Using Agreements as an Abstraction for Access Control Administration

by

André Reyneke

Using Agreements as an Abstraction for Access Control Administration

by

André Reyneke

Dissertation

submitted in fulfillment
of the requirements
for the degree

Magister Technologiae

in

Information Technology

in the

School of Information and Communication Technology

in the

Faculty of Engineering, the Built Environment and Information Technology

of the

Nelson Mandela Metropolitan University

Promoter: Prof. Reinhardt A. Botha

February 2007

Declaration

I, André Reyneke, hereby declare that:

- The work in this dissertation is my own work.
- All sources used or referred to have been documented and recognized.
- This dissertation has not previously been submitted in full or partial fulfillment of the requirements for an equivalent or higher qualification at any other recognized educational institute.

André Reyneke

Acknowledgements

My sincerest gratitude and appreciation are extended to:

- My loving wife Johanette for always being there; supporting and encouraging me when I needed it, and reminding me of my own strength during those long hours in front of the computer screen.
- My promoter, Professor Reinhardt Botha, for his astonishing wealth of knowledge which he made available to me so enthusiastically, patiently and constructively.
- Adéle Botha for proofreading my dissertation as well as providing language expertise.
- My family, who has continuously supported and encouraged me during my studies.
- All my friends for bearing with me.
- The Nelson Mandela Metropolitan University, particulary the School of ICT and the Institute for ICT Advancement, and the National Research Foundation for their administrative and financial support.
- To my God who have blessed me with the ability to complete my studies.

Abstract

The last couple of decades saw lots of changes in the business world. Not only did technology change at a rapid pace, but businesses' views with respect to the role that information plays also changed drastically. Information is now seen as a strategic resource. This change paved the way for the so-called knowledge worker that not only consumes information, but actively participates in creating new knowledge from information.

Employees must therefore be empowered to fulfill their new role as knowledge workers. Empowerment happens through job redefinition and by ensuring that the appropriate information is at hand. Although information is more readily available to employees, appropriate access controls must still be implemented. However, there is conflict between the need to share information and the need to keep information confidential.

These conflicting needs must be reflected in the administration of access control. In order to resolve these conflicts, a finer granularity of access controls must be implemented. However, to implement a finer granularity of access control, an increase in the number of access controls and, therefore, the administrative burden is inevitable.

Access control administrators must cater for a potentially large number of systems. These systems can not only be heterogenous as far as architecture and technology are concerned, but also with respect to access control paradigms. Vendors have realized that human involvement must be minimized, giving birth to so-called "provisioning systems". Provisioning systems, in principle, automate certain parts of access control administration. However, currently implementations are done in an ad hoc manner, that is, without a systematic process of identifying the real access control needs.

This study aims to address this problem by proposing the "agreement abstraction" as a possible vehicle for systematically analyzing the access control

requirements in a business. In essence, the agreement abstraction allows us to identify opportunities where access control can be automated.

A specific methodological approach is suggested whereby the business is analysed in terms of business processes, as opposed to the more traditional resource perspective. Various business processes are used as examples to explain and motivate the proposed agreement abstraction further.

This dissertation therefore contributes to the field of discourse by presenting a new abstraction that can be used systematically to analyse access control administration requirements.

Contents

List of Figures

Chapter 1

Introduction

The saying "change is the only constant" is true in many ways. It certainly also holds true for the world of business. In a constant bid to outwit and outperform their competitors, businesses are undergoing constant change. Restructuring, mergers and take-overs are rife.

It is not strange, therefore, that Eliasson (2005) notes that the way businesses are managed have changed considerably. However, it is not the management activities, per se, which have changed, but the way these activities are performed. Decision makers are required to make more accurate decisions faster than ever. Like predicted by Leavitt and Whisler (1958), management indeed makes extensive use of information technology to provide the relevant information.

Information is thus playing an increasingly important role in the business. Various sources of information exist: some sources are external to the business, but a huge amount of information also exist within a business. Having realized the strategic value of information, businesses are beginning to understand that they should nurture the information sources which reside within the business. This, however, requires a new type of employee; one tasked with using, creating and managing information. These employees are called knowledge workers.

Knowledge workers make extensive use of existing information to create new knowledge. It is therefore imperative that these employees are empowered through access to information. However, this is contradictory to traditional access control thinking, which tries to restrict possible access to information. Naturally, when the focus of a business is information and the

creation of knowledge from that information, this way of thinking is problematic.

This study is therefore primarily motivated by the realization that access to information should empower people.

1.1 Motivation for this study

However, arguing within the context of empowering people, several other realizations that motivate and support this study become evident.

The realization that controlling access does not mean keeping information away from people

The word "control" has a regulatory feeling about it. Merriam-Webster Online Dictionary (2004), in fact, defines the verb form as "to exercise restraining or directing influence over". However, further studying the Angle-French origin of the word – *contrerouler* – also indicates an "audit" aspect; this is further embodied in another interpretation of control: "to check, test, or verify by evidence or experiments" (Merriam-Webster Online Dictionary, 2004). Therefore, if we adopt a more audit-centric interpretation of control in "access control" the focus shifts from prevention or restraint to monitored use. This does not mean that prevention is not necessary for certain classes of information; however, it recognizes that a large body of information in businesses can be best utilized by reasonably freely allowing access, monitoring that access and keeping people accountable for what they do with that information. However, a large number of people and lots of information seem to indicate a lot of work, which is the essence of the next realization.

The realization that access control administration is a lot of work

The size of a business would obviously influence the complexity of a business' access control administration. However, if really small businesses are ignored for the time being it is not difficult to see that managing thousands of users' access to tens of thousands (if not millions) of business objects can become

quite a daunting task. For example, 1000 users and 100000 business objects result in 100000000 (100 million) possible relations to be managed. This is, obviously, no mean feat.

While abstractions such as roles, object classes and user groups can be used to ease the administration burden, it may only have limited effect. In the previous example if the number of relations could be reduced to 1% and only 10% of the 1% needs to be administered in a year, it still equates to 100000 relations to be maintained.

Additional abstractions, however, also require additional cognitive ability. Add on top of this that many access control activities require understanding of the application domain, a very high-level cognitive activity, the job of the access control administrator can become a very important (and difficult) job. In fact, in large organizations it is not uncommon to have several administrators. This, in turn, introduces some more coordination and adds to complexity and therefore probably also in the number of errors that inadvertently happen.

This directly influences the next realization: automation of administration is required.

The realization that automation of administration is required

To lessen the human factor in the administration of access control, consideration must be given to automation technology. Although the job of access control administrator may initially be complex, when it has been mastered it may also become rather repetitive and boring – yet another source of human error.

