0 Tables

0.1 Table of Contents

0.2 Table of Figures

0.3 Table of Tables

0.4 Acknowledgements

I would like to thank my academic advisor, Assistant Professor Marinos Themistocleous, for his guidance and encouragement through the duration of my master thesis project. I wish to express my sincere gratitude to my supervisors Flora Malameteniou and Maria Halkidi for their invaluable assistance and feedback. I am very grateful to the members of the Digital Systems Department, fellow students from the MSc program and all my friends from the University of Piraeus who provided me with friendly and stimulating environment. It is difficult to explain how grateful I am to my parents for their wisdom and care made it possible for me to study.

1 Introduction

A system is only as good as the data within that system. An increasing amount of organizations are discovering this as they upgrade older legacy systems into Enterprise Resource Planning (ERP) and Customer Relations Management (CRM) systems. New technologies in today's world have played an enormous role in the complexity of information sharing. The additions of the Internet and revelations in interface design have resulted in companies being closely intertwined with each other and the consumer. With the simultaneous transfer of information constantly taking place, it is more important than ever for manufacturers to be able to transmit and display quality data.

A basic, but crucial component of data quality is that data must be correct and accessible among systems. If data is not correct, organizations end up with errors, duplicates and inconsistencies resulting in extra costs, rework, and wasted time, in addition to a loss of system credibility and system reliability. For these reasons and more various industries and governments continue to spend large amounts of money on data quality initiatives.

Another key element important to data quality is conformance to user requirements. Often data quality suffers because the data recipient isn't sure how to express the type of data they need. Each master data message to contain a reference to a specific data specification to which the message complies. The data specification is required to be available to the public and it must be possible to check that the message complies with the data specification automatically by software tools.

The second part is provenance. This part allows an organization to describe requirements for representation and exchange of information about provenance of data quality pairs. Provenance information may include the record of origination, transcription, abstraction validation, and transfer of ownership of data. Provenance is important because a data element value represents an observation at a specific moment in time; therefore, the ability

to track the origin in time of a data element value is an essential component of data quality. Another essential component is the owner of the data element. A data element is created as a result of a process owned by an organization and the ability to trace the organization to a data element at a specific point in time is important for data quality. If an organization requires data suppliers to use this part, the organization would be able to provide measures for timeliness as well as traceability which are two important measures of data quality.

Other anticipated parts include, Accuracy and Completeness. These parts will address many of the key components of data quality.

Participation by both data suppliers and data recipients will be necessary in order to succeed. Data recipients will immediately benefit from this because it allows that organization to define their requirements for information quality when purchasing another organization's data. Instead of just accepting data as is, organizations now have a way to enforce levels of information quality when transferring and buying other organization's data.

In the past a concern existed regarding whether or not suppliers would be willing to improve their data quality, but also share their data as well. Previously there wasn't really an incentive for a supplier to have quality data within their system.

The result of the Internet and new technologies has been a shift in attitude of suppliers; now they have realized that, by such a framework, they can implement a managed data integration solution by automating requests for master data from their customers (Grantner, 2010). This allows them to differentiate themselves from other suppliers and helps to form key long-term relationships. Additionally, as the Internet increases the amount of information available to the public, suppliers are looking to increase the visibility of their goods and services. One way in which they are doing this is by publishing their catalog and the specifications of their products, capabilities and services on their web sites. If suppliers can improve the descriptions of data on their web sites, data would be even more visible, searchable and usable.

1.1 Issues regarding distributed RDBMS systems, obsolete data and data entry

1.1.1 Mixed language characters

Is described as a string that contains characters typed using mixed keyboard language settings (e.g. O (latin keyboard) <> O (greek keyboard)). The above results in loss of information during searches or even completely wrong results. This issue heightens because of the fact that many end users type in greeklish.

1.1.2 Abbreviations/Partial strings

An abbreviation (from Latin brevis, meaning short), is described as shortened form of a word or phrase. Usually, it consists of a letter or group of letters taken from the word or phrase. For example, the word abbreviation can itself be represented by the abbreviation abbr., abbrv. or abbrev.

This issue is due to inconsistency of rules and standardization among the various systems involved. For example in one system the father name is three characters long (e.g. in a credit card system) resulting in loss of information, partial results during queries and potential wrong results.

1.1.3 Inconsistency of data format

This is classified as a design issue of the systems involved, lack of interoperability and legacy systems.

For example a date/time field can have YYYYMMDD format and another date/time field can have YYMMDD or DDMMYY.

The above lead to mistakes made by the end users of a system, who must constantly be on alert in order to interpret the presented information correctly.

1.1.4 End user mistakes

In every process that involves data entry there is a percentage of records that are mistakenly typed because of the human factor involved. The most common ones are end user mistakes through data entry, anagrams etc.

This issue is partially resolved through validation rules enforced by the system or even in number sequences a very common approach is the presence of check digit.

1.1.5 Non up-to-date data

Despite the obligation by the customers of the Bank to inform about any change regarding the status of any public issued or demographic changes most of them does not fulfill this obligation that is regarded as bureaucratic.

Such information includes ID Card Number, address etc. This leads inevitably to obsolete data and failure of the Bank to comply with the legislation.

1.1.6 Lack of validation rules vs. user friendly interface

This is also classified as a design issue of the involved system, which satisfied the information needs at a time but lacks scalability. (e.g. Phone numbers < 10 digits, 30th day of February etc.)

1.2 Aims and Objectives

All these distributed databases are in a “live environment” within the organization. It would not be acceptable or even wise to stop these services for any amount of time in order to integrate them. There is also a very high risk in the whole process of merging and integrating two or more DBs, of loosing or mixing up critical information.

The concept is to locate a customer in all breadth and width of his/her banking activities, products and services. To create a bridge and try to convert all the existing product-centric systems in customer-centric, thus unifying the knowledge for the customer.

The ideal condition would be the relationship between customer and products/services to be one to many, meaning each customer to be assigned only one id and under this unique id to assign all the products/service of this customer something that is the foundation of CRM systems.

This is not always what happens in the real banking world.

1.3 Recommended solutions/Methodology

Qualitative research explores attitudes, behavior and experiences through such methods as interviews or focus groups. It attempts to get an in-depth opinion from participants. As it is attitudes, behavior and experiences which are important, fewer people take part in the research, but the contact with these people tends to last a lot longer. Under the umbrella of qualitative research there are many different methodologies.

Quantitative research generates statistics through the use of large-scale survey research, using methods such as questionnaires or structured interviews. If a market researcher has stopped you on the streets, or you have filled in a questionnaire which has arrived through the post, this falls under the

umbrella of quantitative research. This type of research reaches many more people, but the contact with those people is much quicker than it is in qualitative research.

1.4 Thesis outline

In the first Chapter are described the scope and objectives of this thesis with a brief reference in the issues regarding distributed RDBMS systems. Also recommended solutions and the methodology involved. In Chapter 2 there is a thorough reference in the state of the art scientific research in the field of knowledge management and relevant fields In Chapter 3 is described the mathematical representation of Vector Space Model that is used as the main relevance indicator throughout this thesis. Chapter 4 describes the research methodology that was followed throughout the implementation of the project. Finally chapter 5 describes the architecture and interface of the proposed system and also some measurement indices that were used to verify the results.

2 Foundations of Knowledge Management

Knowledge Management (KM) comprises a range of strategies and practices used in an organization to identify, create, represent, distribute, and enable adoption of insights and experiences. Such insights and experiences comprise knowledge, either embodied in individuals or embedded in organizational processes or practice.

An established discipline since 1991 (Nonaka, 1991), KM includes courses taught in the fields of business administration, information systems, management, and library and information sciences (Alavi, et al., 1999). More recently, other fields have started contributing to KM research; these include information and media, computer science, public health, and public policy.

Many large companies and non-profit organizations have resources dedicated to internal KM efforts, often as a part of their "business strategy", "information technology", or "human resource management" departments (Addicott, et al., 2006). Several consulting companies also exist that provide strategy and advice regarding KM to these organizations.

Knowledge Management efforts typically focus on organizational objectives such as improved performance, competitive advantage, innovation, the sharing of lessons learned, integration and continuous improvement of the organization. KM efforts overlap with organizational learning, and may be distinguished from that by a greater focus on the management of knowledge as a strategic asset and a focus on encouraging the sharing of knowledge.

2.1 Data, information, knowledge, intelligence, and wisdom continuum

Data

i. Information, often in the form of facts or figures obtained from experiments or surveys, used as a basis for making calculations or drawing conclusions

ii. Information, for example, numbers, text, images, and sounds, in a form that is suitable for storage in or processing by a computer

Information

i. Definite knowledge acquired or supplied about something or somebody

ii. The collected facts and data about a particular subject

iii. The communication of facts and knowledge

iv. Computer data that has been organized and presented in a systematic fashion to clarify the underlying meaning

Knowledge

i. General awareness or possession of information, facts, ideas, truths, or principles

ii. Clear awareness or explicit information, for example, of a situation or fact

iii. All the information, facts, truths, and principles learned throughout time

iv. Familiarity or understanding gained through experience or study

Wisdom

i. The knowledge and experience needed to make sensible decisions and judgments, or the good sense shown by the decisions and judgments made

ii. Accumulated knowledge of life or in a particular sphere of activity that has been gained through experience

iii. An opinion that almost everyone seems to share or express

iv. Ancient teachings or sayings

Information consists of data, but data is not necessarily information. Also, wisdom is knowledge, which in turn is information, which in turn is data, but, for example, knowledge is not necessarily wisdom. So wisdom is a subset of knowledge, which is a subset of information, which is a subset of data.

The terms Data, Information, Knowledge, and Wisdom are sometimes presented in a form that suggests a scale.

Figure 1

However, in no sense do these four terms define some sort of linear equal-interval scale.

Form a business perspective, data constitutes one of the primary forms of information. It essentially consists of recordings of transactions or events which will be used for exchange between humans or even with machines. As such, data does not carry meaning unless one understands the context in which the data was gathered. A word, a number or a symbol can be used do describe a business result, inserted in a marriage contract or a graffiti on the wall. It is the context which gives it meaning, and this meaning makes it informative.

Information extends the concept of data in a broader context. As such it includes data but it also includes all the information a person comes in contact with as a member of a social organization in a given physical environment. Information like data is carried through symbols. These symbols have complex structures and rules. Information therefore comes in a variety of

forms such as writings, statements, statistics, diagrams or charts. Some information theorists insist on the concept of form as the differentiating factor and the essence of information.

Information becomes individual knowledge when it is accepted and retained by an individual as being a proper understanding of what is true (Lehrer, 1990) and a valid interpretation of the reality. Conversely, organizational or social knowledge exists when it is accepted by a consensus of a group of people. Common knowledge does not require necessarily to be shared by all members to exist; the fact that it is accepted amongst a group of informed persons can be considered a sufficient condition. This is also true of “public domain” knowledge. The fact that it is readily available in writing or published material does not entail that everybody should be knowledgeable about it to meet the condition of being "common knowledge".

2.2 Forms and sources of knowledge

Typically, organizational knowledge is categorized by different forms, for example, knowledge can be explicit or tacit. Data and information are examples of explicit knowledge, and they can be stored in the organization’s information systems. According to Kogut and Zander (Kogut, et al., 1992 p. 386), organizational information includes “facts, axiomatic propositions, and symbols”. Tacit knowledge is usually considered as individuals’ skills and experiences, although it can also be organizational. For example, Kogut and Zander (Kogut, et al., 1992) refer to organizational know-how as “higher-order organizing principles of how to coordinate groups and transfer knowledge” (p. 388). Furthermore, Spender (Spender, 1996) uses the term “collective knowledge,” a social type of knowledge that is “embedded in the firm’s routines, norms and culture” (p. 52). Tacit organizational knowledge is therefore embedded in structures and actions. A third form of knowledge is “potential knowledge,” as suggested by Ståhle and Grönroos (Ståhle, et al., 2000). It refers to new knowledge that is not yet available to the organization but may exist in intuition and weak signals.

As we can see, organizational knowledge is not the same as the sum of the employees' knowledge. Organizational knowledge is more like knowledge collectively stored, shared, and experienced. Although experts working in organizations have much knowledge in their heads, it does not mean that the whole organization is knowledgeable. Therefore, employees' knowledge should be transformed to organizational level in order for this knowledge to facilitate and improve change in the organization.

2.3 How Knowledge Management contributes to an Organization

Knowledge management can improve an organization's ability to achieve development results. In its most basic form, knowledge management is all about converting the available raw data into understandable information. This information is then placed in a reusable repository for the benefit of any future need based on similar kinds of experiences. Knowledge management contributes towards streamlining the ideas, problems, projects and deployment in light of organizational goals driving towards productivity.

Goldsmith, Morgan, & Ogg (Goldsmith, et al., 2003) suggest the idea "*of knowledge management is fundamentally flawed-it involves neither knowledge nor management and therefore cannot be expected to succeed*" (p. 39). Rather, they suggest that the real focus should be upon "the intellectual capital" that workers possess. This creates a wide misunderstanding of the purpose and context of sharing that intellectual capital. Far beyond facts stored in memories of individuals, groups, or computers, intellectual capital deals with applied expertise gained through understanding and experience. Effron continues suggesting by illustration that best practices for hiring new workers may not be knowledge or facts easily gathered and stored. Often, a talented human resources or other organizational leader may possess significant skills and insights not learnable via a book or computer file. He suggests that learning from such individuals can be an important learned and shared intellectual capital.

2.4 Theories and principles of knowledge management

While the field of Knowledge Management has long been studied by scholars of several disciplines, there remain significant challenges for the future. These challenges reside in both theoretical and conceptual studies as well as practice and application. Change will be omnipresent – requiring organizations to make incremental or continuous improvements, and breakthrough or “game-changing” advances. The question is: What are the contributions that Knowledge Management will make as a field of study and a relevant practice (Dierkes, et al., 2003).

According to Reinhardt, Bornemann, Pawlowsky and Schneider (Reinhardt, et al., 2003), "*With knowledge as one of the most important resources today, management obviously should attempt to identify, generate, deploy, and develop knowledge*" (p. 794). The concept of knowledge management and the degree to which its value is outpacing the tangible assets of companies has become an issue of concern for many organizations and managers. "*Human capital is seen as a company's total workforce and its knowledge about the business...It is seen as crucial for marshaling the company's assets, both tangible and intangible*" (Reinhardt, et al., 2003, p. 796).

The theoretical/conceptual challenge lies in the lack of common definition of Knowledge Management. There exists widespread variation in how scholars define it. Like the field of Leadership, there needs to be further study and dialogue on what defines Knowledge Management. It is only from that common understanding that the field itself will flourish rather than becoming a popular management fad.

2.5 Perspectives of knowledge management

According to Rumizen (Rumizen, 2002), "*knowledge management is a systematic process by which knowledge needed for an organization to succeed is created, captured, shared and leveraged.*" For this reason, knowledge management involves leadership establishing processes, also defined as activities or initiatives, to help organizations adapt to an ever changing environment. Successful knowledge management depends on processes that enhance individual and organizational ability, motivations, and opportunities to learn, gain knowledge, and perform in a manner that delivers positive business results. Organizational processes that focus on these three attributes will lead to an effective “management” of knowledge (Argote, et al., 2003)). Rewards and other motivational incentives are keys to the knowledge management process. Argote, et al. (2003) have noted that members of an organization are unlikely to share insights and ideas within the organization if they are not rewarded for the knowledge sharing. They point to the impact of social rewards as being just as important as monetary rewards. A strong social culture within an organization can promote the transfer of knowledge. Within the midst of this strong culture there is a development of a desire for social cohesion and genuine spirit of reciprocity. Argote, et al. (Argote, et al., 2003) point to a less altruistic and a more egocentric motivation for knowledge sharing within an organization with a strong social culture. Often the employee is willing to transfer knowledge in order to protect their own social standing. Demonstrating uncooperative behavior or attitudes will damage one’s reputation and so to afford this social and professional risk, knowledge sharing increases.

In the global and technological environment, the challenge exists to move from an organizational mindset that suggests that knowledge is for the few on the top echelon to an understanding that knowledge once held by the few is available to the masses. Goldsmith, Morgan, and Ogg (Goldsmith, et al., 2003) contend, "*The old days of "continous improvement" seem as leisurely as a picnic from the past. In this chaotic and complex twenty-first century, the pace of evolution has entered warp speed, and those who can't learn, adapt, and change from moment to moment simply won't survive*" (p. 54). The need to rethink the process of knowledge management even in mega-organizations

is of paramount importance. Goldsmith, et al. (Goldsmith, et al., 2003) further contend, "*We're trying to manage something-knowledge-that is inherently invisible, incapable of being quantified, and borne in relationships, not statistics*" (p. 56). The time to understand knowledge management from a multi-directional perspective has come. Goldsmith, et al. says, "*Our most important work is to pay serious attention to what we always want to ignore: the Italic text-human dimension*" (p. 57).

