestone Report,* A Delphi Group white paper. Autonomy, Boston. 2002.

[8] Edgington Theresa. Adopting ontology to facilitate knowledge sharing. *Communication of the ACM.* 2004; 47(11):85-90.

[9] Ginsberg, Matt. *Essentials of artificial intelligence*: Morgan Kaufmann. San Francisco, CA, USA 1994.

[10] GodBout Alain J., GodBout Matine Godbout. Filtering knowledge: changing information into knowledge assets. *Journal of Systematic Management.*1:(Jan 1999); 1998-1999

[11] Gruber, T.R. A translation approach to portable ontology specification. *Journal on Knowledge Acquisition*. 1993; 5(2):199-220.

[12] Hariharan, Arun. Knowledge management: A strategic tool. *Journal of Knowledge Management Practice.* December 2002; http://www.tlainc.com/artic146.htm [Accessed on 24th March 2005]

[13] Heckerman, David. Bayesian networks for data mining and knowledge discovery: *IEEE Intelligent Systems*. 1997: 79-119

[14] Harmno Paul and Brian Sawyer. *UML for Visual Basic 6.0 using visual modeler and Rational Rose 98.* Morgan Kaufmann, California. 1999.

[15] Jonathan R. *Knowledge management best practices for customer service and support*. 2003 http://www.serviceware.com/pdf/whitepaper-kmbestpractices.pdf [Accessed on 9th February 2005]

[16] Kidd, A. *In preface to Knowledqe Acquisition for Expert* (A. Systems, (Kidd ed.), 1987.

[17] Lave J. and Wenger E. *Situated learning: legitimate peripheral participation, Cambridge*. Cambridge University Press. 1991.

[18] Mullins, Craig S. What is knowledge and can it be managed?: *The Data Administration Newsletter.* Platinum Technologies Inc. 2003.

[19] Musen, M.A. Dimensions of knowledge sharing and reuse. *Computers and Biomedical Research*. 1992; 25: 435-467

[20] Nickols Fred. *Communities of practice: Definition, indicators & identifying characteristics*. 2000 http://home.att.net/~discon/KM/CoPCharacteristics.htm [Accessed on 9th February 2005]

[21] Noy, Natalya F and McGuinness, Deborah L. *Ontology development 101: A guide to creating your first ontology.* Stanford University, Stanford. 2000

[22] O'Leary, D. Knowledge management: An analysis based on knowledge use and reuse: *IEEE Intelligent Systems*. 2001;16(1)

[23] OMG. *What is OMG-UML and why is it important?* Object Management Group. USA. 1997.

[24] Preece, A. et al. Better knowledge management through knowledge engineering. *IEEE Intelligent Systems Journal.*2001:36-43

[25] Probasco Leslee. The ten essentials of RUP — the essence of an effective development process. *Rational Software white paper*. 2002

[26] Protégé. *The Protégé Project*. 2000. http://protege.standford.edu

[Accessed on 9th February 2005]

[27] Sacket DL et al. Evidence-based medicine: what it is and what it isn't. *BMJ* 1996; 312:71-72.

[28] ServiceWare. Knowledge management: the key to customer service success- why knowledge enabling the contact center is mission critical. *A ServiceWare white paper*. 2002.

[29] Smith, John. The estimation of effort based on use cases rational. *Software white paper*. 2002

[30] Staab, Steffen Rudi Studer, Hans-Peter Schnurr, York Sure. Knowledge processing and ontologies. *IEEE Intelligent Systems* 2001;16(1):26- 34

[31] Stevens Robert. *What is an ontology*? 2001 http://www.cs.man.ac.uk/~stevensr/onto/node4.html" [Accessed on 9th March 2005]

[32] Storey, Margaret-Anne. *Information visualization and management*. Lecture Notes. 2003 http://www.cs.uvic.ca/~mstorey/teaching/infovis/course_notes/course_overview.pdf [Accessed on 4th February 2005]

[33] Swartout, W. and Tate, A. Guest editors' introduction: ontologies. *IEEE Intelligent Systems*. 1999; 14(1):18-19.

[34] Thearling, Kurt. An introduction to data mining: discovering hidden value in our data warehouse. *A White Paper*. 2004 http://www.thearling.com/text/dmwhite/dmwhite.htm

[35] Tobin, Tom. *Employee self-service: Benefits for the Helpdesk*. A Knova Software, Inc. 2003

[36] VNU Business Media. *Eight things that training and performance improvement professionals must know about knowledge management*. Press Release 2001: Tuesday, 7th August. http://www.destinationkm.com/print/default.asp?ArticleID=325 [Accessed on 4th February 2005]

[37] Webster's New Collegiate Dictionary: Springfield, MA: G. & C. Merriam Co., pp. 1975

[38] Zhu Zhenyu. Requirements determination and requirements structuring. *Information systems analysis review*. University of Missouri. 2003.