Repetitive tasks, however, are good for automation. Provisioning systems have come to the fore, essentially automating some access control administration tasks. Automation, intuitively, has two phases: administration and execution. Administration involves the specification of workflows and its constituent tasks. Identifying the repetitive tasks and abstracting them in such a way that they can be re-used effectively is a daunting task. Initial further investigation has shown that administration of provisioning systems happens in a fairly ad hoc manner.

The realization that administration happens in a fairly ad hoc manner

Previous work regarding access control administration issues dealt with reducing the number of relations to be managed (Sandhu & Munawer, 1999), applying access control principles to access control administration to assist with the separation of responsibility (Saltzer & Schroeder, 1975), and designing security administrative structures. However, little work has been done on automating access control administration.

A notable lack of literature on automated access control administration and provisioning systems exists. Even provisioning system vendors appear silent on implementation details. A personal communication with Warwick Metcalfe (Access control specialist, Synovation.com, 5 August 2004) confirmed that implementation of provisioning systems is happening in a fairly ad hoc manner.

It is therefore in this area, the administration of automated access control administration, that this dissertation aims to contribute. Consider, therefore, a description of the problem that this research project addresses.

1.2 Problem Statement

From the above the problem milieu can be characterized by two competing forces: knowledge workers that need access to information and business that wants to protect information. In this regard provisioning systems can be helpful. However the problem is that analysing the access control requirements happens in an ad hoc manner.

This domain of discourse will therefore be defined in terms of access control theory, administration activities and workflow automation, specifically provisioning systems. The domain of discourse is graphically depicted as a Venn-diagram in Figure 1.1.

The problem addressed by this research can therefore be summarized in terms of the research question: “How can access control requirements within the context of a provisioning system be determined?”

The next section sets more specific objectives in terms of this research question.

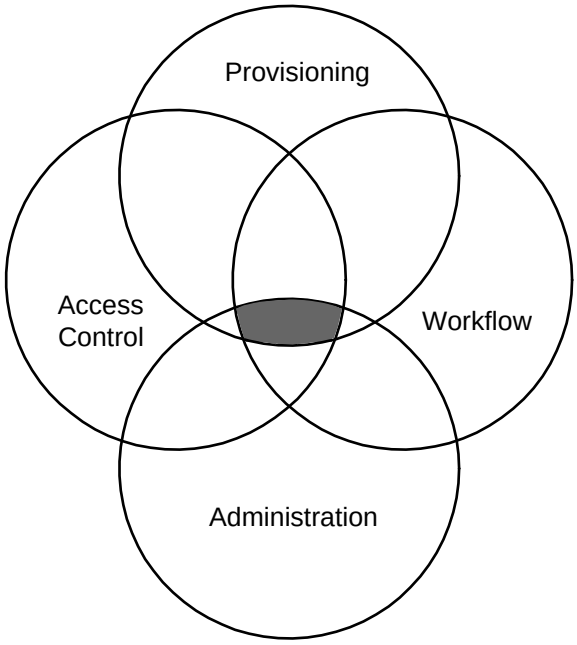

Figure 1.1: Domain Of Discourse

1.3 Objective

When answering questions about "how" something is to be done in the Information Technology (IT) milieu several approaches can be distinguished. Approaches could include, among others, formal methodologies, algorithms, best practices, modeling techniques and languages. All of these can be described as IT artifacts.

Hevner, March, and Park (2004) identify four categories of IT artifacts: constructs, methods, models and instantiations. Constructs represent the symbols and vocabulary that are dealt with, models include abstractions and representations, methods define algorithms, while practices and instantiations refer to implemented prototypes.

These categories are naturally related; instantiations support methods that are based on models that utilize constructs. The author sees this as an inverted pyramid, depicted in Figure 1.2. Several models may be described using a specific vocabulary; similarly the different methods could be based on the same model and one model could be instantiated in a number of ways.

While it may be necessary to explore the full range of artifacts to conclusively answer the research question, this research sets out to answer the

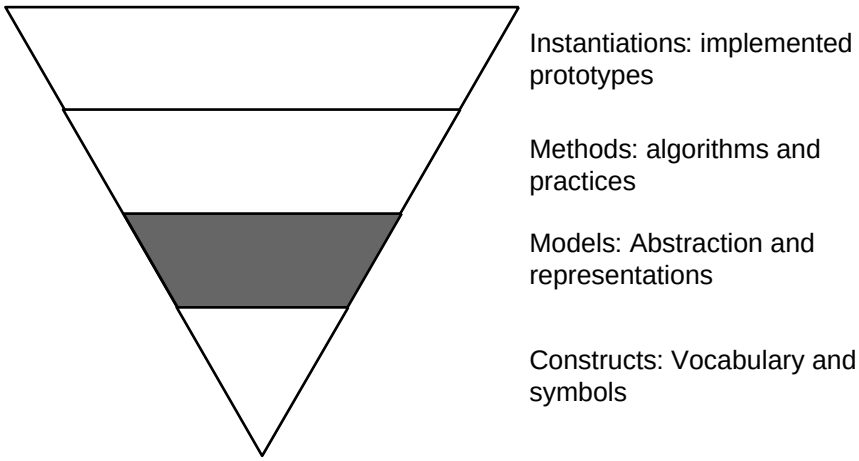

Figure 1.2: Type of IT artifact produced

research question by proposing an abstraction that can be utilized in the analysis of access control administration requirements in the context of provisioning systems.

The next section introduces the methodology applied in conducting this research project.

1.4 Methodology

The methodological design of the research project dealt with three phases: investigating the problem milieu, defining the abstraction and evaluating the abstraction.

The problem investigation involved an in-depth literature study into access control and its administration, as well as provisioning technologies. This theoretical understanding of access control administration was supplemented by studying the access control mechanisms of various systems. Some lessons learned were confirmed informally through discussions with an access control and provisioning specialist.

The development of the abstraction involved a search process and culminated from studying related material in contract law and the financial dis-

ciplines. Conceptually, the abstraction therefore re-uses existing constructs, but does so in a novel manner. The abstraction is formalized by expressing its essence as mathematical expressions.

Taking into consideration the classification of IT artifacts in Figure 1.2 it is argued that an artifact can only be considered useful if it contributes something to the "higher" level, in this case if methods utilizing the model can be devised. The abstraction proposed here is thus evaluated by showing how it may be used in specific case studies.

The results of this research is reported in the dissertation according to the layout described below.

1.5 Layout of Dissertation

The layout of the dissertation is depicted in Figure 1.3. The layout roughly mimics the methodology followed.

The current chapter delineated the research problem. It is followed by Chapter 2 which explores the essential underlying assumption that the business world has changed significantly. Hence the environment in which IT systems operate, and where access control must be administered, has changed. Thereafter Chapter 3 discusses the access control paradigms that have been developed over the years, as well as the problems associated with the administration of access controls. Chapter 4 completes the discussion regarding the problem milieu by explaining the concept of a provisioning system and how it may assist with existing problems.