According to Nonaka (1998), "*Understanding knowledge creation as a process of making tacit knowledge explicit - a matter of metaphors, analogies, and models - has direct implications for how a company designs its organization and defines managerial roles and responsibilities within it*" (p. 36). Nanaka states that this is accomplished within Japanese companies through redundancy, "*the conscious overlapping of company information, business activities, and managerial responsibilities*" (p. 36). As a process, redundancy can become a medium that assists in the management of knowledge within an organization. Though to many western managers redundancy may conjure up mental images of "*unnecessary duplication and waste*" (p. 36), it can assist in the area of employee expectancy, alleviating unnecessary assumptions and confusion.

Creating opportunities for individuals to create, retain, and transfer knowledge can be managed through employee development processes. For example, placing individuals in situations where they can gain new experiences, or share learning from a prior experience will enable knowledge management. Many companies have processes to intentionally move personnel across the organization (across units, regions, functions, etc.) for the purpose of transferring knowledge as well as building learning capability and agility within the individuals.

Ability, while innate, can also be increased through effective training processes and experiences. Training in analogical reasoning, for example, will increase an individual's ability to transfer knowledge between tasks, assignments, or reporting units, thereby spreading knowledge further across the organization.

Recognition and reward processes and systems can also influence the knowledge management process. Members of an organization, who are recognized and rewarded for knowledge transfer are more likely to engage in such sharing of knowledge, especially if it is integrated into the performance management process and will influence their standing or reputation in a positive manner.

Drawing upon Wheatley's (Wheatley, 1999) reference to a system as "a set of processes that are made visible in temporary structures" (p. 23), we might deduce that organizational learning - as a system process, is manifested or made known by the visible temporary structures of behavioral patterns, rhythms, and relationships. In other words, the organization is a "living system" – one that uniquely takes form through "fundamentally similar conditions" that other organizations encounter: *"...self...shared meaning...[and] networks of relationships...[resulting in] information [that] is noticed, interpreted, [and] transform*" (Wheatley, et al., 1999 p. 81) into knowledge. Thus, according to Wheatley (2004), knowledge management cannot be proficiently processed independent of "*creative work that is meaningful, leaders that are trustworthy, and organizations that foster everyone's contribution and support by giving the staff time to think and reflect together*" (Goldsmith, et al., 2003 p. 63).

The sheer volume of information today also presents a process problem. Wheatley describes what creates enormous possibilities for KM, "*world wide web has created an environment that is transparent, volatile, sensitive to the least disturbance, and choked with rumors, misinformation, truths, and passions*" (Trompenaars, et al., 2004 p. 53). The list includes the belief that organizations are a machine, only materials and numbers are real, you can only manage what you can measure, and technology is the best solution. The efforts are ultimately an attempt to make knowledge manageable. Something one can keep track of, keep inventory of, and procure for sale to another who wants it. To manage something you must have some kind of an understanding of it and an ability to control it to some degree. This reasoning leads to the list mentioned above by Wheatley as well as similar lists made by other KM leaders.

Wheatley's list says that humans create knowledge, and it's natural to create and share that knowledge, everyone is a knowledge worker, and people choose to share their knowledge. Another process issue is attaining or gathering knowledge. That knowledge exists throughout any given organization, but the ability to inventory or tap into that knowledge is difficult. Wheatley writes that "*we must recognize that knowledge is everywhere in the organization, but we won't have access to it until, and only when, we create work that is meaningful, leaders that are trustworthy, and organizations that foster everyone's contribution and support by giving staff time to think and reflect together*" (Trompenaars, et al., 2004 p. 63).

Effron (Efron, et al., 2004), asserts that given the definition of knowledge as "*the fact or condition of knowing something with familiarity gained through experience or association*", it is "impossible to acquire "knowledge" without either experiencing something yourself or interacting with someone else who has" (p. 40). Knowledge Management is not synonymous with IT systems and processes. Rather knowledge resides in the experiences of people in different contexts. With regard to Knowledge Management, the aim of an organization is to work within business processes that create, and transfer knowledge throughout the organization. If knowledge is created and transferred via human experiences then these business processes must encompass an understanding of how people learn and transfer their knowledge; that is the business processes must emphasize person-to-person contact (Efron, et al., 2004).

Examples of business processes that will lead to effective knowledge management are:

- The setting of goals and objective – be realistic and recognize the limitations of data mining and information gathering. Make the increase of organizational knowledge a stated and specific goal for the all.
- Employee retention – HR processes should focus on what it takes to retain employees who hold key knowledge. Provide opportunities that are developmental, have purpose, and have a high impact on

business performance. Compensate such employees above typical market rates.

- Employee development processes – pairing experts and apprentices provide opportunities for employees with differing levels of knowledge to work together and increase the organizational knowledge. These relationships allow for a true exchange of knowledge through a human relationship and experience.
- Organized networking and annual conferences – these provide forums for face-to-face interaction and knowledge sharing and can lead to effective organizational knowledge management.
- Accountability – line management, not just IT or HR, should be held accountable for knowledge management. They should be held accountable for management of the human resources and organizational knowledge. They do this through the above business processes of employee development (experiences, developmental assignments, etc.).

In the process of KM there must be significant steps taken to eliminate any barriers that may get in the way of becoming or increasing the ability to be a learning organization. Cummings challenged our intentionality for to effectively help the processes of KM within an organization there must be intentional efforts to remove barriers that would inhibit ideas, talent, and money from getting to the point of best use (Trompenaars, et al., 2004).

Managers and leaders play in important role in the success of knowledge management in their organization. James Robertson (Robertson, 2005) introduces ten key principles to ensure that information management activities are effective and successful. These focus on the organizational and cultural changes required to drive improvements forward. Those principles are:

- Recognize (and manage) complexity
- Focus on adoption
- Deliver tangible & visible benefits
- Prioritize according to business needs
- Take a journey of a thousand steps

- Provide strong leadership
- Mitigate risks
- Communicate extensively
- Aim to deliver a seamless user experience
- Choose the first project very carefully

The practical value of KM is in what it is able to impact, how it impacts, and how well it impacts. The line between KM and business is through the processes of business. KM's biggest impact on business may be in its ability to improve processes and their performance (Nichols, et al., 2000). It is suggested that the changing of processes should take into consideration the role KM plays in this process. In turn, the information that is needed to make decisions to make changes must be identified and well as determining the effects those decisions will generate.

An organization that wishes to begin to use Knowledge Management must begin by specifying specific processes. These processes must be supported by technological resources and must facilitate the sharing of information about problems and solutions, improvement suggestions and information concerning best practices practiced by other organizations. Organizations that follow this plan will develop a framework that catalogues, uses and integrates the knowledge used by individuals as organizational knowledge for driving innovation and organizational change (Hyde, et al., 2000).

Hyde and Mitchell (Hyde, et al., 2000 p. 57) offer six strategies for developing knowledge management processes within organizations:

1. Define a KM business case. What levels of knowledge and innovation will your agency need to stay ahead of your "environment" and be "competitive?"
2. Baseline your intellectual capital. Knowledge is an intangible asset, but human capital is not--measure current and projected workforce capabilities, your HR investments, and expected return on investment. (Get HR involved from the outset.)

3. Collaboration and knowledge sharing begin at the top, not at the bottom. Top management has to see how KM will affect performance and why it is critical for innovation and change.
4. Build KM from the bottom up and across. What is most important about any KM program or process is its ability to facilitate knowledge exchange among those individuals closest to the work, to the customers, and to the processes. KM must be an enabling process that captures both best practices and new ideas while promoting access.
5. Balance external and internal. The value of your KM program is multiplied by its reach--it needs to connect to other agencies, customers, and stakeholders. (Think in terms of strategic alliances.)
6. Think technology last. What products will you need to support your first level of KM development (allocate 75% of your KM IT budget). Save 25% for building your technology strategy to support future KM phases or new investments.

Andrews and Delahaye (Andrews, et al., 2000) found that factors at the individual level greatly influence knowledge processes. These included a person's perceptions of approachability, credibility and trustworthiness, which directly influenced knowledge importing and knowledge sharing. Researchers discovered that scientists in a bio-medical consortium actively filtered knowledge importing by deciding whom they would ask for information, who they would allow to give them input, and with whom they would share their own knowledge. They made decisions based upon what they felt their co-workers would do with the sensitive information. In each case the scientists made a judgment of co-workers as to their perceived trustworthiness.

Knowledge management's importance in organizations affects their competitiveness and the bottom line in significant ways. Ogg and Cummings suggest, "*There are three important things that can be leveraged in large companies to help take advantage of being a big organization, money, talent, and ideas*" (Goldsmith, et al., 2003 p. 103). Managing knowledge and intellectual capital increasingly grows as the critical of these three components that organizations need to align and use as leverage to foster

improvement from within against stiffening competition. The processes necessary to align and create increased leverage against the competition. Larger organizations can struggle to overcome significant barriers to discover, organize, and utilize what Ogg and Cummings call a marketplace of ideas (cf. p. 104). Overcoming barriers and hindrances to sharing and utilizing great ideas takes discipline and cultural values in which new ideas are readily shared, honored, and implemented.

Ogg and Cummings further suggest that fostering an organizational culture that values new ideas necessitates that meetings become places where ideas are shared, appreciated, and implemented in timely fashion. Additionally, infrastructure must connect people in trust relationship with a context where meaningful ideas are shared. Technology and data storage are inadequate to facilitate this kind of transference of new ideas.

2.6 Knowledge management life cycle

Garvin (Garvin, 1993) points to five building blocks that reflect some solid challenges to knowledge management:

1. Systematic problem solving.
2. Experimentation with new approaches.
3. Learning from one's own experience and past history.
4. Learning from the experiences and best practices of others.
5. Transferring knowledge quickly and efficiently throughout the organization.

These five building blocks need to function in harmony and balance with one another. Effective knowledge management can be increased as systems and procedures are developed to address and improve each of these five foundational stones. The challenge facing the organization comes in maintaining the dynamic nature of the interrelationship of these five areas of knowledge management. Garvin (1993) supplies three suggestions for addressing the first building block of systematic problem solving. First is

reliance on the scientific method, hypothesis testing, rather than on guessing when it comes to problem solving. Second, decision making should be based on data, not assumptions (fact-based management). And third, use simple statistical tools (charts, diagrams) to organize and communicate data.

2.7 Organizational enablers for sharing and managing knowledge: management, information and technology

Dierkes, Antal, Child, & Nonaka (2003) state, "*If knowledge is an essential resource for establishing competitive advantage, then management obviously should attempt to identify, generate, deploy, and develop knowledge. Hence, managers need more knowledge about knowledge and about how it can be managed, if it can be managed at all*" (Dierkes, et al., 2003 p. 794). In a world replete with knowledge and information (often similar in meaning), or its possible acquisition, what is often missing within organizations are the processes for dissemination. As with most things, knowledge is only as good as its contextual applicability. Once knowledge/information has been determined to be useful, and applicable to a particular context, its manageability must be determined, i.e., how it should be dispensed, who should be the recipients, what effects it will have on an organization and even the market in general.

2.8 Approaches to implementing knowledge management

A broad range of thoughts on the KM discipline exists with no unanimous agreement as approaches vary by author and school. As the discipline matures, academic debates have increased regarding both the theory and practice of KM, to include the following perspectives:

- Techno-centric with a focus on technology, ideally those that enhance knowledge sharing and creation.

- Organizational with a focus on how an organization can be designed to facilitate knowledge processes best.
- Ecological with a focus on the interaction of people, identity, knowledge, and environmental factors as a complex adaptive system akin to a natural ecosystem.

Regardless of the school of thought, core components of KM include People, Processes, Culture, Structure, Technology, depending on the specific perspective (Spender, et al., 2007). Different KM schools of thought include various lenses through which KM can be viewed and explained.

2.9 Knowledge Management Technologies & Frameworks

Different frameworks for distinguishing between knowledge exist. One proposed framework for categorizing the dimensions of knowledge distinguishes between tacit knowledge and explicit knowledge. Tacit knowledge represents internalized knowledge that an individual may not be consciously aware of, such as how one accomplishes particular tasks. At the opposite end of the spectrum, explicit knowledge represents knowledge that the individual holds consciously in mental focus, in a form that can easily be communicated to others (Alavi, et al., 2001). Similarly, Hayes and Walsham (2003) describe content and relational perspectives of knowledge and knowledge management as two fundamentally different epistemological perspectives. The content perspective suggest that knowledge is easily stored because it may be codified, while the relational perspective recognizes the contextual and relational aspects of knowledge which can make knowledge difficult to share outside of the specific location where the knowledge is developed.

The Knowledge Spiral as described by Nonaka & Takeuchi. Early research suggested that a successful KM effort needs to convert internalized tacit knowledge into explicit knowledge in order to share it, but the same effort must also permit individuals to internalize and make personally meaningful any codified knowledge retrieved from the KM effort. Subsequent research into KM suggested that a distinction between tacit knowledge and explicit knowledge represented an oversimplification and that the notion of explicit knowledge is self-contradictory. Specifically, for knowledge to be made explicit, it must be translated into information (i.e., symbols outside of our heads) (Serenko, et al., 2004). Later on, Ikujiro Nonaka proposed a model (SECI for Socialization, Externalization, Combination, Internalization) which considers a spiraling knowledge process interaction between explicit knowledge and tacit knowledge (Nonaka, et al., 1995) (p. 284). In this model, knowledge follows a cycle in which implicit knowledge is “extracted” to become explicit knowledge, and explicit knowledge is “re-internalized” into

implicit knowledge. More recently, together with Georg von Krogh, Nonaka returned to his earlier work in an attempt to move the debate about knowledge conversion forwards (Nonaka, et al., 2009).

A second proposed framework for categorizing the dimensions of knowledge distinguishes between embedded knowledge of a system outside of a human individual (e.g., an information system may have knowledge embedded into its design) and embodied knowledge representing a learned capability of a human body's nervous and endocrine systems (Sensky, 2002).

A third proposed framework for categorizing the dimensions of knowledge distinguishes between the exploratory creation of "new knowledge" (i.e., innovation) vs. the transfer or exploitation of "established knowledge" within a group, organization, or community. Collaborative environments such as communities of practice or the use of social computing tools can be used for both knowledge creation and transfer.

2.10 Developing a KM infrastructure and architecture

Knowledge management in most organizations is more left to individualistic initiatives of managers rather than having a structured system or process to be followed. Frameworks and models can provide a way of trying to tie together disparate initiatives and to also provide overarching strategies. Weaving metaphors into models or frameworks are useful ways of creating a highly understandable form of describing these in a way that is both context rich and resonates with the receiver's cultural perspective. Understanding a plan or framework for advancing innovation through knowledge should be made more tangible even though it may embed tacit cultural knowledge.

The model described in the K-Adv has three major infrastructure components. The key to this concept is competitive advantage. Figure 2 illustrates the key to delivering the goal as clearly being the result of the efforts of the people infrastructure. This is supported by a leadership infrastructure that provides resources and the necessary organizational support. "*An ICT infrastructure provides the linking support that allows people use business process to better*

collaborate and create, share and use knowledge" (Walker, et al., November 26th -29th 2006).

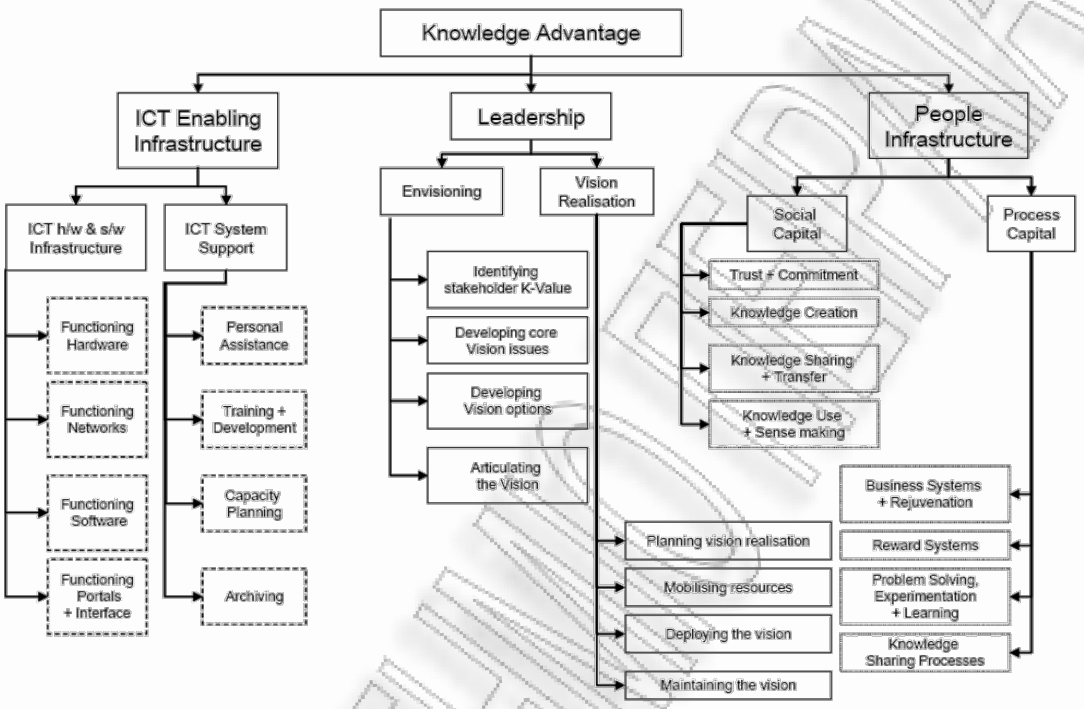

Figure 2 – The K-Adv model (Walker, et al., November 26th -29th 2006)

2.11 Business Intelligence

Business intelligence (BI) refers to computer-based techniques used in identifying, extracting, and analyzing business data, such as sales revenue by products and/or departments, or by associated costs and incomes.