# APPENDIX 1

## DEVELOPING A SYSTEM VISION

1. What is the problem with the current system?
   ..................................................................................
   ..................................................................................
2. Who are the main stakeholders of the system (Managers)?
   ..................................................................................
   ..................................................................................
3. Who are the users of the system?
   ..................................................................................
   ..................................................................................
4. Is there any economic buyer for the system?
   ..................................................................................
   ..................................................................................
5. Who else will be affected by the outputs that the system produces?
   ..................................................................................
   ..................................................................................
6. Who will evaluate and bless the system when it is delivered and deployed?
   ..................................................................................
   ..................................................................................
7. Are there any other internal or external users of the system whose needs must be addressed?
   ..................................................................................
   ..................................................................................
8. Who will maintain the new system?
   ..................................................................................
   ..................................................................................
9. Is there anyone else?
   ..................................................................................
   ..................................................................................

# APPENDIX 2

## USER REQUIREMENTS QUESTIONNAIRE

This is a short questionnaire designed to achieve user satisfaction in designs. The objective of this survey is purely academic. All the data given will be treated with absolute confidentiality. Permission has been extended from ICTS for this survey to be carried out.

1) Department
   ..............................................................................

2) Status (Pass Staff, Academic, Undergraduate, Postgraduate, Contractor, etc)

   ....................................................................................
   ....................................................................................
   ......

3) Do you regard yourself as a technical/ non-technical person?
   ....................................................................................
   ....................................................................................

4) Do you use computers in your day-to-day activities?
   a. Yes
   b. No

   If yes, how often do you experience computer related problems?
   a. Very often
   b. Often
   c. Seldom
   d. Rarely.

5) How do you go about solving computer problems if at all you experience any?
   a. Call IT Helpdesk
   b. Ask a colleague
   c. Solve it yourself
   d. Use ICTS Web-based Knowledge Base
   e. Other method (please specify)

   ....................................................................................
   ....................................................................................

4. Have you used the ICTS Web-based Knowledge Base before?
   http://www.heat.its.uct.ac.za/fls/HEAT/2/1/00000010000.htm?n=UCT_IT_Helpdesk_Knowledge_base

   a) Yes,
   b) No

   If No, why not?
   ....................................................................................
   ....................................................................................

   If yes, were you able to locate the information you wanted?
   a) Yes
   b) No

5. What would you like to see changed (on this knowledge base) to enhance the usability of this function?

   ………………………………………………………………………………………

   ………………………………………………………………………………………

6. Please give any more suggestions you feel important for improving the ICTS web-based knowledge base.

   ………………………………………………………………………………………

   ………………………………………………………………………………………

**Thank you for your time and comments**

# APPENDIX 3

## USABILITY TESTING FOR THE TROUBLESHOOTER PROTOTYPE

## SECTION A

1. Task 1
   a) From the ontology, select **Printing error displayed.**
   b) Select **Printer is displaying a 79 error**
   c) Click submit button
      i. Are you able to see the solution to that printing problem?
         a) Yes b) No
      ii. Are you able to understand the computer language used to describe the solution?
2. Task 2
   a) Search for printing objects as a computer problem from the ontology.
   b) From the symptoms displayed, select **Objects** and **text not printed well**.
   c) Click submit
   d) Look at the solution given and the image.

      i. Is the image helpful in giving more meaning to the solution?
         a) Yes b) No
      ii. Would you wish to have an accompanying image to a given textual solution?
         a) Yes b) No
         If yes, why?

   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
3. Task 3
   a. Search for a solution describing how to fix a keyboard to a computer from the ontology.
   b. Briefly describe the navigation steps you used to arrive to the above mentioned task.

   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………

## SECTION B

1. Do you consider yourself to be a
   a. Computer expert
   b. Novice user (ordinary user)
   c. First time user
   d. I cannot rate myself
2. Were you happy with the way the text solutions were presented to you?
   a. Yes, please give reasons
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   b. No, please gives reasons
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
3. This software is
   a. Easy to use
   b. User friendly
   c. Difficult to use
   d. Non of the above
   e. Others (Please specify)
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
   ……………………………………………………………………………
4. Please give any other comments and suggestions
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………

***Thank you for your time.***

# APPENDIX 4

## COMPARISON TEST OF APPLICATIONS

## SECTION A

4. Task 1
   a) From printing problems in the ontology select **Print drives**
   b) Select **Installing print drivers**
   c) Click submit button
      i. Are you able to see the solution to that printing problem?
         a) Yes b) No
      ii. Are you able to understand the computer language used to describe the solution?
      iii. Does the accompanying image help you understand better the steps to follow while solving your problem?
         a) Yes b) No
5. Task2 : http://www.heat.its.uct.ac.za/fls/HEAT/2/1/00000010000.htm?n=UCT_IT_Helpdesk_Knowledge_base
   **e)** From Printing Problems on the ICTS Knowledge base, select **Local Printer**
   f) Select **print drivers**
   g) Look at the solution given
   h) What is your opinion of how the solution is presented?
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   i) Were you successful in accomplishing the task?
      a) Yes b) No
      i. Would you wish to have an accompanying image to this textual solution?
         a) Yes b) No
         If yes, why?
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………
   ………………………………………………………………………

## SECTION B

5. In your opinion, which application do you consider better and more user friendly?
    a. ICTS Knowledge Base
    b. The prototype
6. Please give reasons for your choice
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………
7. Please give any other comments and suggestions
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………
   ………………………………………………………………………………

***Thank you for your time.***