The agreement abstraction is formally defined in Chapter 5, after which Chapter 6 shows how the agreement abstraction can be put to good use by discussing a case study.

Finally Chapter 7 concludes this dissertation. In addition to the main text of the dissertation, Appendix A provides some evidence regarding practical investigations done.

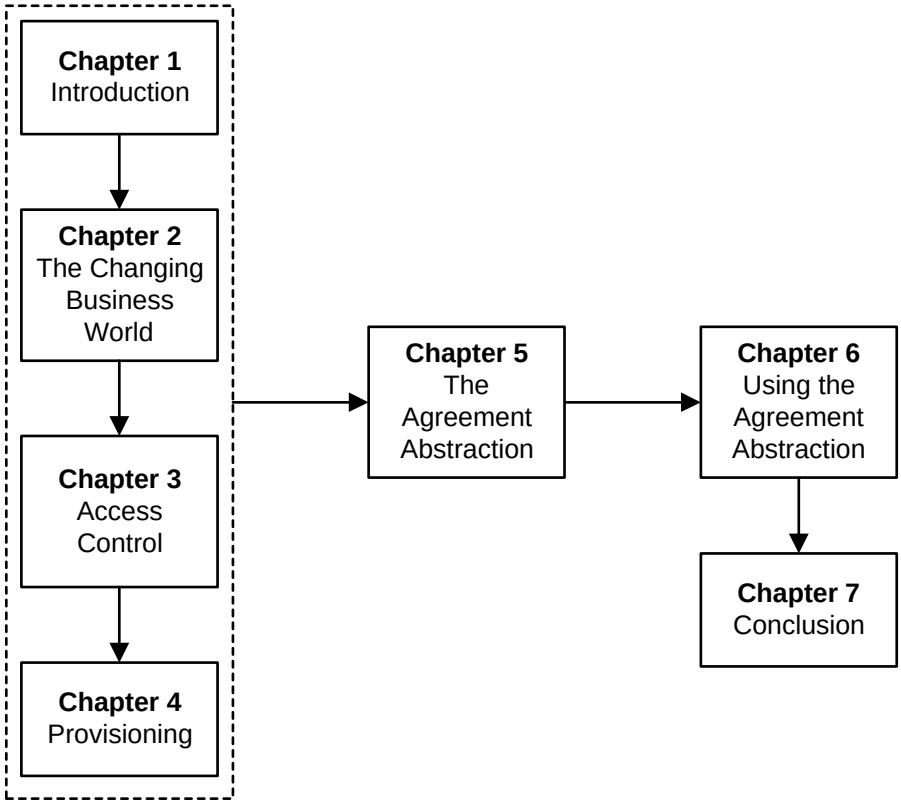

Figure 1.3: Layout of the dissertation

Chapter 2

The Changing Business World

The abstraction presented in this dissertation is based on the premise that current access control administration models are not efficient. This inefficiency, the author upholds, is a result (at least partly) of changes in the business world. Activities are no longer performed in the same way they were 20 or more years ago.

Eliasson (2005) notes that business management has changed considerably over the past few years. The activities of business management, however, are still those identified by Massie (1979): making decisions, coordinating business activities, overseeing people and evaluating business performance. The difference lies in the way these activities are performed.

Thus, this chapter sets out to describe the changes in the business world in a bid to motivate for the foundational concepts used in the proposed abstraction.

The changes are discussed from three perspectives. Firstly, section 2.1 describes the move towards a knowledge-based economy. In a knowledge-based economy, workers are principally empowered through access to information. The changes in technology, described in section 2.2 facilitate information sharing. If business is conducted differently, it follows logically that the way in which business' success is measured has to change as well. This is discussed in section 2.3.

2.1 Towards knowledge workers

In the late 18th and early 19th century, an economy based on manual labour was replaced by one dominated by industry and the manufacturing of machinery. This is commonly referred to as the Industrial Revolution (Massie, 1979). During the industrial age, the focus was on output. This view resulted in businesses up to the early 1970s perceiving the world to be predictable and local (Eliasson, 2005). Scientific Management (Sadler, 1997; Massie, 1979) saw jobs as narrow and rigidly defined, and specific work methods were enforced. Even office work was reduced to a "manufacturing line" approach, with careful division of labour and high specialization.

However, in the 1980s, this started to change. The growth in telecommunications and computing lead to information becoming increasingly valuable. This importance of information, and its link to information technology, was predicted two decades before by Leavitt and Whisler (1958).

Hammer and Champy (2003, p. 87) highlight that during the re-engineering process, information technology plays a crucial part. It is said that information technology is the "essential enabler". However, the dawn of the information age combined with other forces changed the focus of businesses. Hammer and Champy (2003) identify three forces: customers, competition and change, called the three Cs, which induced a change in focus. Firstly, the business market is no longer manipulated by the seller: the customer is now able to manipulate the seller. Secondly, the competition has intensified. In the past, if your business had the best product or service at the best price, the business was yours. In today's day and age customers value relationship and service levels. Finally, the physical nature of the business world is constantly changing, which means that businesses must adjust to changes in the market quickly and without effort, to have a competitive advantage.

Nowadays businesses realize that the world is not predictable and local; instead, it turned out to be highly complex. It is believed that the complexity of business management can be overcome through the use of technology, and that these technologies would assist managers in making decisions.

By 1995, the business world already had become much more heterogeneous, complex, as well as very unpredictable, due to new computing and communications' technologies. These technologies also provided managers

Table 2.1: Salient differences between the Industrial Economy and the Knowledge Economy

Industrial Economy	**Knowledge Economy**
Output focused	Customer focused
Narrow Focus (Taylorism)	Wide focus
Specialization	Generalization

with the means to make better decisions, based on the fact that information could now be gathered and processed, as well as analyzed. In order for top and middle management to make good decisions, they need to have the best quality information at hand (Higson, Zimmermann, & Itter, 1997). Decisions based on incorrect information could result in a business losing a great deal of money. However making decisions based on up-to-date and accurate information, at any given time, can be greatly beneficial to any business.

In the 1970s, the business world was seen as predictable and local, now it is seen as heterogeneous, complex and unpredictable. It can, therefore, be stated that the world of business has changed considerably.

Table 2.1 lists the salient differences between the Industrial Economy and the new Knowledge Economy.

According to Sherif and Xing (2006), the only sustained sources which are proven to give a business a competitive advantage are knowledge and the management thereof (Nomura, 2001). Thus, managers are currently relying quite heavily on information to make important decisions, and therefore, it is no surprise that the amount of knowledge workers who exist in a business, has steadily grown from the 1950s to 2000 (Wolff, 2005). Knowledge workers are employees within businesses who extensively use information as well as create new information from existing information.

Drucker (1973) stated the following: "The manual worker is yesterday...The basic capital resource, the fundamental investment, but also the cost centre for a developed economy is the knowledge worker who puts to work what he has learned in systematic education, that is, concepts, ideas and theories, rather than the man who puts to work manual skill or muscle."