BI technologies provide historical, current and predictive views of business operations. Common functions of business intelligence technologies are reporting, online analytical processing, analytics, data mining, business performance management, benchmarking, text mining and predictive analytics.

Business intelligence aims to support better business decision-making. Thus a BI system can be called a decision support system (DSS).Though the term

business intelligence is sometimes used as a synonym for competitive intelligence, because they both support decision making, BI uses technologies, processes, and applications to analyze mostly internal, structured data and business processes while competitive intelligence gathers, analyzes and disseminates information with a topical focus on company competitors. Business intelligence understood broadly can include the subset of competitive intelligence.

2.12 Business intelligence in the corporate environment: application, systems and processes

Business Intelligence can be applied to the following business purposes in order to drive business value:

1. Measurement – program that creates a hierarchy of Performance metrics (see also Metrics Reference Model) and Benchmarking that informs business leaders about progress towards business goals (a.k.a. Business process management).
2. Analytics – program that builds quantitative processes for a business to arrive at optimal decisions and to perform Business Knowledge Discovery. Frequently involves: data mining, statistical analysis, Predictive analytics, Predictive modeling, Business process modeling
3. Reporting/Enterprise Reporting – program that builds infrastructure for Strategic Reporting to serve the Strategic management of a business, NOT Operational Reporting. Frequently involves: Data visualization, Executive information system, OLAP
4. Collaboration/Collaboration platform – program that gets different areas and departments to work together through Data sharing and Electronic Data Interchange.
5. Knowledge Management – program to make the company data driven through strategies and practices to identify, create, represent, distribute, and enable adoption of insights and experiences that are true business knowledge. Knowledge

Management leads to Learning Management and Regulatory compliance/Compliance.

2.13 Semi-structured or unstructured data

Businesses create a huge amount of valuable information in the form of e-mails, memos, notes from call-centers, news, user groups, chats, reports, web-pages, presentations, image-files, video-files, and marketing material and news. According to Merrill Lynch, more than 85 percent of all business information exists in these forms. These information types are called either semi-structured or unstructured data. However, organizations often only use these documents once.

The management of semi-structured data is recognized as a major unsolved problem in the information technology industry. According to projections from Gartner (2003), white workers spend anywhere from 30%-40% of their time searching, finding and assessing unstructured data. BI uses both structured and unstructured data, but the former is easy to search, and the latter contains a large quantity of the information needed for analysis and decision making. Because of the difficulty of properly searching, finding and assessing unstructured or semi-structured data, organizations may not draw upon these vast reservoirs of information, which could influence a particular decision, task or project. This can ultimately lead to poorly-informed decision making.

Therefore, when designing a Business Intelligence/DW-solution, the specific problems associated with semi-structured and unstructured data must be accommodated for as well as those for the structured data.

2.14 Unstructured data vs. Semi-structured data

Unstructured and semi-structured data have different meanings depending on their context. In the context of relational database systems, it refers to data that cannot be stored in columns and rows. It must be stored in a BLOB

(binary large object), a catch-all data type available in most relational database management systems.

But many of these data types, like e-mails, word processing text files, PPTs, image-files, and video-files conform to a standard that offers the possibility of metadata. Metadata can include information such as author and time of creation, and this can be stored in a relational database. Therefore it may be more accurate to talk about this as semi-structured documents or data, but no specific consensus seems to have been reached.

Problems with semi-structured or unstructured data

There are several challenges to developing BI with semi-structured data. According to Inmon & Nesavich (Inmon, et al., 2007), some of those are:

- Physically accessing unstructured textual data – unstructured data is stored in a huge variety of formats.
- Terminology – Among researchers and analysts, there is a need to develop a standardized terminology.
- Volume of data – As stated earlier, up to 85% of all data exists as semi-structured data. Couple that with the need for word-to-word and semantic analysis.
- Searchability of unstructured textual data – A simple search on some data, e.g. apple, results in links where there is a reference to that precise search term. But a simple search is crude, as it does not find references to query terms.

2.15 The use of metadata

To solve the problem with the searchability and assessment of the data, it is necessary to know something about the content. This can be done by adding context through the use of metadata. A lot of system already captures some metadata, e.g. filename, author, size etc. But much more useful could be metadata about the actual content – e.g. summaries, topics, people or

companies mentioned. Two technologies designed for generating metadata about content is automatic categorization and information extraction.

2.16 Business intelligence strategies and systems

Before implementing a BI solution, it is worth taking different factors into consideration before proceeding. According to Kimball et al. These are the three critical areas that you need to assess within your organization before getting ready to do a BI project:

- The level of commitment and sponsorship of the project from senior management
- The level of business need for creating a BI implementation
- The amount and quality of business data available.

The commitment and sponsorship of senior management is according to Kimball et al, the most important criteria for assessment. This is because having strong management backing will help overcome shortcomings elsewhere in the project. But as Kimball et al state: "*even the most elegantly designed DW/BI system cannot overcome a lack of business management sponsorship*". It is very important that the management personnel who participate in the project have a vision and an idea of the benefits and drawbacks of implementing a BI system. The best business sponsor should have organizational clout and should be well connected within the organization. It is ideal that the business sponsor is demanding but also able to be realistic and supportive if the implementation runs into delays or drawbacks. The management sponsor also needs to be able to assume accountability and to take responsibility for failures and setbacks on the project. It is imperative that there is support from multiple members of the management so the project will not fail if one person leaves the steering group. However, having many managers that work together on the project can

also mean that the there are several different interests that attempt to pull the project in different directions. For instance if different departments want to put more emphasis on their usage of the implementation. This issue can be countered by an early and specific analysis of the different business areas that will benefit the most from the implementation. All stakeholders in project should participate in this analysis in order for them to feel ownership of the project and to find common ground between them. Another management problem that should be encountered before start of implementation is if the Business sponsor is overly aggressive. If the management individual gets carried away by the possibilities of using BI and starts wanting the DW or BI implementation to include several different sets of data that were not included in the original planning phase. However, since extra implementations of extra data will most likely add many months to the original plan. It is probably a good idea to make sure that the person from management is aware of his actions.

Implementation should be driven by clear business needs.

Because of the close relationship with senior management, another critical thing that needs to be assessed before the project is implemented is whether or not there actually is a business need and whether there is a clear business benefit by doing the implementation. The needs and benefits of the implementation are sometimes driven by competition and the need to gain an advantage in the market. Another reason for a business-driven approach to implementation of BI is the acquisition of other organizations that enlarge the original organization it can sometimes be beneficial to implement DW or BI in order to create more oversight.

The amount and quality of the available data ought to be the most important factor, since without good data – it does not really matter how good your management sponsorship or your business – driven motivation is. If you do not have the data, or the data does not have sufficient quality any BI implementation will fail. Before implementation it is a very good idea to do data profiling, this analysis will be able to describe the “content, consistency and structure” of the data. This should be done as early as possible in the

process and if the analysis shows that your data is lacking; it is a good idea to put the project on the shelf temporarily while the IT department figures out how to do proper data collection.

Other scholars have added more factors to the list than these three. In his thesis “Critical Success Factors of BI Implementation” Naveen Vodapalli does research on different factors that can impact the final BI product. He lists 7 crucial success factors for the implementation of a BI project, they are as follows:

1. Business-driven methodology and project management
2. Clear vision and planning
3. Committed management support & sponsorship
4. Data management and quality
5. Mapping solutions to user requirements
6. Performance considerations of the BI system
7. Robust and expandable framework

2.17 Ethical issues related to business intelligence.

Specific considerations for business intelligence systems have to be taken in some sectors such as governmental banking regulations. The information collected by banking institutions and analyzed with BI software must be protected from some groups or individuals, while being fully available to other groups or individuals. Therefore BI solutions must be sensitive to those needs and be flexible enough to adapt to new regulations and changes to existing laws.

2.18 Principles and concepts of knowledge discovery and data mining

Knowledge discovery is a concept of the field of computer science that describes the process of automatically searching large volumes of data for patterns that can be considered knowledge about the data. It is often described as deriving knowledge from the input data. This complex topic can be categorized according to (Jiawei Han & Micheline Kamber, 2006):

- What kind of data is searched
- In what form is the result of the search represented.

Knowledge discovery developed out of the Data mining domain, and is closely related to it both in terms of methodology and terminology.

The most well-known branch of data mining is knowledge discovery, also known as Knowledge Discovery in Databases (KDD). Just as many other forms of knowledge discovery it creates abstractions of the input data. The knowledge obtained through the process may become additional data that can be used for further usage and discovery.

With recent tremendous technical advances in processing power, storage capacity, and inter-connectivity of computer technology, data mining is seen as an increasingly important tool by modern business to transform unprecedented quantities of digital data into business intelligence giving an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud detection, and scientific discovery. The growing consensus that data mining can bring real value has led to an explosion in demand for novel data mining technologies.

The related terms data dredging, data fishing and data snooping refer to the use of data mining techniques to sample portions of the larger population data set that are too small for reliable statistical inferences to be made about the validity of any patterns discovered. These techniques can, however, be used in the creation of new hypotheses to test against the larger data populations.

2.19 The knowledge discovery process

Before one attempts to extract useful knowledge from data, it is important to understand the overall approach. Simply knowing many algorithms used for data analysis is not sufficient for a successful data mining (DM) project. The process defines a sequence of steps (with eventual feedback loops) that should be followed to discover knowledge (e.g., patterns) in data.

To formalize the knowledge discovery processes (KDPs) within a common framework, the concept of a process model is introduced. The model helps organizations to better understand the KDP and provides a roadmap to follow while planning and executing the project. This in turn results in cost and time savings, better understanding, and acceptance of the results of such projects. It is required to understand that such processes are nontrivial and involve multiple steps, reviews of partial results, possibly several iterations, and interactions with the data owners. There are several reasons to structure a KDP as a standardized process model:

- The end product must be useful for the user/owner of the data.
- A well-defined KDP model should have a logical, cohesive, well-thought-out structure and approach that can be presented to decision-makers who may have difficulty understanding the need, value, and mechanics behind a KDP.
- Knowledge discovery projects require a significant project management effort that needs to be grounded in a solid framework.
- Knowledge discovery should follow the example of other engineering disciplines that already have established models.
- There is a widely recognized need for standardization of the KDP.

The KDP model consists of a set of processing steps to be followed by practitioners when executing a knowledge discovery project. The model describes procedures that are performed in each of its steps. It is primarily used to plan, work through, and reduce the cost of any given project.

Since the 1990s, several different KDPs have been developed. The initial efforts were led by academic research but were quickly followed by industry. The first basic structure of the model was proposed by Fayyad et al. and later

improved/modified by others. The process consists of multiple steps that are executed in a sequence. Each subsequent step is initiated upon successful completion of the previous step, and requires the result generated by the previous step as its input. Another common feature of the proposed models is the range of activities covered, which stretches from the task of understanding the project domain and data, through data preparation and analysis, to evaluation, understanding, and application of the generated results. All the proposed models also emphasize the iterative nature of the model, in terms of many feedback loops that are triggered by a revision process. The main differences between the models described here lie in the number and scope of their specific steps. A common feature of all models is the definition of inputs and outputs. Typical inputs include data in various formats, such as numerical and nominal data stored in databases or flat files; images; video; semi-structured data, such as XML or HTML; etc. The output is the generated new knowledge — usually described in terms of rules, patterns, classification models, associations, trends, statistical analysis, etc.

2.20 Data preparation

Before data mining algorithms can be used, a target data set must be assembled. As data mining can only uncover patterns already present in the data, the target dataset must be large enough to contain these patterns while remaining concise enough to be mined in an acceptable timeframe. A common source for data is a datamart or data warehouse. Pre-process is essential to analyze the multivariate datasets before data mining.

The target set is then cleaned. Data cleaning removes the observations with noise and missing data.

Data cleansing or data scrubbing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database. Used mainly in databases, the term refers to identifying incomplete,

incorrect, inaccurate, irrelevant etc. parts of the data and then replacing, modifying or deleting this dirty data.

After cleansing, a data set will be consistent with other similar data sets in the system. The inconsistencies detected or removed may have been originally caused by different data dictionary definitions of similar entities in different stores, may have been caused by user entry errors, or may have been corrupted in transmission or storage.

Data cleansing differs from data validation in that validation almost invariably means data is rejected from the system at entry and is performed at entry time, rather than on batches of data.

The actual process of data cleansing may involve removing typographical errors or validating and correcting values against a known list of entities. The validation may be strict (such as rejecting any address that does not have a valid postal code) or fuzzy (such as correcting records that partially match existing, known records).

High quality data needs to pass a set of quality criteria. Those include:

- Accuracy: An aggregated value over the criteria of integrity, consistency and density
- Integrity: An aggregated value over the criteria of completeness and validity
- Completeness: Achieved by correcting data containing anomalies
- Validity: Approximated by the amount of data satisfying integrity constraints
- Consistency: Concerns contradictions and syntactical anomalies
- Uniformity: Directly related to irregularities
- Density: The quotient of missing values in the data and the number of total values ought to be known
- Uniqueness: Related to the number of duplicates in the data

2.21 Survey of knowledge repositories: document management systems, content management systems, data warehousing

A document management system (DMS) is a computer system (or set of computer programs) used to track and store electronic documents and/or images of paper documents. It is usually also capable of keeping track of the different versions created by different users (history tracking). The term has some overlap with the concepts of content management systems. It is often viewed as a component of enterprise content management (ECM) systems and related to digital asset management, document imaging, workflow systems and records management systems.

Content management system (CMS) is the collection of procedures used to manage work flow in a collaborative environment. These procedures can be manual or computer-based. The procedures are designed to do the following:

- Allow for a large number of people to contribute to and share stored data
- Control access to data, based on user roles (defining which information users or user groups can view, edit, publish, etc.)
- Aid in easy storage and retrieval of data
- Reduce repetitive duplicate input
- Improve the ease of report writing
- Improve communication between users

In a CMS, data can be defined as nearly anything: documents, movies, pictures, phone numbers, scientific data, and so forth. CMSs are frequently used for storing, controlling, revising, semantically enriching, and publishing documentation. Serving as a central repository, the CMS increases the version level of new updates to an already existing file. Version control is one of the primary advantages of a CMS.

An enterprise content management system (ECM) is content, documents, details and records related to the organizational processes of an enterprise. The purpose and result is to manage the organization's unstructured

information content, with all its diversity of format and location. The system manages the content related commercial organizations.

A data warehouse (DW) is a database used for reporting. The data is offloaded from the operational systems for reporting. The data may pass through an operational data store for additional operations before it is used in the DW for reporting. A data warehouse maintains its functions in three layers, Staging, Integration and Access (Figure 4).

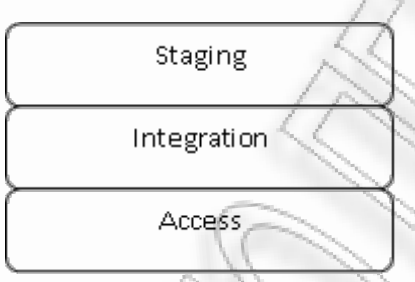

Figure 3 – The three layers of data warehouse functions

Staging is used to store raw data for use by developers (analysis and support). The integration layer is used to integrate data and to have a level of abstraction from users. The access layer is for getting data out for users.