However, in order to open the full potential of knowledge workers, such a culture needs to be established within businesses.

There are many definitions of the term culture. One such definition by Kroeber and Parsons (1958, p. 582) stated that culture is: "... transmitted and created content and patterns of values, ideas and other symbolic meaningful systems as factors in the shaping of human behavior and the artifacts produced through the behavior" (Kroeber & Parsons, 1958). Thus, culture is a group of ideas or beliefs and values that a specific group of entities share (Ramachandran & Rao, 2006). An organizational culture can be seen as the way things are done within an organization.

An information culture is established within a business when everybody in the business sees that information is important, and recognizes that information can be used to the benefit of that business. In such a business, information would be gathered, analyzed, and stored in an archive in an accessible way.

A change towards an information culture can only be initiated by top management. Such change is non-trivial. Culture changes can only be induced through leading by example and extensive communication and incentive programs. When such an information culture is established, all the business employees will realize the effect that information can have on their job, as well as on the business itself. This raised awareness will result in more information being created within that business.

When knowledge workers are constantly creating more knowledge in a business, new problems are created. Nomura (2001) raised the question of how do a business evaluate and decide what knowledge is important? A second problem is how businesses identify and understand exactly how each employee creates knowledge. These problems, however, will not be addressed further in this study. Despite these problems, the fact remains that employees of a business will constantly create new knowledge, and access to this information needs to be controlled.

It can be questioned whether this move to knowledge workers is really happening. While evidence to this regard is anecdotal at best, good arguments can be made that this is indeed the case. According to Cappelli and Hamori (2005) the hierarchy in businesses is flattening. (Nohria, 1991) also provides evidence in reporting that executives are reaching top management positions more rapidly. It can be argued that fewer managerial positions result in employees needing to take on more responsibility (while not neces-

sarily moving up in the organizational hierarchy), thus becoming knowledge workers.

Of course, giving people more responsibility is not the silver bullet in knowledge management. Davenport and Beck (2000) note that each employee has only a set amount of attention. This implies that each employee should clearly focus on the activities that are most important. It also follows that giving employees access to everything is not a panacea, instead it may deplete their available attention and lead to an information overload. Limited access is therefore not only a security and privacy concern, but a practicality.

Furthermore, activities which could possibly be automated should thus be further investigated and automated, if possible. New and existing technologies can provide the business with the means for this automation process.

The next section investigates the changes in technology and how these have impacted on business strategies.

2.2 Technological changes

Computers have had a significant impact on our everyday life, since their introduction. However, much has changed in the technology from when it was first developed.

The following section briefly discusses the various technologies as well as the impact they have had.

2.2.1 Power of technology

The development of computers has come a long way in the last decade. This can be seen in the way computers are used for research, in successful businesses and within many households. However, this change did not come easily.

The development of computers has been categorized into four generations based on the technologies used. The first generation can be seen as having existed up to 1955 (Palmer & Morris, 1980, p. 7). Vacuum tubes were used, as the hardware and operation were only done sequentially. The second generation, from 1955 to 1960, used transistors, and magnetic core memories. The user was able to execute both input and output operations simultaneously. From 1960 to 1970 the third generation existed. The computer

hardware was changed to integrated circuits. This meant that higher speeds, as well as greater capacities, could be found. Many different programs could also now be run at the same time. The fourth generation was from 1970s onwards, and showed the birth of real-time processing. It was around this time that computers were starting to be sold. IBM, still known today, was one of the businesses selling computers.

From the 1970s, much progress was made in the computing arena. In 1971, Ted Hoff, A Mazor and F. Fagin developed the first single chip, called the Intel 4004 (Computer Staff, 1996). In 1972, the Intel 8008, the first 8 bit microprocessor, was seen, and soon replaced by the 8080. Hand held calculators also became popular in that year. In 1975, the first personal computer (PC), the Altair 8800, was made available in kit form. The progress made by engineers on computers was predicted by Gordon E. Moore's paper, published in the 35th anniversary edition of the *Electronics* magazine (Mollick, 2006). In this paper, he stated that the number of components that could be placed on a single chip would double every year. This prediction was later changed by him to doubling every two years.

Even today, computer technology changes on a daily basis. It is not only the processor technology which has grown, storage technology has too.

The size of storage is also becoming smaller (Paulson, 2005). The normal hard drive found in desktop computers is 5.25 inches in diameter. Capacities range from 80 GB to more than 300 GB currently. The capacity of hard drives has not only grown, but their price has decreased as well. Notebook hard drives are fitted with 2.5 inch hard drives. Toshiba have announced that they now have a hard drive of 0.85 diameters.

Technology of storage has excelled in such a way that businesses are now able to store the vast amount of information that knowledge workers generate. Because of the storage capacity and price, businesses are given more choices as to what they can store, the speed at which it can be extracted, as well as the processing done on the stored data. This was previously not an option because of the cost and availability of storage capacity.

Although it is possible for businesses to store vast amount of information, this information needs to be freely available to be used by its employees. The business internal network and connectivity to the Internet are common technologies; however they have also come a long way.

2.2.2 Connectivity Changed

In 1966, Lawrence G. Roberts set up the first computer network (Roberts, 1986). This was done by means of a 1200bps dial channel between a Lincoln Laboratory TX-2 and System Development Corporation's Q-32. A few months later, in October 1967, the initial plan for ARPANET, the first computer network, was published, and December 1969 showed the first implementation. In October 1972, at the first International Conference on Computer Communication (ICCC), ARPANET was first showed in public. ARPANET later became what we now know as the Internet.

At the same time, a system called Alohanet was being developed at the University of Hawaii (Cisco Systems, 2003a; Abramson, 1985, p. 252). The system enabled various stations on the island to be connected via radio waves to each other.

The reason for these projects being executed was because various needs were raised. Computers were purchased and used as individual stand-alone computers by businesses (Cisco Systems, 2003a, p. 44). Some of these computers had connected printers, which enabled employees to print documents. However if employees wanted to print, they had to put the file to be printed on a floppy drive, walk over to the computer which had the printer connected, and print it there. This was later termed sneaker net (Thomas, 1997). It was soon realized that this was not very efficient nor cost effective. Various other problems also existed. Consider the following example. Employee A and B have access to a specific file. Employees A and B make changes to this file. This means that there are now 3 versions of the same file.

Local Area Networks (LANs) were developed to overcome these problems. A LAN enables a business to share various files and printers by means of the installed computer systems (Cisco Systems, 2003a; Lacoble, 1987, p. 49). A LAN only cover a small geographical area, like, for instance, a building or a campus.

Wide Area Networks (WANs) are networks which span a large geographical area (Cisco Systems, 2003a; Fraser, 1996, p. 50). It enables a business to communicate with other computers over a large distance. With WANs it is possible to sit in your office in South Africa and be able to talk to a business partner in Germany.

As previously mentioned, ARPANET later on became the Internet. The

Internet is a large network consisting of thousands of servers, all connected to each other. According to the web site (Internet Statistics, 2006) the Internet has grown 200.9% in the last 6 years. This means, from the year 2000 to 2006, 1, 086,250,903 users, 16.7% of the world population, have been caught using the Internet. By looking at this statistics it is evident that the Internet has grown tremendously. The reason why the Internet has grown so much is because businesses as well as users have realized the value of the Internet.

However, the types of technologies which could be implemented on the Internet were dependent on the speed at which users could access the Internet. Many users previously, and still today, can only access it via an analog dialup modem. The maximum speed at which an analog dialup modem can access the Internet is relatively slow. The reason for this is that the equipment used by the telecommunication network providers could only operate at 4 kHz (Peden & Young, 2001). This meant that all the communication had to work over one copper cable. With the introduction of newer digital transmission other signals than the original 4 kHz signals could be used, meaning that bandwidth could increase.

The latest technology to be commercially deployed is ADSL. Although ADSL has enabled many users to have high speed access to the Internet (Knight, 1999), 80 percent of large businesses in the United States are still using T1 leased lines to access the internet as well as connect to other offices and business parters (Gerwig, 2001).

T1 leased lines, referred to as E1 in Europe and the rest of the world, have been used by businesses all over the world from the 1950's (Sherburne & Fitzgerald, 2004). However, the copper leased lines are being replaced by faster fiber optic lines.

Because of the increase in bandwidth, the technologies which can be implemented by business have changed. These technologies have also changed the way businesses conduct their operations. Businesses should thus focus more on Internet commerce activities. Businesses should realize that because more customers have access to broadband, they are going to receive more business from these users.

Because bandwidth has increased and many more users have access to it, applications have been developed. Many of these technologies have had a major impact on both businesses and individual users.

2.2.3 Application Technology

There are various application technologies which have been developed. These enable businesses to be more effective and efficient. One such technology which is used by businesses is called email.

Email is a technology whereby an electronic mail message, possibly with an attachment, is sent to a recipient. The only delay between the sender of a message and the recipient is the time the email servers take to forward the message. In many cases, the message will be delivered within seconds or minutes. Although the time taken for an email to be delivered to a recipient is little, the need was seen for users to communicate in real time irrespective of where they are in the world.

A collaborative system enables a team of employees to work together, irrespective of where each employee is situated (Takahashi & Yana, 2000). While geographical distances do play a role, collaboration software enables employees to work in teams that are both cross-functional and cross-cultural, as well as separated by different time-zones. It is due to the Internet that global collaboration has been made possible.

Another technology which has changed the way businesses function is called workflow management. The Workflow Management Coalition defines "workflow" in the Workflow Reference Model as: "the computerized facilitation or automation of a business process, in a whole or part."(Hollingsworth, 1995). Workflow helps businesses facilitate the automation of procedures. These procedures are set up based on the defined set of rules to achieve, or on a specific business goal. These procedures could include the passing of documents, information or even tasks. The passing of the second, information, has become imperative in today's organizations.

Organizations have realized that they are warehouses of valuable information. This information has to be managed and secured properly. If managed and secured properly, an organization can effectively use this information advantageously against its competitors.

In order for an organization to manage its information, it must make use of content management. Content management is the process of realizing what information an organization has and identifying the information an organization needs (Boiko, 2002). Content management can also be seen as the overall process of collecting, managing, and publishing information.

Because it has been realized that information is of such importance, and the Internet contains much more information since the 1990s, the number Internet users has grow substantially (Townsend, 2002).

However, mobile, and specifically, wireless, communication devices have grown more than the Internet. This shows that mobile technologies are changing the lives of people (Chen & Adams, 2004). A mobile communication solution gives users an alternative to a wired solution. Employees can now go to a client and still have access via their mobile devices to the businesses' resources.

For example, imagine trying to urgently get hold of a business partner away on a trip, when on holiday, 15 years ago. It could be very time consuming, or even impossible. Today, you can pick up your mobile phone and dial the other business partner's mobile number. If he wants to speak to you, he will answer. That business partner might be thousands of kilometers away, still, you will most probably be able to contact him/her.

Based on the discussion, there are various technologies, introduced in the last few years, which businesses can use. These technologies have changed the way businesses operate. Not only did the technology available for businesses change, as discussed in this section the world in which organizations conduct business has also changed considerably. If the world of business has changed so dramatically, how does a business measure its performance? This will be discussed next.

2.3 Business performance measures

In section one of this chapter, the main focus was that the business world has changed. The previous section discussed various technologies which are at an organization's disposal to use, in order to have a competitive advantage. These technologies have changed the way that businesses conduct their operations, and in so doing function in a more efficient manner. It can thus be said that if the business world has changed, as well as the technologies businesses use, the way business measures performance should also change to better suite the needs of businesses.

According to Nadler and Tushman (1997) there are three main criteria for evaluating performance of a business (Nadler & Tushman, 1997). The first

criterion is goal attainment. In each business a strategy exists which contains the objectives of that business. Goal attainment is about how successfully these objectives are met.

The second criterion is the amount of resources that are utilized. A business should utilize as much of its available resources within the business in order to meet its objectives. Finally, there is the adaptability of a business. It is very important for a business to make the most of the opportunities which are identified, as well as address the threats which arise in the ever-changing world of business. One method that is used to evaluate business performance is by means of a balanced scorecard.

2.3.1 Balanced Scorecard

In the early 1990s Robert Kaplan and David Norton identified the need for a mechanism to measure the performance of a business, based on both the financial, as well as non-financial measures. Therefore the balanced scorecard was developed (Hwang & Leitch, 2005). The reason for this was based on the premise that although a business could be showing a good profit, it could be possible that other areas of the business were not performing that well, and it may be possible that problems were on the horizon, which could not be identified. Financial indicators are very important, but other indicators should support the financial ones to ensure that the future success of the business could be predicted (Niven, 2005, p.13). That brings us back to the question: How does a business measure its performance?

The balanced scorecard measures the performance based on the vision and the strategy of the business. Based on the balanced scorecard, a business can only measure its performance if the business has set clear objectives. (Niven, 2005, p.13). If the balanced scorecard is used correctly, it should provide the business with the means to easily, as well as successfully, measure performance of employees working on the floor, right up to top management.

Kaplan and Norton (1996) identified four key focus areas each business should consider in order to measure performance: financial, customer, internal business process, as well as learning and growth measurements. Each of these areas is depicted in figure 2.1.