This definition of the data warehouse focuses on data storage. The main source of the data is cleaned, transformed, catalogued and made available for use by managers and other business professionals for data mining, online analytical processing, market research and decision support (O'Brien, et al., 2009). However, the means to retrieve and analyze data, to extract, transform and load data, and to manage the data dictionary are also considered essential components of a data warehousing system. Many references to data warehousing use this broader context. Thus, an expanded definition for data warehousing includes business intelligence tools, tools to extract, transform and load data into the repository, and tools to manage and retrieve metadata.

2.22 Conclusions

All researchers noted that effective transfer of knowledge (knowledge management, knowledge sharing) relies, inter alia, on the political and historical context of that knowledge being appropriately captured, classified, safeguarded and disseminated as part of the transfer process.

Establishing such a "knowledge framework" in a professional, high quality, organization-wide manner is the traditional role of archivists and records managers - namely to provide the right information to the right people at the right time, at reasonable cost.

Well run archives and records management services provide value to knowledge-based organizations and their staff by:

1. facilitating access to information through standards-based classification, selective dissemination of new information to appropriate users, and the provision of user-friendly on-line finding aids, and
2. ensuring the information accountability required by legislation and good business practice.

The challenges are to do the above jobs well in an increasingly digital environment. The change to digital information in our organizations has :

1. allowed staff to communicate in a highly networked manner without central nodes, making the capture of substantive information more difficult;
2. required more effort to maintain the organization-wide overview of activities;
3. allowed numerous local information storage and retrieval tools to be established - many of which ignore or are incompatible with records management requirements.

Therefore, it is recommended that archivists and records managers formulate an awareness and training program emphasizing the assistance provided to the knowledge transfer activities in their organization; on-line tools to be developed and deployed that address specific user problem areas, including inter alia :

- addition of records management features to the organisation's electronic mail services, such as selection, classification and collection of substantive e-mail
- assistance in implementing information security regimes for electronic information
- inclusion of records management regimes in the development projects for web-based portals and discussions forums, for example ensuring that the resulting knowledge objects can be classified according to standard taxonomies that enable the information to be included in records management processes.

Researchers also recommend the above to its members for use within their organizations, and charges the members to follow up on the above items, via an electronic discussion forum in order to develop best practices.

Knowledge management or knowledge sharing can be defined as combining people, processes and technology, to share information in order to gain a competitive advantage. This information is not necessarily codified in a database.

2.23 Problems facing the Organizations

Organizations are facing is the ever increasing quantity of data available (in fact doubling every year) and the challenge is how and what to select, where to put it and how to find it.

Another problem is the development of complex networks. In the past, all the information (incoming and outgoing) went through one unit (organ). In today's organizations, exist a multitude of networks of people meeting together, sharing information sometime on an ephemeral basis: these communities of practice are composed of experts, getting together on issue(s) then disappearing. The challenge is to capture this information.

Cultivate organizational culture on how to convince people to share with others their knowledge. The role of archivists, librarians and IT is to participate in this culture change. It is also to predict what will be the Organization's needs in the next years to come.

1. Electronic mail: quantity, selection and format
2. Decentralized repositories:
   2.1. common search
   2.2. common indexing
   2.3. simple retrieval
   2.4. common security
3. Capturing the information which has not yet been recorded
4. Lifecycle of knowledge:
   4.1. thinking in advance how knowledge is created
   4.2. where it is deposited
   4.3. how it can be optimized
   4.4. Archivists should be focused on the Organizations goals/ missions.
5. Ldba: learning before, during and after
6. A knowledge object:
   6.1. a book/piece of information/document that some attributes attached to it which specifies how it can be used
   6.2. the role of archivists and records managers is to handle novel tasks in a collective manner
7. Online finding aids
8. Give value to internal information
9. Train others
10. Today everyone believes is an information expert show how to assist users
11. Marketing skills - providing quality assurance of information
12. Capturing information from portals
13. Capturing context of information
14. Archives and records management is a process of facilitating retention and access of information

3 Vector Space Model

In a document retrieval, or other pattern matching environment where stored entities (documents) are compared with each other or with incoming patterns (search requests), it appears that the best indexing (property) space is one where each entity lies as far away from the others as possible; in these circumstances the value of an indexing system may be expressible as a function of the density of the object space; in particular, retrieval performance may correlate inversely with space density. An approach based on space density computations is used to choose an optimum indexing vocabulary for a collection of documents.

3.1 Definitions of the model

Documents and queries are represented as vectors.

$$d_j = w_{1,j}, w_{2,j}, \ldots w_{t,j}$$

$$q = w_{1,q}, w_{2,q}, \ldots w_{t,q}$$

Each dimension corresponds to a separate term. If a term occurs in the document, its value in the vector is non-zero. Several different ways of computing these values, also known as (term) weights, have been developed. One of the best known schemes is tf-idf weighting.

The definition of term depends on the application. Typically terms are single words, keywords, or longer phrases. If the words are chosen to be the terms, the dimensionality of the vector is the number of words in the vocabulary (the number of distinct words occurring in the corpus). Vector operations can be used to compare documents with queries.

3.2 Similarity Coefficients

Two documents in the Vector Space Model represent two points in a multidimensional term space (each term is assumed to be an independent dimension). If we define a notion of distance in this space, we can compare documents against each other and thus start looking for similarities or dissimilarities. Any distance metric applicable to a multidimensional vector space is applicable, but two methods are widely used: Euclidean distance and cosine measure. A simple Euclidean distance is quite often used, but requires document vector length normalization prior to calculation or the number of words (proportion of weights) in each document will distort the result.

Cosine measure is a more robust technique stemming from the observation that if two vectors have approximately the same features then they should "point" at a very similar direction in the space determined by the term-document matrix, regardless of their Euclidean distance. To calculate similarity between two documents we need to look at the angle between them, which we can calculate using the dot product between their document vectors. To simplify things even more, we can use the cosine of this angle which is easier to compute (does not require hyperbolic function).

The cosine measure is widely used in text clustering any many other text processing applications because its definition is quite intuitive and its implementation efficient. However, it is also known that in highly dimensional spaces any two random vectors are very likely to be orthogonal. An attempt to solve this problem is to reduce the dimensionality dimensionality reduction of the feature space using feature selection, feature construction or term-document matrix decomposition techniques

3.3 Applications of VSM

Relevance rankings of documents in a keyword search can be calculated, using the assumptions of document similarities theory, by comparing the deviation of angles between each document vector and the original query

vector where the query is represented as same kind of vector as the documents.

In practice, it is easier to calculate the cosine of the angle between the vectors, instead of the angle itself:

$$\cos\theta = \frac{d_2 \cdot q}{\|d_2\| \cdot \|q\|}$$

Where $d_2 \cdot q$ is the intersection (i.e. that dot product) of the document (d_2 in the figure 3) and the query (q in the figure) vectors, $\|d_2\|$ is the norm of vector d_2, and $\|q\|$ is the norm of vector q. The norm of a vector is calculated as such:

$$\|v\| = \sqrt{\sum_{i=1}^{n} v_i^2}$$

A cosine value of zero means that the query and document vector are orthogonal and have no match (i.e. the query term does not exist in the document being considered).

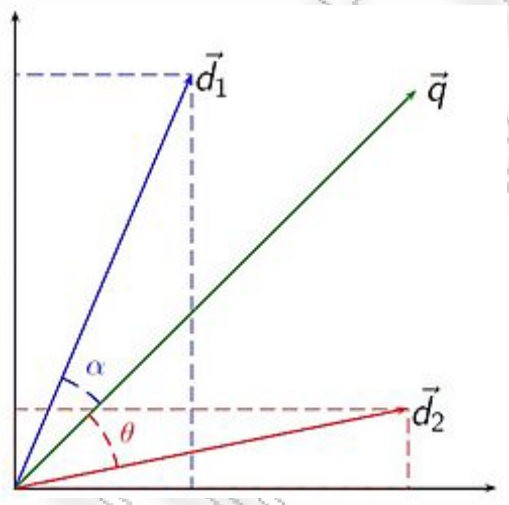

Figure 4 – Vector Space Model

3.4 tf - idf (term frequency–inverse document frequency) weighting

The $tf - idf$ weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the $tf - idf$ weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query.

The term count in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents, which may have a higher term count regardless of the actual importance of that term in the document, to give a measure of the importance of the term t_i within the particular document d_j. Thus we have the term frequency, defined as follows:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Where $n_{i,j}$ is the number of occurrences of the considered term (t_j) in document d_j, and the denominator is the sum of number of occurrences of all terms in document d_j, that is, the size of the document $\|d_j\|$.

The inverse document frequency is a measure of the general importance of the term (obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient).

$$idf_i = log \frac{|D|}{|\{j: t_i \in d_j\}|}$$

With:

$|D|$: Cardinality of D, or the total number of documents in the corpus

$|\{j: t_i \in d_j\}|$: Number of documents where the term t_i appears ($n_{i,j} \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to use $1 + |\{j: t_i \in d_j\}|$

Then:

$$(tf - idf)_{i,j} = tf_{i,j} \times idf_i$$

A high weight in $tf - idf$ is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms. The $tf - idf$ value for a term will be greater than zero if and only if the ratio inside the idf's log function is greater than 1. Depending on whether a 1 is added to the denominator, a term in all documents will have either a zero or negative idf, and if the 1 is added to the denominator a term that occurs in all but one document will have an idf equal to zero.

Various (mathematical) forms of the $tf - idf$ term weight can be derived from a probabilistic retrieval model that mimicks human relevance decision making.

3.5 Advantages

The vector space model has the following advantages over the Standard Boolean model:

- Simple model based on linear algebra
- Term weights not binary
- Allows computing a continuous degree of similarity between queries and documents
- Allows ranking documents according to their possible relevance
- Allows partial matching

3.6 Limitations

The vector space model has the following limitations:

- Long documents are poorly represented because they have poor similarity values (a small scalar product and a large dimensionality)
- Search keywords must precisely match document terms; word substrings might result in a "false positive match"
- Semantic sensitivity; documents with similar context but different term vocabulary won't be associated, resulting in a "false negative match".
- The order in which the terms appear in the document is lost in the vector space representation.
- Assumes terms are statistically independent
- Weighting is intuitive but not very formal

3.7 Internet Technologies & Applications

3.8 Web-based architectures: design, security and management

Web applications present a complex set of security issues for architects, designers, and developers. The most secure and resilient Web applications are those that have been built from the ground up with security in mind.

In addition to applying sound architectural and design practices, incorporate deployment considerations and corporate security policies during the early design phases. Failure to do so can result in applications that cannot be deployed on an existing infrastructure without compromising security.

The stateless nature of HTTP means that tracking per-user session state becomes the responsibility of the application. As a precursor to this, the application must be able to identify the user by using some form of authentication. Given that all subsequent authorization decisions are based on the user's identity, it is essential that the authentication process is secure and that the session handling mechanism used to track authenticated users is equally well protected. Designing secure authentication and session management mechanisms are just a couple of the issues facing Web application designers and developers. Other challenges occur because input and output data passes over public networks. Preventing parameter manipulation and the disclosure of sensitive data are other top issues.

Table 1 – Vulnerabilities linked with design issues

Vulnerability Category	Potential Problem Due to Bad Design
Input Validation	Attacks performed by embedding malicious strings in query strings, form fields, cookies, and HTTP headers. These include command execution, cross-site scripting (XSS), SQL injection, and buffer overflow attacks.
Authentication	Identity spoofing, password cracking, elevation of privileges, and unauthorized access.

Authorization	Access to confidential or restricted data, tampering, and execution of unauthorized operations.
Configuration Management	Unauthorized access to administration interfaces, ability to update configuration data, and unauthorized access to user accounts and account profiles.
Sensitive Data	Confidential information disclosure and data tampering.
Session Management	Capture of session identifiers resulting in session hijacking and identity spoofing.
Cryptography	Access to confidential data or account credentials, or both.
Parameter Manipulation	Path traversal attacks, command execution, and bypass of access control mechanisms among others, leading to information disclosure, elevation of privileges, and denial of service.
Exception Management	Denial of service and disclosure of sensitive system level details.
Auditing and Logging	Failure to spot the signs of intrusion, inability to prove a user's actions, and difficulties in problem diagnosis.

(Source: http://msdn.microsoft.com/en-us/library/ff648647.aspx)

During the application design phase, the designer should review the corporate security policies and procedures together with the infrastructure the application is to be deployed on. Frequently, the target environment is rigid, and the application design must reflect the restrictions. Sometimes design tradeoffs are required, because of protocol or port restrictions, or specific deployment topologies. The designer should identify constraints early in the design phase in order to avoid surprises later and involve members of the network and infrastructure teams to help with this process.

3.9 Internet standards and E-business components: XML, SOAP, WSDL, UDDI

Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form. It is defined in the XML 1.0 Specification produced

by the W3C, and several other related specifications, all gratis open standards. XML's design goals emphasize simplicity, generality, and usability over the Internet. It is a textual data format with strong support via Unicode for the languages of the world. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services.

Many application programming interfaces (APIs) have been developed that software developers use to process XML data, and several schema systems exist to aid in the definition of XML-based languages.

SOAP, originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of Web Services in computer networks. It relies on Extensible Markup Language (XML) for its message format, and usually relies on other Application Layer protocols, most notably Remote Procedure Call (RPC) and Hypertext Transfer Protocol (HTTP), for message negotiation and transmission. SOAP can form the foundation layer of a web services protocol stack, providing a basic messaging framework upon which web services can be built. This XML based protocol consists of three parts: an envelope, which defines what is in the message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing procedure calls and responses.

As an example of how SOAP procedures can be used, a SOAP message could be sent to a web-service-enabled web site such as a real-estate price database, with the parameters needed for a search. The site would then return an XML-formatted document with the resulting data, e.g., prices, location, features. With the data being returned in a standardized machine-parseable format, it can then be integrated directly into a third-party web site or application.

The SOAP architecture consists of several layers of specifications: for message format, Message Exchange Patterns (MEP), underlying transport protocol bindings, message processing models, and protocol extensibility.

SOAP is the successor of XML-RPC, though it borrows its transport and interaction neutrality and the envelope/header/body from elsewhere.

A Web service is a method of communication between two electronic devices over a network.

The W3C defines a Web service as "*a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically Web Services Description Language WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards*."

The W3C also states, "*We can identify two major classes of Web services, REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of "stateless" operations; and arbitrary Web services, in which the service may expose an arbitrary set of operations.*"

The Web Services Description Language (WSDL) is an XML-based language that provides a model for describing Web services. The WSDL defines services as collections of network endpoints, or ports. The WSDL specification provides an XML format for documents for this purpose. The abstract definitions of ports and messages are separated from their concrete use or instance, allowing the reuse of these definitions. A port is defined by associating a network address with a reusable binding, and a collection of ports defines a service. Messages are abstract descriptions of the data being exchanged, and port types are abstract collections of supported operations. The concrete protocol and data format specifications for a particular port type constitutes a reusable binding, where the operations and messages are then bound to a concrete network protocol and message format. In this way, WSDL describes the public interface to the Web service.

WSDL is often used in combination with SOAP and an XML Schema to provide Web services over the Internet. A client program connecting to a Web

service can read the WSDL file to determine what operations are available on the server. Any special datatypes used are embedded in the WSDL file in the form of XML Schema. The client can then use SOAP to actually call one of the operations listed in the WSDL file.

Universal Description, Discovery and Integration (UDDI) is a platform-independent, XML-based registry for businesses worldwide to list themselves on the Internet and a mechanism to register and locate web service applications. UDDI is an open industry initiative, sponsored by the Organization for the Advancement of Structured Information Standards (OASIS), enabling businesses to publish service listings and discover each other and define how the services or software applications interact over the Internet.

UDDI was originally proposed as a core Web service standard. It is designed to be interrogated by SOAP messages and to provide access to Web Services Description Language (WSDL) documents describing the protocol bindings and message formats required to interact with the web services listed in its directory.

A UDDI business registration consists of three components:

- White Pages — address, contact, and known identifiers;
- Yellow Pages — industrial categorizations based on standard taxonomies;
- Green Pages — technical information about services exposed by the business.

White pages give information about the business supplying the service. This includes the name of the business and a description of the business - potentially in multiple languages. Using this information, it is possible to find a service about which some information is already known, for example, locating a service based on the provider's name. Contact information for the business is also provided - for example the businesses address and phone number.

Yellow pages provide a classification of the service or business, based on standard taxonomies. These include the Standard Industrial Classification

(SIC), the North American Industry Classification System (NAICS), or the United Nations Standard Products and Services Code (UNSPSC). Because a single business may provide a number of services, there may be several Yellow Pages (each describing a service) associated with one White Page.