The first one, financial measurement, is the more traditional measure of a business. This measurement looks at the profit a business has made. The

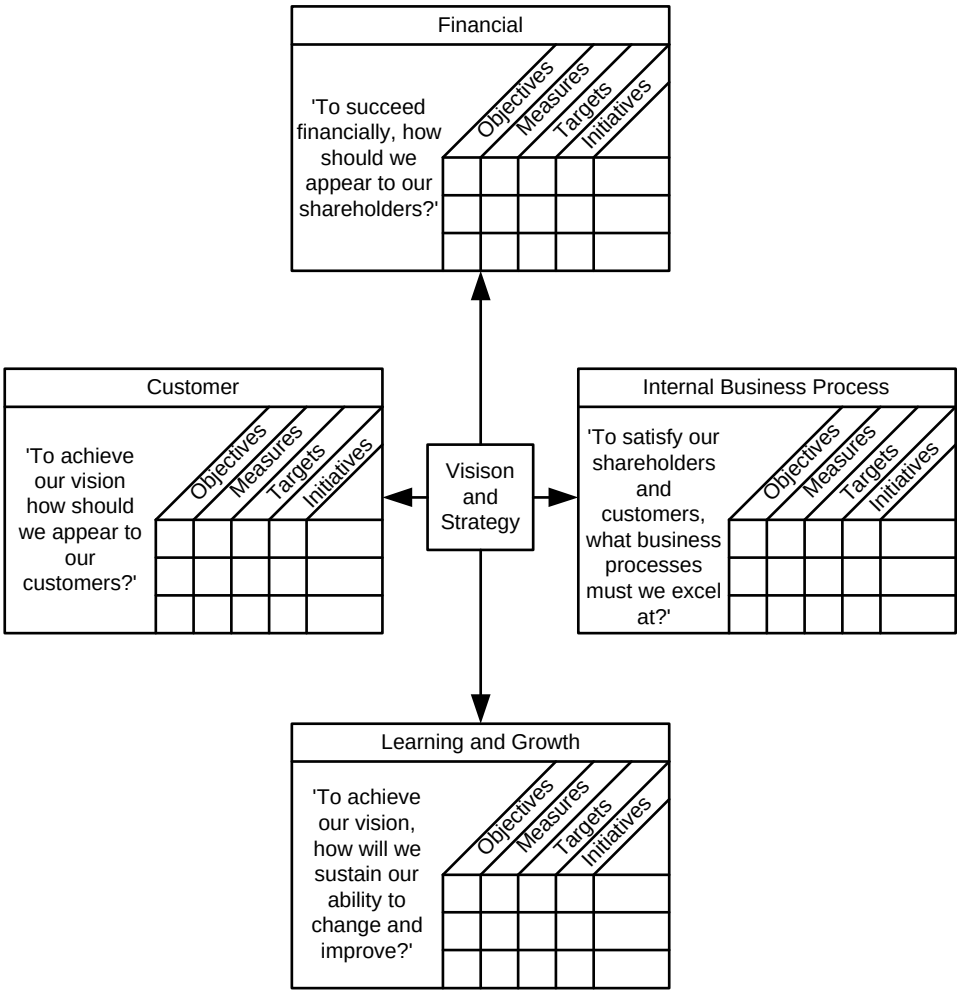

Figure 2.1: The Balanced Scorecard

(Kaplan & Norton, 1996)

customer measurement takes into consideration whether the customers are happy with the service they are getting. The internal process measurement identifies key processes within the business, which could be improved in order to provide a better product or service to a customer (Niven, 2005, p. 13). Finally a learning and growing measurement is the process whereby the business needs to identify where the problems lie. This will enable the business to know where to give more attention.

The balance, in which these four main focus areas work, is one of the most important aspects of the balanced scorecard (Meyers, 2002, p. 87). Any business should have a clear objective. A clear list of specific actions or projected results for individual employees, as well as units within the business, should then be constructed based on the four focus areas. Scores are then given to each of the listed items. Based on the scores, the business can then evaluate whether it is performing according to its business goals.

Measuring business performance is very important. Furthermore it could be beneficial to a business to predict what the performance of a business will be.

2.3.2 Information Orientation

According to Marchand, Kettinger, and Rollins (2000), business performance can be predicted by Information Orientation. Information Orientation is a metric which indicates a company's capability to effectively manage and use information.

Marchand et al. (2000) define several competencies that measure the information orientation of a business. These are divided according to three information capabilities. The first category, Information Technology Practices, highlights information technology applications as well as infrastructure, and the capability of the business to effectively manage and use them. The second category, Information Management Practices deals with the capability of a business to effectively manage information over the life cycle of information. The third category, Information Behaviour and Values, measures the business' ability to instill and promote a culture of effective information use in its employees.

The effective use of information by top and middle management, as well as the role Information Technology would play by presenting this informa-

tion, was predicted by Leavitt and Whisler (1958) decades ago. The work by Marchand et al. (2000) on relating information orientation and business success confirms the prediction made by Leavitt and Whisler (1958). In order for top and middle management to make good decisions, they need to have the best quality information at hand.

IT can indeed provide management with the means to have precise and up-to-date information at hand. However, Marchand et al. (2000) emphasize the importance of managing information through the information life cycle. They emphasize several competencies in the category Information Management Practices that are critical for good information orientation. Employees are using information on a regular basis to make decisions, and therefore the Information Management Practices within the organization should facilitate the process of sensing when information could possibly have a benefit for the organization. Employees should, furthermore, be able to understand that this piece of information is relevant to that specific business.

Once it has been noticed that information could possibly be important, a decision should be taken to either include it in the appropriate systems within the business or to discard it.

If information management becomes part of the life cycle, the amount of information contained within that business will grow drastically. Thus, the main focus of information management should be the context in which decisions are made by managers as well as the employees. Each and every employee should realize that a huge responsibility exists for information management.

When information contained within the organization is openly shared, it is possible for employees, as well as managers, to use this information to create new information or to use the information in new ways.

While the prediction by Leavitt and Whisler (1958) regarding management and Information Technology was spot on, they clearly did not anticipate the empowering role that information technology would have on normal employees. Today managers and employees can leverage the available information together. They can either use existing information in new ways or create new information.

2.3.3 Regularity compliance and Corporate Governance

Corporate governance has been receiving much attention over the last few years. The focus became even more intense after the collapse of various large businesses like Enron, WorldCom and Parmalat (Marnet, 2007). Because of these collapses, government was forced to implement drastic measures as well as change legislations to restore faith from investors. One such measurement introduced in America, with which businesses need to comply, is the Sarbanes-Oxley Act of 2002, furtheron called SOX.

SOX and other compliance acts as such, stipulate exactly what businesses should focus on. The act also specifies the annual reports of financial results which must be produced by all major businesses (Watts, 2006). Apart from the financial reports which need to be reported upon as specified by SOX, other data security and privacy regulations have been introduced which businesses need to comply with (Robinson, 2005). If businesses fail to comply with these standards, or if irregularities are found in the business, various parties will be held responsible. Therefore, top management have realized that they need to ensure that their data is kept safe (Robinson, 2005).