Green pages are used to describe how to access a Web Service, with information on the service bindings. Some of the information is related to the Web Service - such as the address of the service and the parameters, and references to specifications of interfaces. Other information is not related directly to the Web Service - this includes e-mail, FTP, CORBA and telephone details for the service. Because a Web Service may have multiple bindings, as defined in its WSDL description, a service may have multiple Green Pages, as each binding will need to be accessed differently.

3.10 Web application development techniques: client and server-side programming

Web Development can be split into many areas and a typical and basic web development hierarchy might consist of:

Client Side Coding

- Ajax Asynchronous JavaScript provides new methods of using JavaScript, and other languages to improve the user experience.
- Flash Adobe Flash Player is an ubiquitous browser plugin ready for RIAs. Flex 2 is also deployed to the Flash Player (version 9+).
- JavaScript Formally called ECMAScript, JavaScript is a ubiquitous client side platform for creating and delivering rich Web applications that can also run across a wide variety of devices.
- Microsoft Silverlight Microsoft's browser plugin that enables animation, vector graphics and high-definition video playback, programmed using XAML and .NET programming languages.
- Real Studio Web Edition is a rapid application development environment for the web. The language is object oriented and is similar

to both VB and Java. Applications are uniquely compiled to binary code.

- HTML5 and CSS3 Latest HTML proposed standard combined with the latest proposed standard for CSS natively supports much of the client-side functionality provided by other frameworks such as Flash and Silverlight

Server Side Coding

- ASP (Microsoft proprietary)
- CSP, Server-Side ANSI C
- ColdFusion (Adobe proprietary, formerly Macromedia, formerly Allaire)
- CGI and/or Perl (open source)
- Groovy (programming language) Grails (framework)
- Java, e.g. Java EE or WebObjects
- Lotus Domino
- PHP (open source)
- Python, e.g. Django (web framework) (open source)
- Real Studio Web Edition
- Ruby, e.g. Ruby on Rails (open source)
- Smalltalk e.g. Seaside, AIDA/Web
- SSJS Server-Side JavaScript, e.g. Aptana Jaxer, Mozilla Rhino
- Websphere (IBM proprietary)
- .NET (Microsoft proprietary)

The World Wide Web has become a major delivery platform for web development a variety of complex and sophisticated enterprise applications in several domains. In addition to their inherent multifaceted functionality, these web applications exhibit complex behavior and place some unique demands on their usability, performance, security and ability to grow and evolve. However, a vast majority of these applications continue to be developed in an ad-hoc way, contributing to problems of usability, maintainability, quality and reliability. While web development can benefit from established practices from

other related disciplines, it has certain distinguishing characteristics that demand special considerations. In recent years of web development there have been some developments towards addressing these problems and requirements. As an emerging discipline, web engineering actively promotes systematic, disciplined and quantifiable approaches towards successful development of high-quality, ubiquitously usable web-based systems and applications. In particular, web engineering focuses on the methodologies, techniques and tools that are the foundation of web application development and which support their design, development, evolution, and evaluation. Web application development has certain characteristics that make it different from traditional software, information system, or computer application development.

Web engineering is multidisciplinary and encompasses contributions from diverse areas: systems analysis and design, software engineering, hypermedia/hypertext engineering, requirements engineering, human-computer interaction, user interface, information engineering, information indexing and retrieval, testing, modeling and simulation, project management, and graphic design and presentation. Web engineering is neither a clone, nor a subset of software engineering, although both involve programming and software development. While web engineering uses software engineering principles, web development encompasses new approaches, methodologies, tools, techniques, and guidelines to meet the unique requirements for web-based applications.

Client Side & Server Side

- Google Web Toolkit provides tools to create and maintain complex JavaScript front-end applications in Java.
- Pyjamas is a tool and framework for developing Ajax applications and Rich Internet Applications in python.
- Tersus is a platform for the development of rich web applications by visually defining user interface, client side behavior and server side processing. (open source)

However lesser known languages like Ruby and Python are often paired with database servers other than MySQL (the M in LAMP). Below are examples of

other databases currently in wide use on the web. For instance some developers prefer a LAPR (Linux/Apache/PostgreSQL/Ruby on Rails) setup for development.

Database Technology

- Apache Derby
- DB2 (IBM proprietary)
- Firebird
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL
- SQLite
- Sybase

3.11 Ontologies, Taxonomies and Hierarchies

Information overload continues to be a challenge for our end users. For instance, in the corporate world, knowledge workers are faced with solving the information overload problem and we are working to connect end users with the information they need. Simple search is not always the answer because end users tend to enter concepts that are either too broad or are so specific that they do not retrieve key relevant information. Information management principles and practices, taxonomies, and other controlled vocabularies all

serve as knowledge management tools that librarians can use to help organize content and make connections between people and the information they need.

An ontology is a formal representation of knowledge as a set of concepts within a domain, and the relationships between those concepts. It is used to reason about the entities within that domain, and may be used to describe the domain. Its meaning is vastly different from the word Ontology in philosophy.

In theory, an ontology is a formal, explicit specification of a shared conceptualization. An ontology provides a shared vocabulary, which can be used to model a domain — that is, the type of objects and/or concepts that exist, and their properties and relations.

Ontologies are the structural frameworks for organizing information and are used as a form of knowledge representation about the world or some part of it. The creation of domain ontologies is also fundamental to the definition and use of an enterprise architecture framework.

Contemporary ontologies share many structural similarities, regardless of the language in which they are expressed. As mentioned above, most ontologies describe individuals (instances), classes (concepts), attributes, and relations. In this section each of these components is discussed in turn.

Common components of ontologies include:

- Individuals: instances or objects (the basic or "ground level" objects)
- Classes: sets, collections, concepts, classes in programming, types of objects, or kinds of things
- Attributes: aspects, properties, features, characteristics, or parameters that objects (and classes) can have
- Relations: ways in which classes and individuals can be related to one another
- Function terms: complex structures formed from certain relations that can be used in place of an individual term in a statement

- Restrictions: formally stated descriptions of what must be true in order for some assertion to be accepted as input
- Rules: statements in the form of an if-then (antecedent-consequent) sentence that describe the logical inferences that can be drawn from an assertion in a particular form
- Axioms: assertions (including rules) in a logical form that together comprise the overall theory that the ontology describes in its domain of application. This definition differs from that of "axioms" in generative grammar and formal logic. In those disciplines, axioms include only statements asserted as a priori knowledge. As used here, "axioms" also include the theory derived from axiomatic statements
- Events: the changing of attributes or relations

Ontologies are commonly encoded using ontology languages.

A taxonomy is a controlled vocabulary with each term having hierarchical (broader and narrower) and equivalent (synonymous) relationships. Because of its hierarchical nature, a taxonomy imposes a topical structure on information.

Broader and narrower terms are essential for a browsable hierarchy. If the terminology is too specific and you cannot retrieve anything, you can move up the hierarchy to less specificity. The reverse is also true as if too much information is retrieved; moving down within the hierarchy will narrow the results. The use of hierarchical relationships is the primary feature that distinguishes a taxonomy from other lesser forms of controlled vocabularies, such as lists and synonym rings.

Equivalent relationships (synonyms) are also embedded in a taxonomy. Synonyms gather together all concepts of a similar nature. The use of the synonym ring helps cast a wide net for information recall.

By using the terms in the taxonomy, one can consistently categorize the information available. Using taxonomic subject categories in searches simplifies the search construction process. The searcher does not have to

define the subject or master the vocabulary of terms unique to that subject in order to search for information.

As far as possible, category labels should be:

- Phrased in the user's language.
- Unambiguous.
- Mutually exclusive (non-overlapping), so users know where to look.
- Comprehensive: i.e. completely partition the parent category, so users do not suspect a category is missing.

4 Research Methodology

4.1 Overview

Qualitative research explores attitudes, behavior and experiences through such methods as interviews or focus groups. It attempts to get an in-depth opinion from participants. As it is attitudes, behavior and experiences which are important, fewer people take part in the research, but the contact with these people tends to last a lot longer. Under the umbrella of qualitative research there are many different methodologies.

Quantitative research generates statistics through the use of large-scale survey research, using methods such as questionnaires or structured interviews. If a market researcher has stopped you on the streets, or you have filled in a questionnaire which has arrived through the post, this falls under the umbrella of quantitative research. This type of research reaches many more people, but the contact with those people is much quicker than it is in qualitative research.

A research method is a strategy of inquiry which moves from various and underlying assumptions to research design and data collection. The choice of research method influences the way in which the researcher collects data. Specific research methods also imply different skills, assumptions and research practices. Three fundamental research methods will be discussed below. These are:

- Action research
- Case study research and
- Grounded theory

4.2 Qualitative versus quantitative inquiry

Over the years there has been a large amount of complex discussion and argument surrounding the topic of research methodology and the theory of how inquiry should proceed. Much of this debate has centered on the issue of qualitative versus quantitative inquiry – which might be the best and which is more 'scientific'. Different methodologies become popular at different social, political, historical and cultural times in our development, and, in my opinion, all methodologies have their specific strengths and weaknesses. These should be acknowledged and addressed by the researcher. Certainly, if you were to do so, it would help you to think about your research methodology in considerable depth.

4.3 Action Research

Action research is an established research method in use in the social and medical sciences since the mid-twentieth century, and has increased in importance for information systems toward the end of the 1990s. Action research has developed a history within information systems. Action research varies in form, and responds to particular problem domains. The most typical form is a participatory method based on a five-step model, which will be explained later on. The method produces highly relevant research results, because it is grounded in practical action, aimed at solving an immediate problem situation while carefully informing theory.

Adapting Hult and Lennung's definition (1980) four major characteristics of IS action research, are distinguishable:

1. Action research aims at an increased understanding of an immediate social situation, with emphasis on the complex and multivariate nature of this social setting in the Information System domain.
2. Action research simultaneously assists in practical problem solving and expands scientific knowledge. This goal extends into two important process characteristics: First, there are highly interpretive assumptions

being made about observation; second, the researcher intervenes in the problem setting.

3. Action research is performed collaboratively and enhances the competencies of the respective actors. A process of participatory observation is implied by this goal.
4. Action research is primarily applicable for the understanding of change processes in social systems.

Action research refers to a class of research approaches, rather than a single, monolithic research method. As a class, the various forms of action research share some agreed characteristics, and these characteristics distinguish action research from other approaches to social enquiry. A careful survey of the action research literature finds widespread agreement by action research authorities on four common characteristics: an action and change orientation; a problem focus; a process involving systematic and sometimes iterative stages and collaboration among participants

Action research has been described as a technique characterized by intervention experiments that operate on problems or questions perceived by practitioners within a particular context. The type of learning created by action research represents enhanced understanding of a complex social-organizational problem. The domain of information systems action research is clearest where the human interacts with information systems.

The ideal domain of the action research method is characterized by a social setting where:

1. The researcher is actively involved, with expected benefit for both researcher and organization,
2. The knowledge obtained can be immediately applied, there is not the sense of the detached observer, but that of an active participant wishing to utilize any new knowledge based on these observations,
3. The research is a cyclical process linking theory and practice

One clear area of importance in the ideal domain of action research is new or changed systems development methodologies. Studying new or changed

methodologies involves the introduction of such changes. Theoretically, the study of a newly invented technique is impossible without intervening in some way to system been changed or developed and apply new technique into this environment. Action research is one of the few valid research approaches that we can study the effects of specific alterations in systems development methodologies in Information Systems.

The most prevalent action research description Susman & Evered (1978) details a five phases, cyclical process. The approach first requires the establishment of a client-system infrastructure or research environment. Then, five identifiable phases are iterated:

1. Diagnosing
2. Action planning
3. Action taking
4. Evaluating and
5. Specifying learning

The figure below illustrates this action research structural cycle.

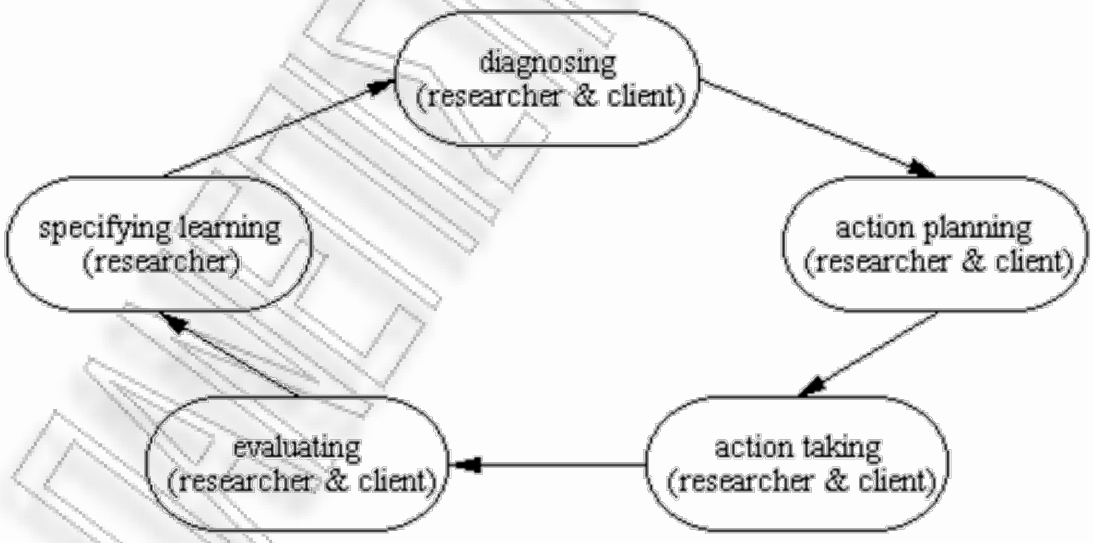

Figure 5 - Action Research Structural Cycle (Source: http://www.scu.edu.au/schools/gcm/ar/arr/arow/kms.html)

4.3.1 Diagnosing

Diagnosing corresponds to the identification of the primary problems that are the underlying causes of the organization's desire for change. Diagnosing involves self-interpretation of the complex organizational problem, not through reduction and simplification, but rather in a holistic context. This diagnosis will develop certain theoretical assumptions about the nature of the organization and its problem domain.

4.3.2 Action Planning

Researchers and practitioners then collaborate in the next activity, action planning. This activity specifies organizational actions that should relieve or improve these primary problems. The discovery of the planned actions is guided by the theoretical framework, which indicates both some desired future state for the organization, and the changes that would achieve such a state. The plan establishes the target for change and the approach to change.

4.3.3 Action Taking

Action taking then implements the planned action. The researchers and practitioners collaborate in the active intervention into the client organization, causing certain changes to be made. Several forms of intervention strategy can be adopted. For example, the intervention might be directive, in which the research "directs" the change, or non-directive, in which the change is sought indirectly. Intervention tactics can also be adopted, such as recruiting intelligent laypersons as change catalysts and pacemakers. The process can draw its steps from social psychology, e.g., engagement, unfreezing, learning and re-framing.

4.3.4 Evaluating

After the actions are completed, the collaborative researchers and practitioners evaluate the outcomes. Evaluation includes determining whether the theoretical effects of the action were realized, and whether these effects relieved the problems. Where the change was successful, the evaluation must

critically question whether the action undertaken, among the myriad routine and non-routine organizational actions, was the sole cause of success. Where the change was unsuccessful, some framework for the next iteration of the action research cycle (including adjusting the hypotheses) should be established.

4.3.5 Specifying Learning

While the activity of specifying learning is formally undertaken last, it is usually an ongoing process. The knowledge gained in the action research (whether the action was successful or unsuccessful) can be directed to three audiences. First, what Argyris and Schön (1978) call "double-loop learning," the restructuring of organizational norms to reflect the new knowledge gained by the organization during the research. Second, where the change was unsuccessful, the additional knowledge may provide foundations for diagnosing in preparation for further action research interventions.

Finally, the success or failure of the theoretical framework provides important knowledge to the scientific community for dealing with future research settings.

The action research cycle can continue, whether the action proved successful or not, to develop further knowledge about the organization and the validity of relevant theoretical frameworks. As a result of the studies, the organization thus learns more about its nature and environment, and the constellation of theoretical elements of the scientific community continues to benefit and evolve

.

4.3.6 Data Collection Method

Sample dataset based on regulatory restrictions.

5 Knowledge Management Measurement

5.1 Role of performance measurement in KM

The intangible nature of knowledge itself causes some KM practitioners to assume that the effects of KM will also be intangible. According to research, firms can and do effectively measure the impact of KM. In fact, those who invest the most and measure most rigorously are achieving a financial return on investment (ROI) of approximately 2:1 for the amount spent per participating employee - a healthy ROI by any standard. These returns are added to valuable intangibles such as increased sense of belonging, faster socialization of issues and change, cross-fertilization of ideas, and so on.

5.2 KM performance measures: financial, customer, internal processes, innovation and growth

The following, offers five steps for creating and sustaining successful KM measurement programs (APQ11).