Businesses thus have a privacy and security responsibility towards the government as well as to their shareholders. A much closer look at what the business is doing with its information is necessary in order to maintain a high level of security.

However, in Chapter 1 it was argued that employees should be given access to relevant information. Yet, since businesses are forced to protect their information, access to information cannot be given freely. It is therefore imperative that businesses must be able to defend their actions. To do this, it is perhaps maybe not as important to prevent access as it is important to audit the fact that persons were granted the specific access.

Within the context of the Direct-Control Cycle proposed by Von Solms and Von Solms (2006) this would imply that some measures to identify suspicious access control administration activities to the relevant management layers must exist. Business should be able to produce reports that interpret underlying system activities in terms of the bigger business risk. Ultimately The Board of a business should be able to report thereon and prove that the business was not exposed to undue risk. This illuminates an interesting perspective on the possible importance of access control log files.

Therefore, it should be possible to audit the access which is given to employees. When working in the IT environment it is possible to store such an audit trail on the information accessed. It has thus been argued that information security governance forms part of good IT and corporate governance (Von Solms, 2005).

2.4 Conclusion

This chapter explained that the world of business has changed over the last few years. Reasons for this change included that because business was at first seen as predictable and stable, and later seen as unpredictable and ever changing. Businesses realized that utilizing information can give them an advantage over their business competitors.

Because information is playing a growing role in business, more and more employees became knowledge workers, who collected information and created new information from that.

The point was raised that the world of business, as well as technologies used by business, change, and that the ways businesses measure their performance also have to change. There are various aspects of a business that should be measured. One such aspect is the performance of a business' access control administration. This aspect is discussed in the following chapter.

Chapter 3

Access Control

Chapter 2 showed that businesses should empower their knowledge workers by ensuring that they have access to the appropriate information. However, as more and more employees become knowledge workers, the amount of information within a business grows exponentially.

This leads to a conundrum: While we want employees to have access to information, it would also be irresponsible to not exercise appropriate control over the data. The problem at hand is how to manage access to this information in a way appropriate to an emerging knowledge economy.

This chapter explores existing access control paradigms. In particular the investigation pays attention to how access control administration is done. It highlights that employees, who might not have a complete knowledge of the business, might be implementing security. In conclusion, this chapter argues that due to the wide-spread implementation of current access control paradigms, new paradigms may not be accepted easily. The chapter, therefore, does not propose new access control paradigms, but asks for increased attention to the administration of these access controls.

The chapter commences with positioning access control within the general ambit of information security.

3.1 Information Security

Chapter 2 has shown that information and the management of that information have become invaluable to organizations. In fact, we reflected on research that showed that businesses, which effectively use and manage their

information, perform better. It is, therefore, increasingly important for businesses to secure their information.

In order to ensure *secure* information, a number of security objectives must be met. These include that information must have an appropriate level of *confidentiality*, maintain a state of *integrity* and have *availability*.

Typically, these objectives are implemented by means of various security services. Five main categories of security services are defined by the ISO 7498-2 standard (ISO, 1989): Authentication, Access Control, Data Confidentiality, Data Integrity and Non-Repudiation. The main focus of this dissertation is on the Access Control security service.

Access controls prevent unauthorized access to the resources of a system (Oppliger, 2001), as well as prevent systems to be used in an unauthorized manner. For example, if an employee leaves his/her office, he/she must first log out of his/her computer. Logging out from the computer is a form of access control to ensure that other employees cannot have access to his/her computer. At the same time when the employee is logged in, his/her access to resources in the system is appropriately constrained.

The way these access controls are implemented into the various systems is based on the type of access-control paradigm used. The next section elaborates on various access control paradigms.

3.2 Access Control Paradigms

Merriam-Webster Online Dictionary (2004) defines a paradigm as: *"A philosophical and theoretical framework of a scientific school or discipline within which theories, laws, and generalizations and the experiments performed in support of them are formulated"*.

Within access control, the paradigm dictates the way security administrators think about the question, *"Who has which kind of access to what resources?"*. The assumptions made, and the general school of thought used when answering this question provide the conceptual foundation on which access-control mechanisms are built.

This section reviews several access control paradigms. Based on these paradigms, models can be developed to facilitate the implementation of access control into systems. This research does not explore the physical imple-

mentation of access control as the proposed abstraction operates at a higher level of abstraction. However, several existing access control mechanisms have been investigated to gain a better understanding of the access control philosophies prevalent in current environments. These are briefly reported on in Appendix A.

Consider the first paradigm: access control based on security labels.

3.2.1 Access control based on labels

In an access control paradigm based on labels, employees and resources within an organization are assigned security labels (Sandhu & Samarati, 1997; Sandhu, 1993). When applied to resources, these security labels define the sensitivity of the resource. An employee is labelled according to his/her clearance. Labels such as "Top Secret", "Secret", "Confidential", "Restricted" or "Open" are related to one another according to a partial ordering.

Employees may work with documents of a similar sensitivity level, or lower. It is, however, counter-intuitive that, for example a major in the army whose clearance level is "Top Secret", will have access to all the other resources within the army.

In practice, this problem was addressed by grouping different categories together to form a compartment. Thus, an employee can only get access to a compartment, if his/her security label matches, or is higher than, the highest security label found in that particular compartment.

The access-control paradigm, based on labels, is mostly used by the military, because that institution has a strong hierarchy. But such strong hierarchical structures do not exist within businesses. For this reason, this paradigm has not been well implemented in the business world. Access control based on ownership was therefore developed for use in business.

3.2.2 Access control based on ownership

Access control based on ownership allows the resource owner to decide who receives access to what resource. This is done by using Access Control Lists (ACL) (Sandhu & Samarati, 1997).

There are, however, a few problems with this paradigm. Once an employee has the authorization to access a resource, that employee is able to share the resource with whoever they like (Bertino, de Capitani di Vimercati, Ferrari, & Samarati, 1998). Access control paradigms based on ownership are also susceptible to Trojan-horse attacks (Bertino, Samarati, & Jajodia, 1993).

The question needs to be raised as to who the owner of information is within a business? If a person works for the business and creates information, it does not necessarily mean that employee is the owner of the created information. The owner of the information is the business. The role-based access-control paradigm focuses more on the organizational aspects of implementing access control, rather than on the ownership of the information. This brings us to access control based on roles.

3.2.3 Access control based on roles

Access-control paradigms based on roles are of the most widely used. It simplifies the implementation of access control by associating various permissions with a specific role (Sandhu, Coyne, Feinstein, & Youman, 1996). Each employee is assigned a specific role, which gives that employee the necessary permissions to effectively fulfill his/her job.