Step No. 1: Start with a measurement paradigm that drives KM linkage to business needs.

Far too many KM measurement attempts focus exclusively on activity metrics such as number of communities of practice, number of documents downloaded, and number of people who participate. While these are critical indicators of the health and adoption of knowledge-sharing practices, they are not an end in themselves. KM and business alignment framework provides a different paradigm. Starting with the value chain along the top, a KM measurement system should incorporate business outcomes as the focal point for the KM strategy and a way to measure its effectiveness.

Once an organization defines the business objectives for KM, the knowledge flow processes – such as communities – need to be established and their

activity levels tracked. The goal is to correlate trends in activity measures to business outcomes. Clear business outcomes provide the ROI to justify investment in the KM approaches as well as the necessary people and technology enablers and infrastructure that any successful initiative requires.

Step No. 2: Select measures that are appropriate to your organization's particular KM approach, its objectives, and its stage of development.

In the early stages of deployment, any KM strategy needs measures of alignment with business strategy, acceptance, and behavior change, as well as a method to predict desired business outcomes and begin tracking them. However, the way in which an organization measures the particular costs and impacts of its KM program depends on the KM approach(es) adopted. For example, a KM initiative focused on improving sales force effectiveness would track the reuse of effective proposals (activity) and sales (outcome), but such measures would probably be irrelevant to a KM initiative centered on building new knowledge in an engineering discipline. Likewise, an enterprise whose goal is to implement communities of practice would measure success differently than wound an organization that wants to install a content management system.

Step No. 3: Understand the linkage between inputs, process changes, and desired outcomes.

The value path model shows the relationships among inputs (investments), processes (KM-related activities and behaviors), and outcomes (organization objectives). Depending on the particular KM activities being performed, examples of inputs might include time, salaries, and IT costs. Process changes might include cycle time, participation, and contribution to a body of knowledge. Examples of outcomes important to the organization might include employee and customer retention, reduced costs per transaction, or increased revenue.

Step No. 4: Create a measurement system that actually works.

Many organizations have lists of measures, but lack the necessary processes and accountability for collecting, organizing, reporting, and using the

measures to improve their KM programs and drive funding and investment. In addition, a measurement system that captures intangible benefits such as social cohesion, job satisfaction, and time-to-competency will provide a broader perspective of successful KM efforts.

Step No. 5: In addition to metrics, provide compelling examples of success.

At every stage of KM deployment, organizations need examples of success that can help justify past and future investments and provide management with a vision of what is possible. Collect success stories that illustrate the value path from inputs to outcomes.

Although measurement has inherent esthetic and social value, its utilization value comes when it propels one from point A to point B--from ignorance to understanding or informed action. A measurement system that links KM activities to business impact provides a rationale for investment beyond the esthetics and intangibles that KM brings to an organization.

5.3 Description of the system

5.3.1 Mixed language characters:

We start by mapping all greek characters to the latin keyboard

We convert all strings to upper case

We convert all strings based on the mapping above by removing annotation

We convert backwards this time from latin to greek characters.

Exception: We exclude from the above process customers with passport numbers and strings that contain characters unique to latin keyboard (J,W,Q,C,F,L,S,U,V)

The objective of the above process is to cleanse all characters of string fields in one of two character types (all greek or all latin).

5.3.2 Inconsistency of data format:

Will be applied in certain fields (such as ID card number) where clear logical rules will be applied for data cleansing.

For example 0 will be replaced to null or empty string/remove any non-alphanumeric character or replace it with another alphanumeric.

5.3.3 End user mistakes:

In numeric fields such as IRS number strict validation rules will be applied.

In alphanumeric fields, spelling or anagram mistakes can be countered by phonetic algorithms.

Soundex is a phonetic algorithm for indexing names by sound, as pronounced in English. The goal is for homophones to be encoded to the same representation so that they can be matched despite minor differences in spelling. The algorithm mainly encodes consonants; a vowel will not be encoded unless it is the first letter. Soundex is the most widely known of all phonetic algorithms, as it is a standard feature of MS SQL and Oracle, and is often used (incorrectly) as a synonym for "phonetic algorithm". Improvements to Soundex are the basis for many modern phonetic algorithms

The above is very important in a contact center because the majority of contacts offered are through the phone.

5.3.4 Non up-to-date data

The approach to this is vector space model (Chapter 3)

Vector space model (or term vector model) is an algebraic model for representing text documents (and any objects, in general) as vectors of identifiers, such as, for example, index terms. It is used in information filtering, information retrieval, indexing and relevancy rankings. Its first use was in the SMART Information Retrieval System.

5.3.5 Abbreviations/Partial strings

Is probably solved through stemming as the goal is to match strings based on their semantic information.

Ex. THEO should match THEODORE and THEODOR

5.3.6 Lack of validation rules vs. user friendly interface

Is solved only by a strategic decision that should take into account the levels of digital literacy within the organization and the technological maturity of the end users.

Key issues

In a credit card search a customer registered as PATSILINAKOS THEO cannot be found if the criteria used are PATSILINAKOS THEODORE

5.4 Architecture and interface

The Microsoft enterprise application development platform is a comprehensive platform for building connected systems based on the .NET Framework. The .NET Framework provides a cohesive and comprehensive development environment, as well as a core runtime that enables effective deployment, operations and management of enterprise applications. The .NET Framework and the Microsoft enterprise application development platform, however, were not designed for a homogeneous enterprise. Microsoft based its application platform strategy on the support for industry standards and the understanding that enterprises need the ability to integrate with existing infrastructure and applications, and may desire to select elements of their platform from other vendors but have them easily integrate with .NET-based applications and packaged products from Microsoft. .NET

fully enables Web Services, which are fundamentally designed around pluggable services that allow interoperability with other systems. This strategy not only allows companies to create robust, scalable and interoperable applications using current technologies, but positions users of the Microsoft enterprise application development platform to build service-oriented applications today while preserving their investments in existing platforms.

5.4.1 The .NET Framework Class Libraries

The .NET Framework class library is a collection of reusable services that tightly integrate with the common language runtime. The class library is object-oriented, providing types from which your own managed code can derive functionality. This not only makes the .NET Framework types easy to use, but also reduces the time associated with learning new features of the .NET Framework. In addition, third-party components can integrate seamlessly with classes in the .NET Framework, and the Framework can be extended by ISVs and corporations with custom-developed functionality. For example, the .NET Framework Windows Form Controls implements a set of interfaces that developers can use to develop their own Windows Form Controls. These controls will blend seamlessly with the controls already supplied in the .NET Framework.

As one would expect from an object-oriented class library, the .NET Framework types enable developers to accomplish a range of common programming tasks, including tasks such as string management, data collection, database connectivity, and file access. In addition to these common tasks, the class library includes types that support a variety of specialized development scenarios. For example, you can use the .NET Framework to develop the following types of applications and services:

- Console applications
- Windows graphical user interface (GUI) applications (Windows Forms)
- ASP.NET Web applications
- Web services based on the SOAP industry standard

- Windows services

5.4.2 Building platform

The platform on which future service-oriented applications will be built requires extensive support for current Internet standards and, more importantly, emerging XML Web Services standards. Visual Studio is built from the ground up to enable integration through XML Web services. By allowing applications to share data using Internet standards and protocols, XML Web services enable developers to assemble applications from new and existing code, regardless of platform, programming language, or object model. The rich Web Services support makes Visual Studio an excellent tool for integrating services and components written in other languages, such as J2EE.

5.5 SoundEx Algorithm

The Soundex code for a name consists of a letter followed by three numerical digits: the letter is the first letter of the name, and the digits encode the remaining consonants. Similar sounding consonants share the same digit so, for example, the labial consonants B, F, P, and V are each encoded as the number 1. Vowels can affect the coding, but are not coded themselves except as the first letter. However if "h" or "w" separate two consonants that have the same soundex code, the consonant to the right of the vowel is not coded.

The correct value can be found as follows:

1. The first letter of the name is the letter of the Soundex code, and is not coded to a number.
2. Replace consonants with digits as follows (after the first letter):
   - 2.1. b, f, p, v => 1
   - 2.2. c, g, j, k, q, s, x, z => 2
   - 2.3. d, t => 3

2.4. l => 4

2.5. m, n => 5

2.6. r => 6

2.7. h, w are not coded

3. Two adjacent letters with the same number are coded as a single number. Letters with the same number separated by an h or w are also coded as a single number.

4. Continue until you have one letter and three numbers. If you run out of letters, fill in 0s until there are three numbers.

5.6 System Architecture

In the following figure is presented the architecture of the proposed system

Figure 6 - Proposed System Architecture

One of the main advantages of the proposed system from a security point of view is that it does not interfere with existing security and privileges of any of the involved systems. The end user does not receive any information that was not allowed to view prior to the installation of this middleware.

5.7 Operation and results

Requirements

Administrator privileges to the application server and database server

One read-only user per database involved in the system

.NET Framework 3.0 installed to the clients

Installed web browser to the clients

System roles

Administrator: Performs the entire initial configuration, adds new users and performs all the required maintenance to the proposed system.

User: Is any user allowed to perform unified searches.

After the initial setup process the application *administrator* role is performing the initial configuration of the system. This involves active connections with the databases and also an initial mapping of the database fields with the fields in the unified database (Figure 7).

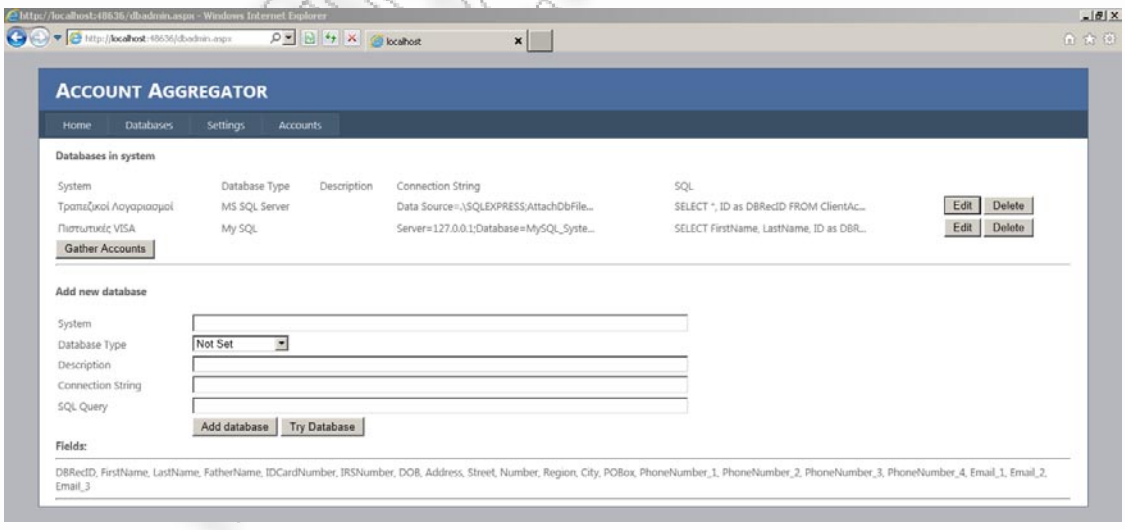

Figure 7 - Configuration Page

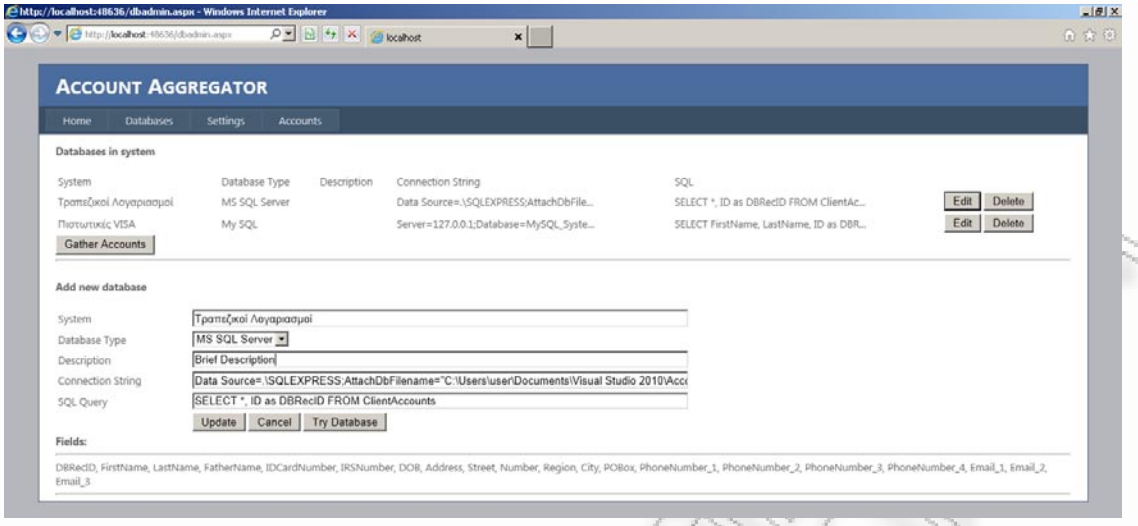

Figure 8 - New database connection Page

The administrator role can schedule the retrieval process of distributed data to the aggregated table. System administrators can also configure the sensitivity of the models used by adjusting the percentage of the threshold. The administrator or the business owner can also configure the fields where matching record will trigger a query in the related fields

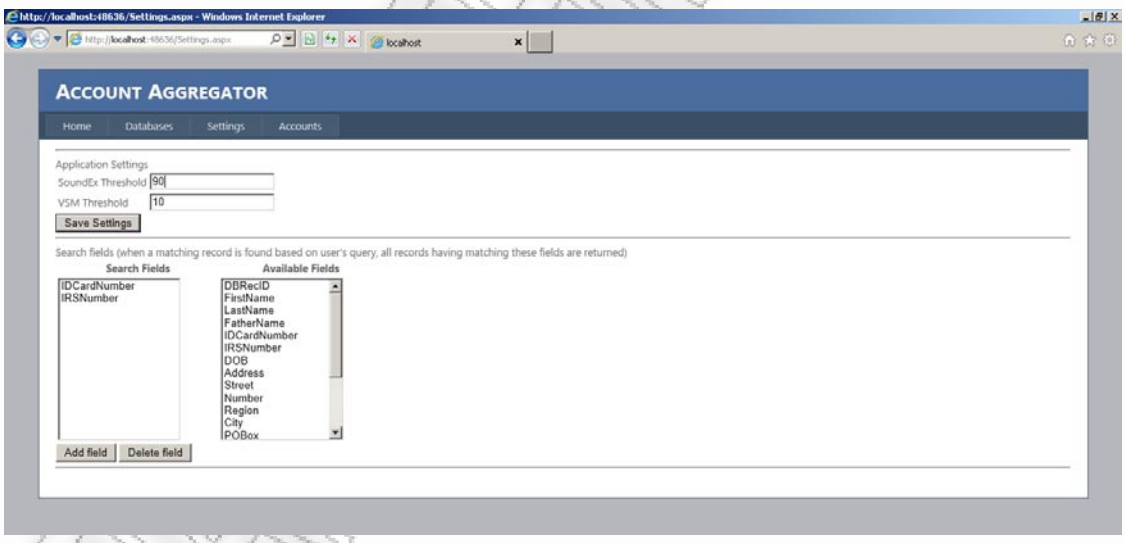

Figure 9 - Configuration Page

The *user* role can then login to the system and perform search. The user interface (UI) of the application is quite simple, as the user fills the respective fields with all possible information collected by the customer (Figure 10, next page). The *user* can also provide feedback after each search in order to

increase the success rate of the future requests, simply by “associating” two or more of the results.

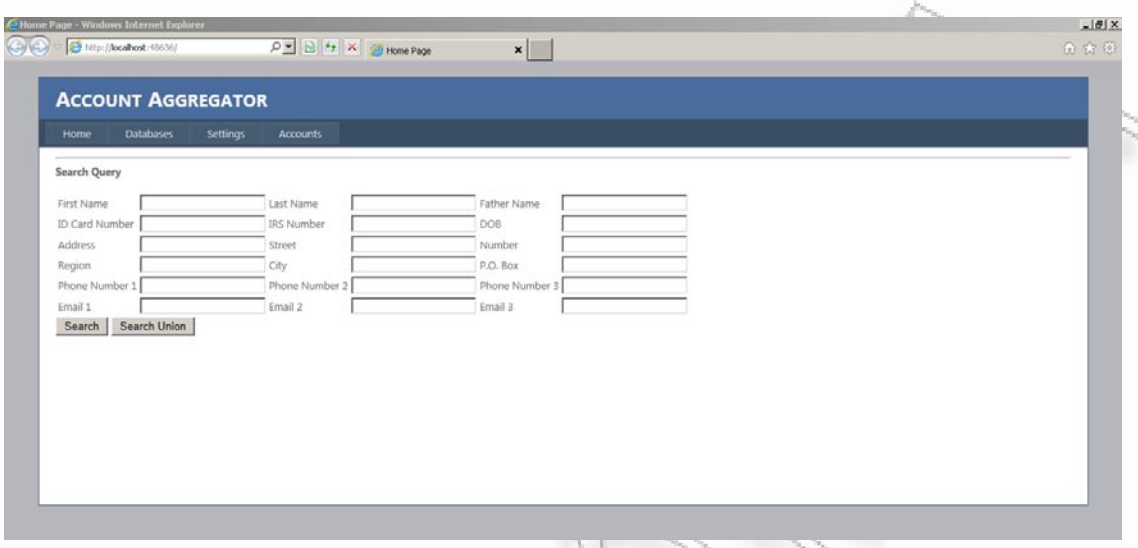

Figure 10 - Initial startup Page

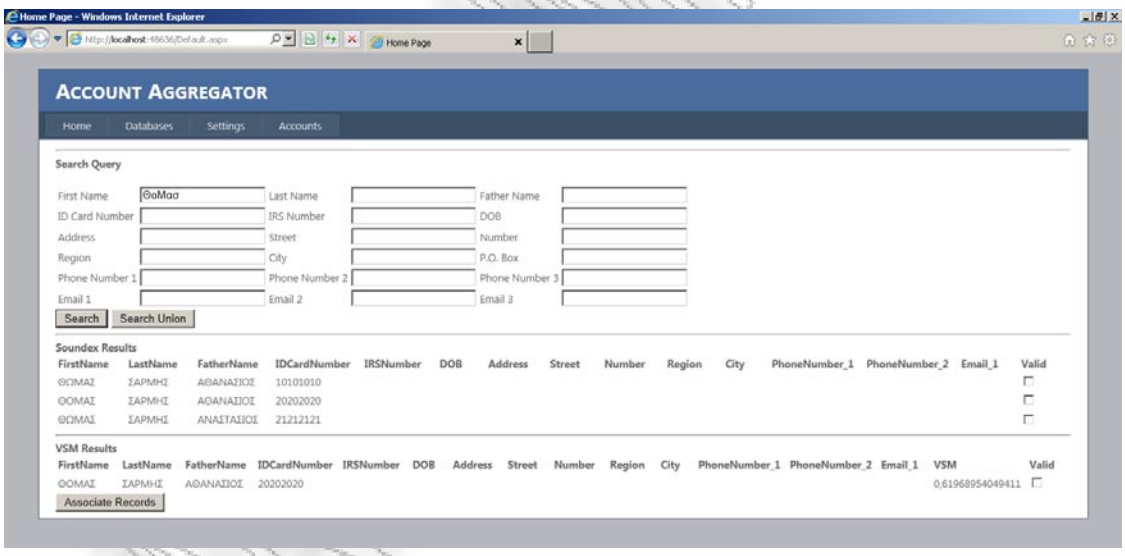

Figure 11 - Sample results

For the sole purpose of this thesis a user friendly tab was created in order to add new records to the connected databases (Figure 12, next page). This was implemented for assistance during test/debugging phase.

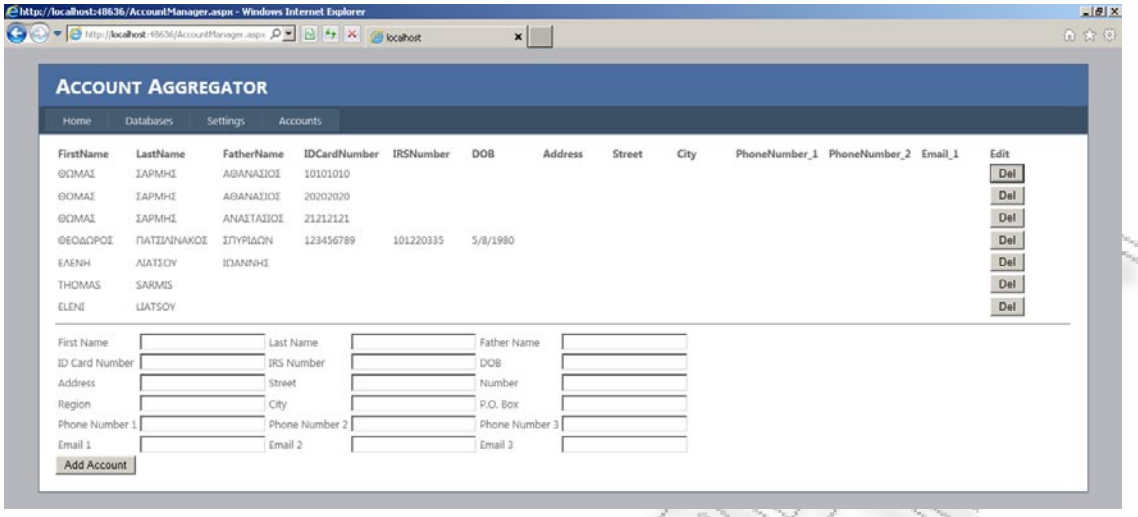

Figure 12 - Test accounts input Page

The proposed system was tested in a live environment of a major Greek bank. Due to the sensitive information regarding the clientele of the organization, real data are not presented in this thesis.

In order to emulate a live environment, a subset of sample records was created following the distribution of the issues described at 5.3. The percentages of this distribution are shown in Table 2

Table 2 - Sample Dataset

		Database #01	**Database #02**
		Debit Cards Holders	**Credit Cards Holders**
	Total Records Count	103.562	93.461
	Common records (based in unique bank customer number)	62.768	
a	Mixed characters	2.536	852
b	Obsolete data	24.913	18.751
c	Data entry errors (estimation 1% of the total)	1.035	934

d	Abbrevations	13.578	18.645

5.8 Measurement & KPIs

Many different measures for evaluating the performance of information retrieval systems have been proposed. The measures require a collection of documents and a query. All common measures described here assume a ground truth notion of relevancy: every document is known to be either relevant or non-relevant to a particular query. In practice queries may be ill-posed and there may be different shades of relevancy.

5.8.1 Precision

Precision is the fraction of the documents retrieved that are relevant to the user's information need.

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

In binary classification, precision is analogous to positive predictive value. Precision takes all retrieved documents into account. It can also be evaluated at a given cut-off rank, considering only the topmost results returned by the system. This measure is called precision at n or P@n.

Note that the meaning and usage of "precision" in the field of Information Retrieval differs from the definition of accuracy and precision within other branches of science and technology.

5.8.2 Recall

Recall is the fraction of the documents that are relevant to the query that are successfully retrieved.

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

In binary classification, recall is called sensitivity. So it can be looked at as the probability that a relevant document is retrieved by the query.

It is trivial to achieve recall of 100% by returning all documents in response to any query. Therefore recall alone is not enough but one needs to measure the number of non-relevant documents also, for example by computing the precision.

5.8.3 Fall-Out

The proportion of non-relevant documents that are retrieved, out of all non-relevant documents available:

$$\text{fall-out} = \frac{|\{\text{non-relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{non-relevant documents}\}|}$$

In binary classification, fall-out is closely related to specificity (1 − specificity). It can be looked at as the probability that a non-relevant document is retrieved by the query.

It is trivial to achieve fall-out of 0% by returning zero documents in response to any query.

5.8.4 F-measure

The weighted harmonic mean of precision and recall, the traditional F-measure or balanced F-score is:

$$F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{(\text{precision} + \text{recall})}.$$

This is also known as the F1 measure, because recall and precision are evenly weighted.

The general formula for non-negative real β is:

$$F_\beta = \frac{(1+\beta^2)\cdot(\mathrm{precision}\cdot\mathrm{recall})}{(\beta^2\cdot\mathrm{precision}+\mathrm{recall})}.$$

Two other commonly used F measures are the F2 measure, which weights recall twice as much as precision, and the F0.5 measure, which weights precision twice as much as recall.

The F-measure was derived by van Rijsbergen (1979) so that Fβ "measures the effectiveness of retrieval with respect to a user who attaches β times as much importance to recall as precision". It is based on van Rijsbergen's effectiveness measure.

$$E = 1 - \frac{1}{\frac{\alpha}{P} + \frac{1-\alpha}{R}}$$

Their relationship is Fβ = 1 – E, where $\alpha = \frac{1}{1+\beta^2}$.

5.8.5 Average precision

Precision and recall are single-value metrics based on the whole list of documents returned by the system. For systems that return a ranked sequence of documents, it is desirable to also consider the order in which the returned documents are presented. Average precision emphasizes ranking relevant documents higher. It is the average of precisions computed at the point of each of the relevant documents in the ranked sequence:

$$\mathrm{AveP} = \frac{\sum_{r=1}^{N}(P(r)\times \mathrm{rel}(r))}{\text{number of relevant documents}}$$

Where r is the rank, N the number retrieved, rel() a binary function on the relevance of a given rank, and P(r) precision at a given cut-off rank:

$$\mathrm{P(r)} = \frac{|\{\text{relevant retrieved documents of rank r or less}\}|}{r}$$

This metric is also sometimes referred to geometrically as the area under the Precision-Recall curve.

Note that the denominator (number of relevant documents) is the number of relevant documents in the entire collection, so that the metric reflects performance over all relevant documents, regardless of a retrieval cutoff.

5.8.6 R-Precision

Precision at R-th position in the ranking of results for a query that has R relevant documents. This measure is highly correlated to Average Precision.

5.8.7 Mean average precision

Mean average precision for a set of queries is the mean of the average precision scores for each query.

$$\mathrm{MAP} = \frac{\sum_{q=1}^{Q} \mathrm{AveP(q)}}{Q}$$

Where Q is the number of queries.

5.8.8 Discounted cumulative gain

DCG uses a graded relevance scale of documents from the result set to evaluate the usefulness, or gain, of a document based on its position in the result list. The premise of DCG is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result.

The DCG accumulated at a particular rank position p is defined as:

$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}.$$

Since result set may vary in size among different queries or systems, to compare performances the normalized version of DCG uses an ideal DCG. To this end, it sorts documents of a result list by relevance, producing an ideal DCG at position p (IDCGp), which normalizes the score:

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCGp}.$$

The nDCG values for all queries can be averaged to obtain a measure of the average performance of a ranking algorithm. Note that in a perfect ranking algorithm, the DCGp will be the same as the IDCGp producing an nDCG of 1.0. All nDCG calculations are then relative values on the interval 0.0 to 1.0 and so are cross-query comparable.

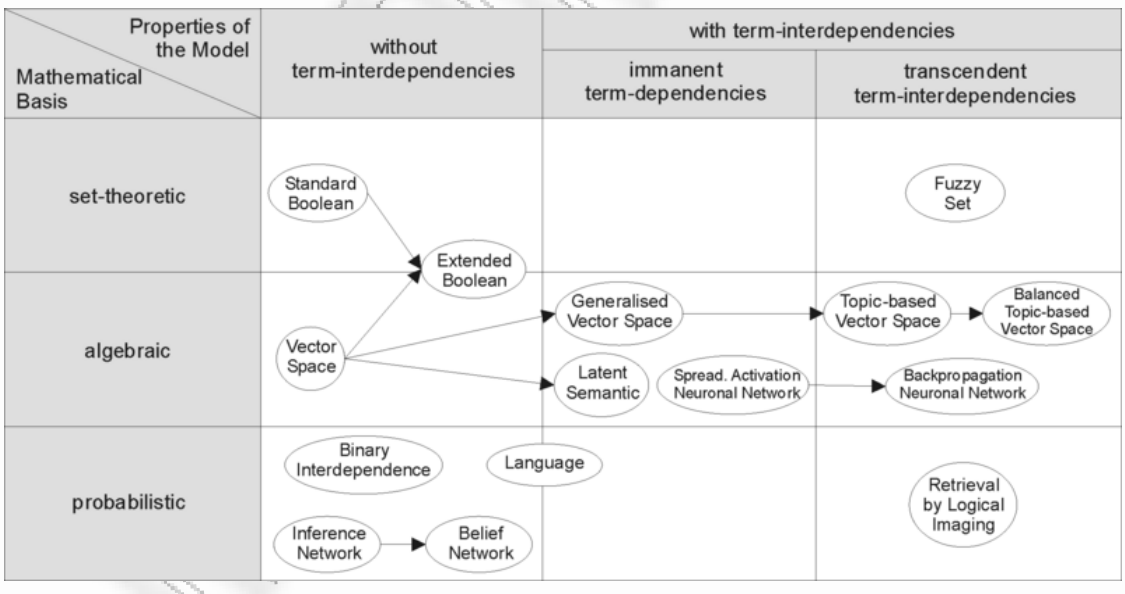

Figure 13 - Categorization of IR-models (src: ISBN 978-3-8325-0514-1)

6 References

Addicott Rachael, McGivern Gerry and Ferlie Ewan Networks, Organizational Learning and Knowledge Management: NHS Cancer Networks [Article] // Public Money & Management . - 2006. - 26. - pp. 87-94. - doi:10.1111/j.1467-9302.2006.00506.x.

Alavi Maryam and Leidner Dorothy E. Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues [Report] : Review. - [s.l.] : MIS Quarterly, 2001. - pp. 107-136. - 25.

Alavi Maryam and Leidner Dorothy E. Knowledge management systems: issues, challenges, and benefits [Article] // Communications of the AIS. - 1999.

Andrews K. M. and Delahaye B. L. Influences on knowledge processes on organisational learning: The psychosocial filter [Article] // Journal of Management Studies. - 2000. - 6 : Vol. 37.

APQC [Online] // APQC website. - KM Edge. - 2011. - http://kmedge.org/features/fivetips.html.

Argote Linda, McEvily Bill and Reagans Ray Managing Knowledge in Organizations: An Integrative Framework and Review of Emerging Themes [Article] // MANAGEMENT SCIENCE. - April 4, 2003. - 4 : Vol. 49. - pp. 571-582.

Dierkes Meinolf [et al.] Handbook of organizational learning and knowledge [Book]. - [s.l.] : Oxford University Press, 2003. - p. 979. - 0-19-829583-9.

Efron Miles [et al.] Machine Learning for Information Architecture [Conference]. - [s.l.] : Proceedings of the ACM & IEEE Joint Conference on Digital Libraries (JCDL 2004), 2004.

Garvin A. D. Building a learning organization [Book Section] // Harvard Business Review. - [s.l.] : Harvard University, 1993. - Vol. 71. - 4.

Goldsmith Marshall, Morgan Howard and Ogg Alexander J. Leading Organizational Learning: Harnessing the Power of Knowledge [Book]. - 2003.

Grantner Emily ISO 8000 - A Standard for Data Quality [Online] // http://findarticles.com/. - Logistics Spectrum, October 29, 2010. - http://findarticles.com/p/articles/mi_qa3766/is_200710/ai_n27997242/.

Grisham Consulting [Online] // Grisham Consulting. - Novemeber 2006. - 2011. - http://www.thomasgrisham.com/attachments/File/CIB_Dubai_Knowledge_Management.pdf.

Hyde A. C. and Mitchell K. Knowledge Management: The Next Big Thing [Article] // The Public Manager. - 2000. - Vol. 29.

Inmon William and Nesavich Anthony Tapping into unstructured data: integrating unstructured data and textual analytics into business intelligence [Book]. - [s.l.] : Prentice Hall Press, 2007. - 9780137136889.

Jiawei Han & Micheline Kamber Data-Mining Concepts and Techniques Solution Manual [Online] // http://www.scribd.com/. - University of Illinois, 2006. - http://www.scribd.com/doc/11712189/Data-Mining-Concepts-and-Techniques-Solution-Manual.

Kogut B. and Zander U. Knowledge of the firm, combinative capabilities, and the replication of technology [Article] // Organization Science. - 1992. - 3 : Vol. 3. - pp. 383-397.

Nichols M. D. [et al.] DEBORA: Developing an interface to support collaboration in a digital library [Conference] // Proceedings of the Fourth European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2000). - [s.l.] : Springer, 2000. - pp. 239-248.

Nonaka Ikujiro and Takeuchi Hirotaka The knowledge creating company: how Japanese companies create the dynamics of innovation [Book]. - [s.l.] : New York: Oxford University Press, 1995. - ISBN 9780195092691.

Nonaka Ikujiro and von Krogh Georg Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory [Article] // Organization Science. - 2009. - 3 : Vol. 20. - pp. 635–652.

Nonaka Ikujiro The knowledge-creating company [Article] // Harvard Business Review. - November-December 1991. - 69. - pp. 96-104.

O'Brien A. J. and Marakas G. M. Management information systems [Book]. - Boston : McGraw-Hill/Irwin, 2009. - 9th.

Reinhardt R. [et al.] Intellectual Capital and Knowledge Management [Book]. - 2003.

Robertson James 10 principles of effective information management. - Broadway : Step Two Designs Pty Ltd, November 1st, 2005. - http://www.steptwo.com.au/files/kmc_effectiveim.pdf.

Rumizen Melissie Clemmons The Complete Idiot's Guide to Knowledge Management [Book]. - [s.l.] : Alpha, 2002. - p. 315. - 978-0028641775.

Salton G., Wong A. and Yang C. S. A vector space model for automatic indexing [Article] // Communications of the ACM. - Nov. 1975. - 11 : Vol. 18. - pp. 613-620.

Sensky Tom Knowledge Management [Article] // Advances in Psychiatric Treatment. - 2002. - 5 : Vol. 8. - pp. 387–395.

Serenko Alexander and Bontis Nick Meta-review of knowledge management and intellectual capital literature: citation impact and research productivity rankings [Article] // Knowledge and Process Management. - 2004. - 3 : Vol. 11. - pp. 185–198.

Spender J.-C. and Scherer G. A. The Philosophical Foundations of Knowledge Management: Editors' Introduction [Conference]. - [s.l.] : Organization, 2007. - Vol. 14. - pp. 5-28. - Available at SSRN: http://ssrn.com/abstract=958768.

Spender J.-C. Making Knowledge the Basis of a Dynamic Theory of the Firm [Article] // Strategic Management Journal. - 1996. - Vol. 17. - pp. 45-62.

Ståhle P. and Grönroos , M. Dynamic Intellectual Capital: Knowledge Management in Theory and Practice [Conference] // WSOY. - Helsinki : [s.n.], 2000.

Trompenaars Fons and Hampden-Turner Charles Managing People Across Cultures (Culture for Business Series) [Book]. - 2004.

Walker Derek H. T. [et al.] Frameworks For Knowledge Management Initiatives In The Field Of Project Management-Using Metaphor for Improved Visibility [Conference]. - Dubai : [s.n.], November 26th -29th 2006.

Wheatley M. and Kellner-Rogers M. A simpler way [Book]. - San Francisco : Berrett-Koehler Publishers, 1999.

Wheatley M. Leadership and the new science [Book]. - San Francisco : Berret-Koehler Publishers, 1999.

I. Appendix – Description of classes

Table 3 - Appendix I, Description of Classes

Class / Description	Method/Field description
Vector Space Model	*RunVSM(datatable, NameValueCollection, Double)* returns a DataTable objext
SoundEx	
SoundEx is executed every time a new account is aggregated in the application master database	*SoundEx (string, integer)* function returns a four-digit integer that represents the soundex encoding for this string
	CompareSoundex(aSoundEx As String, bSoundEx As String) returns a double
Other	
	AssociateAccounts(List(Of Integer))
	CreateConnection(DataBaseType, String) Creates a new database connection and the string is the connection string to the database. dotNet supports natively SQL, MySQL and Oracle. There is an adapter for DB2 connection

II. Appendix – Code Samples

a. References

```
Imports System.Data.SqlClient
Imports System.Data.Common
Imports MySql.Data.MySqlClient
Imports System.Collections.Specialized
```

b. SoundEx Algorithm

```
Public Shared Function SoundEx(ByVal Word As String, ByVal Length As
Integer) As String
        ' Value to return
        Dim Value As String = ""
        ' Size of the word to process
        Dim Size As Integer = Word.Length
        ' Make sure the word is at least two characters in length
        If (Size > 1) Then
            ' Convert the word to all uppercase
            Word = Word.ToUpper()
            ' Conver to the word to a character array for faster processing
            Dim Chars() As Char = Word.ToCharArray()
            ' Buffer to build up with character codes
            Dim Buffer As New System.Text.StringBuilder
            Buffer.Length = 0
            ' The current and previous character codes
            Dim PrevCode As Integer = 0
            Dim CurrCode As Integer = 0
            ' Append the first character to the buffer
            Buffer.Append(Chars(0))
            ' Prepare variables for loop
            Dim i As Integer
            Dim LoopLimit As Integer = Size - 1
' Loop through all the characters and convert them to the proper character
code
            For i = 1 To LoopLimit
                Select Case Chars(i)
                    Case "A", "E", "I", "O", "U", "H", "W", "Y"
                        CurrCode = 0
                    Case "B", "F", "P", "V"
                        CurrCode = 1
                    Case "C", "G", "J", "K", "Q", "S", "X", "Z"
                        CurrCode = 2
                    Case "D", "T"
                        CurrCode = 3
                    Case "L"
                        CurrCode = 4
                    Case "M", "N"
                        CurrCode = 5
                    Case "R"
                        CurrCode = 6
```

```
                End Select
' Check to see if the current code is the same as the last one
                If (CurrCode <> PrevCode) Then
' Check to see if the current code is 0 (a vowel); do not proceed
                    If (CurrCode <> 0) Then
                        Buffer.Append(CurrCode)
                    End If
                End If
' If the buffer size meets the length limit, then exit the loop
                If (Buffer.Length = Length) Then
                    Exit For
                End If
            Next
            ' Padd the buffer if required
            Size = Buffer.Length
            If (Size < Length) Then
                Buffer.Append("0", (Length - Size))
            End If
            ' Set the return value
            Value = Buffer.ToString()
        End If
       ' Return the computed soundex
       Return Value
End Function
```

c. RunSoundEx

```
   Public Function RunSoundEx(ByRef SearchTerms As NameValueCollection,
ByVal Threshold As Double) As DataTable
        Dim SE As New Soundex
        Dim CS As New ConvertString
        Dim ResultTable As DataTable = Nothing
        Dim MatchingRecordIDs As List(Of Int32) = New List(Of Int32)
        Dim SearchTermsPhonetic As NameValueCollection = New
NameValueCollection()

        ' Convert the search terms to their phonetic representation.
        For Each Key In SearchTerms.Keys
            Dim strSoundEx = ""
            If  Not String.IsNullOrEmpty(SearchTerms(Key).ToString()) then
strSoundEx = SE.GetSoundex (CS.LatinLetters(SearchTerms(Key).ToString()))
            SearchTermsPhonetic.Add(Key, strSoundEx)
        Next

        ' Load the Phonetic Table and the Normal Table
        Dim AccountsPhonetic = ReadTable(1,
MainDBConnection.ConnectionString, New SqlCommand("SELECT * FROM
AccountsPhonetic"))
        Dim Accounts = ReadTable(1, MainDBConnection.ConnectionString, New
SqlCommand("SELECT * FROM Accounts"))


        ' Search in the Phonetic Table
        For Each Row As DataRow In AccountsPhonetic.Rows
            For Each Key As String In SearchTermsPhonetic
                If Not SearchTermsPhonetic(Key) = "00000000"
```

```
                    If Not ( String.IsNullOrEmpty(Row(Key).ToString())  or
String.IsNullOrEmpty(SearchTermsPhonetic(Key)) )
                        If CompareSoundex(SearchTermsPhonetic(Key),
Row(Key)) > SoundExThreshold then
                            MatchingRecordIDs.Add(Row("ID"))
                        End If
                    End If
                End IF
            Next
        Next

        ' Search in the Normal Table
        For Each Row As DataRow In Accounts.Rows
            For Each Key As String In SearchTerms
                If Not String.IsNullOrEmpty(SearchTerms(Key))
                    If String.Compare (SearchTerms(Key),
ConvertToUCase(Row(Key).ToString()), true) = 0 then
                        MatchingRecordIDs.Add(Row("ID"))
                    End If
                End If
            Next
        Next

        ' Load the Matching records (Soundex/String Comparison) to a table
        If MatchingRecordIDs.Count > 0 then
            Dim CommandText As String = "SELECT * FROM Accounts WHERE ID IN
(" + String.Join(", ",MatchingRecordIDs) + ");"
            ResultTable = ReadTable(1, MainDBConnection.ConnectionString,
New SqlCommand(CommandText))
        End If

        Return ResultTable
    End Function
```

d. Compare SoundEx

```
    Private Function CompareSoundex(aSoundEx As String, bSoundEx As String)
As Double
        Dim Matches As Integer

        Dim L = aSoundEx.Length()
        If bSoundEx.Length < L then L = bSoundEx.Length()

        For i As Integer = 0 To L - 1
            If aSoundEx(i) = bSoundEx(i) Then
                Matches += 1
            End If
        Next

        Return Matches * (100 / L) ' / 12.5
    End Function

    Public Function SearchRecords(SearchTerms As NameValueCollection,
VSMOnSoundex As Boolean) As ComparisonResults
        Dim Result = New ComparisonResults()
```

```
        Result.SoundExResults = RunSoundEx(SearchTerms, SoundExThreshold)

        Dim Cmd As SqlCommand = New SqlCommand()

        Dim MatchingRecordIDs = New List(Of String)
        Dim Table As DataTable
        For Each Row As DataRow In Result.SoundExResults.Rows
          For Each FieldName In SearchFields
            If Not  String.IsNullOrEmpty(Row(FieldName).ToString())
                Cmd.CommandText = String.Format("SELECT * FROM Accounts
WHERE {0}=@Param", FieldName)
                Cmd.Parameters.Clear
                Cmd.Parameters.Add("@Param", SqlDbType.NVarChar).Value =
Row(FieldName).ToString()

                Table = ReadTable(1, MainDBConnection.ConnectionString,
Cmd)
                For Each ResultRow As DataRow In Table.Rows
                     MatchingRecordIDs.Add(ResultRow("ID"))
                Next
            End If
          Next
        Next

        Dim CommandText As String = "SELECT * FROM Accounts WHERE ID IN ("
+ String.Join(", ",MatchingRecordIDs) + ");"
        Table = ReadTable(1, MainDBConnection.ConnectionString, New
SqlCommand(CommandText))

        SetPrimaryKeyColumn(Result.SoundExResults, "ID")
        For Each Row As DataRow In Table.Rows
            If Result.SoundExResults.Rows.Find(Row("ID")) Is Nothing
                Result.SoundExResults.ImportRow(Row)
            End If
        Next

        If VSMOnSoundex then
            Result.VSMResults = RunVSM(Result.SoundExResults, SearchTerms,
-1)
        Else
            Dim Data = ReadTable(1, MainDBConnection.ConnectionString, New
SqlCommand("SELECT * FROM Accounts;"))
            Result.VSMResults = RunVSM(Data, SearchTerms, VSMThreshold/100)
        End If

        SetPrimaryKeyColumn(Result.VSMResults, "ID")

        'Can I augment the list of resuls here ?

        AddAssociatedAccounts(Result.VSMResults)
        AddAssociatedAccounts(Result.SoundExResults)

        Return Result
    End Function
```

e. RunVSM

```
    Public Function RunVSM(ByRef Data As DataTable, ByRef SearchTerms As
NameValueCollection, ByVal Threshold As Double) As DataTable
        ' Prepare the VSM Parameters...
        Dim ResultTable = Data.Clone

        Dim VSMArguments As String()
        ReDim VSMArguments(Data.Rows.Count)
        Dim ArgumentsIndex = 1

        ' Exclude System Columns
        Dim SystemColumns As String() = {"ID", "DBID", "DBRecID"}

        For Each Row As DataRow In Data.Rows
            For Each Column As DataColumn In Data.Columns
                If Not SystemColumns.Contains(Column.ColumnName)
                    VSMArguments(ArgumentsIndex) =
VSMArguments(ArgumentsIndex) + Row(Column.ColumnName).ToString() + " "
                End If
            Next
          ArgumentsIndex = ArgumentsIndex + 1
        Next

        For Each Key As String In SearchTerms
            VSMArguments(0) = VSMArguments(0) +
ConvertToUCase(SearchTerms(Key).ToString()) + " "
        Next

        For i As Integer = 0 to VSMArguments.Count - 1
            VSMArguments(i) = VSMArguments(i).Trim()
        Next

        Dim VSMResults As Dictionary(Of Double, String) = Nothing

        ' Run the VSM Comparison.

        Try
            VSMResults =  VectorSpaceModel.VSM.Compare(VSMArguments)

            ResultTable.Columns.Add("VSMRating",
Type.GetType("System.Double"))

            For Each Item In VSMResults
              If (Item.Key > Threshold) or (Item.Key <> Item.Key) ' Way to
go VB.Net Key <> Key means Key is not a number....
                ResultTable.ImportRow(Data.Rows(Item.Value-1))
                ResultTable.Rows(ResultTable.Rows.Count-1)("VSMRating") =
Item.Key
              End If
            Next

            'ResultTable = ResultTable.Select("ID > 0", "VSMRating DESC")
        Catch
            'Just silence the exception.
        End Try


        Return ResultTable
```

```
    End Function
```

f. AssociateAccounts

```
    Public Sub AssociateAccounts(ByRef AccountsID As List(Of Integer))
        Dim CommandText = "SELECT * FROM Associations WHERE AccountID IN
("+String.Join(",", AccountsID)+")"
        Dim Associations As DataTable = ReadTable(1,
MainDBConnection.ConnectionString, New SqlCommand(CommandText))
        Dim GroupIDs As List(Of Integer) = New List(Of Integer)
        Dim AccountIDsToAdd As List(Of Integer) = New List(Of Integer)

        AccountIDsToAdd = AccountsID

        For Each Row As DataRow In Associations.Rows
          If Not GroupIDs.Contains(Row("GroupID")) then
            GroupIDs.Add(Row("GroupID"))
          End If
          AccountIDsToAdd.Remove(Row("AccountID"))
        Next

        Dim GroupID As Integer = -1

        If (GroupIDs.Count > 0) then
            GroupID = GroupIDs(0)
            CommandText = "UPDATE Associations SET
GroupID="+GroupID.ToString()+" WHERE GroupID in ("+String.Join(",",
GroupIDs)+")"
            ExecQuery(MainDBConnection.ConnectionString, New
SqlCommand(CommandText))
        Else
            Dim Cmd = New SqlCommand()
            Cmd.Connection = MainDBConnection
            Cmd.Connection.Open
            Cmd.CommandText = "SELECT MAX(ID) FROM Associations;"
            Try
              GroupID = Cmd.ExecuteScalar
            Catch
              GroupID = 1
            End Try
            GroupID = GroupID + 1
            Cmd.Connection.Close
        End If

        For Each AccountID As Integer In AccountIDsToAdd
            CommandText = String.Format("INSERT INTO Associations (GroupID,
AccountID) VALUES ({0}, {1});", GroupID, AccountID)
            ExecQuery(MainDBConnection.ConnectionString, New
SqlCommand(CommandText))
        Next

    End Sub


    Public Sub AddAssociatedAccounts(ByRef aDataTable As DataTable)
```

```
        Dim AccountIDs = New List(Of Integer)
        For Each Row As DataRow In aDataTable.Rows
          AccountIDs.Add(Row("ID"))
        Next

        Dim CommandText As String
        CommandText = "SELECT * FROM Associations WHERE AccountID in (" +
String.Join(",", AccountIDs) + ");"
        Dim AssociationsTable = ReadTable(1,
MainDBConnection.ConnectionString, New SqlCommand(CommandText))

        Dim GroupIDs As List(Of Integer) = New List(Of Integer)
        For Each Row As DataRow In AssociationsTable.Rows
          Dim GroupID As Integer = Row("GroupID")
          If Not GroupIDs.Contains(GroupID) then GroupIDs.Add(GroupID)
        Next

        If GroupIDs.Count > 0 then
            CommandText ="SELECT Accounts.* FROM Accounts INNER JOIN
Associations ON Accounts.ID = Associations.AccountID WHERE
Associations.GroupID IN (" + String.Join(",", GroupIDs) + ");"
            Dim AssociatedAccounts = ReadTable(1,
MainDBConnection.ConnectionString, New SqlCommand(CommandText))
            ' Import Associated Accounts records in aDataTable...

            For Each Row As DataRow In AssociatedAccounts.Rows
                If aDataTable.Rows.Find(Row("ID")) Is Nothing then
aDataTable.ImportRow(Row)
            Next
       End If
    End Sub
```

g. Utility Functions

```
    'Database diversity Handling
    Enum DatabaseType
      Unknown = 0
      MSSQL = 1
      MySQL = 2
      Oracle = 3
    End Enum

    Public Function CreateConnection(aDatabaseType As DataBaseType,
aConnectionString As String) As DbConnection
        If aDatabaseType = DatabaseType.MSSQL then
          Return New SqlConnection(aConnectionString)
        Else If aDatabaseType = DatabaseType.MySQL then
          Return New MySqlConnection(aConnectionString)
        'Else If aDatabaseType = DatabaseType.Oracle  then
        '  Return New
        Else
           Throw New Exception("Invalid Database Type")
        End If
    End Function
```