Role-based access control, however, is not without its shortcomings. In order to enable, or restrain, access to a resource within a business, an administrator must define and set up each and every user of the system, the group the user belongs to, all the resources, as well as the permissions. In a large organization especially, this could prove to be a daunting task (Shin, Ahn, Cho, & Jin, 2003). Even if all of the users, groups, resources and permissions are defined and set up, a user may, nevertheless, be able to misuse this permission. It is due to these problems that other access-control paradigms have been developed (Thomas & Sandhu, 1993, 1994, 1997; Atluri & Huang, 1996; Botha, 2001; Samarati, 2002). One such enhancement is that employees are only able to use the access-control permissions assigned to them when specific tasks need to be executed, depending also on the context in which those tasks occur.

3.2.4 Access control based on tasks and context

As previously mentioned, access control based on roles assists the assigning of job categories and permissions to employees. Employees within an organization are assigned these permissions to enable them to view information that could be sensitive in nature. The problem thus is that employees can misuse their permissions to act in fraudulent ways.

As far back as 1993, Thomas and Sandhu (1993, 1994, 1997) highlighted that a problem existed with the access-control paradigm, based on the above. Thomas and Sandhu (1993, 1994, 1997) proposed an active approach to authorization management, which was called the Task-Based Authorization Control Model. This model enabled roles to be automatically activated and/or deactivated. This message was echoed by Atluri and Huang (1996), who developed the Workflow Authorization Model (WAM). Atluri and Huang (1996) argued that employees should only be able to obtain access to a resource when a specific task requires it.

Bertino, Ferrari, and Atluri (1999) and Botha (2001) reinforced the view that an employee should not be able to access resources within workflow systems at all times. In their cases, the workflow management systems provided the context for access-control decisions.

Core to the abovementioned paradigm is the principle that an employee should not be able to access a resource, based only on his/her identity. An employee's access should also be based on the context of each task, most often dictated by the business processes. This paradigm thus relies on determining at the design time which information a particular task requires. Furthermore, organizational policies, such as separation of duty policies, should influence the access-control decisions made.

All of these models restrain employees from gaining access to resources to which they are not permitted. However, all the models differ from the traditional role-based access control by including the means of deciding when these access controls should take effect. For these models to be successful, it is imperative that access-control decisions occur at the correct time. Under this paradigm these decisions depend on the premise that, in order for employees to complete their work, the minimum set of permissions required at that moment must be assigned to them.

Such a strict least-privilege approach prevents accidental (and inten-

tional) misuse of permissions. However, in a knowledge worker environment, it also severely restricts the user's ability to be creative and innovative, as essentially all tasks are moulded at design time.

Credential-based access control is an access-control paradigm which differs to others by using the credentials of an employee to decide whether to enable, or to restrain, access to a particular resource.

3.2.5 Access control based on credentials

The main idea of credential-based access control is that each employee has various properties or credentials (Samarati, 2002). In order for an employee to gain access to a resource, his/her credentials are matched to the minimum set needed to gain access to it. If the employee's credentials match these minimums, the employee receives access to that resource. Access control based on credentials as mentioned above, focuses on whom has access control permissions to what under what conditions.

Obligations is a new type of analysis which include the above mentioned criteria as well as specify the actions to be taken with the permissions.

3.3 From Access Control to Obligations

The main focus of access control abstractions is in order to restrain unauthorized access to resources. Therefore, in order to enable access for users the question, "who needs access to what?" was asked. As discussed later the question grew from "who need access to what...", to "who need access to what, when?". However, when any ad hoc constraints are necessary on these access controls, the constraints need to be implemented. Therefore, implementation time of such access control systems is increased (Gama & Ferreira, 2005). A new method, obligations, which identifies these constraints have been introduced and could possibly be the solution to the problem of implementing functional constraints.

According to Breaux, Vail, and Anton (2006) an obligation is: "a duty bound to an obligated party that must be complied with, often accompanied with a penalty for non-compliance". All the activities which must or must not be done to a set of target objects, are specified by means of an obligation

policy (Foster, Uchitel, Magee, & Kramer, 2006). Therefore all the actions which a specific subject is obliged to do, is maintained in the obligation policy (Irwin, Yu, & Winsborough, 2006).

However, the relationship between access control and obligations has not been well studied, according to Irwin et al. (2006).

It is also noted by Irwin et al. (2006) as well as Gama and Ferreira (2005), that with the introduction of obligations the administration of access control becomes even more of a problem. It is not just with the use of obligations that access control administration becomes a problem.

As gathered from all of the above mentioned access-control paradigms, access control and the administration thereof can be problematic. Many researchers have devoted their research to issues around access control. However, no complete enterprise solution exists.

If there is no complete access-control solution available, how do businesses administer access to their information?

3.4 Access Control Administration

This section sets out to explain how access control administration is done, and why it is such a difficult task. Firstly, the section mentions that growth within businesses causes complexity as well as heterogeneity. Thereafter, it is discussed that due to this adding of complexity to the systems environment, as well as the integration of these various sometimes complex systems, access control administration becomes even more of a problem. This is due to the various levels at which access control needs to be administered. An example of the amount of access control permissions is then discussed which shows that the administration of access control becomes a problem as the size of the business increases. Finally, the section discusses the impact a security administrator has on the efficiency of access control administration.

3.4.1 Evolutionary Nature of Systems

New businesses usually start as small entities. This could lead to a typical textbook network, such as the one depicted in figure 3.1. Usually not many computer systems will be in place. A textbook example of a small network is depicted in figure 3.1.

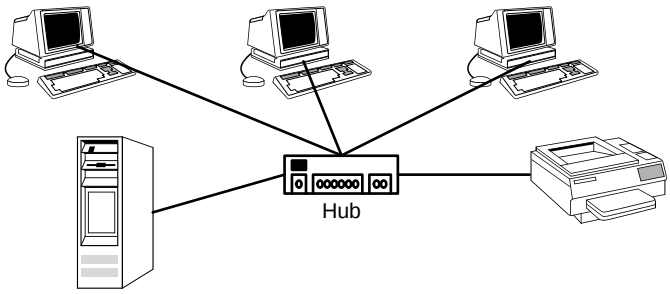

Figure 3.1: A Small Network

However, the way a business appears and operates changes over time. Businesses deploy new technologies that are not yet mature, upgrade, extend and re-structure their network. In particular, the re-structuring process of a business has a major impact on that business' systems environment. Such environments tend to get very complex, which means that it becomes difficult to keep them operating efficiently and effectively.

Apart from the fact that the structure of a business changes over time, businesses also acquire other businesses. Suddenly, the business' original systems and the systems of the newly acquired business must be integrated. The result is that what starts as a perfect textbook environment, quickly becomes a complex environment. Such a hypothetical complex environment is depicted in figure 3.2. To gain an understanding of how such environments develop, we dwell here on the evolution of this hypothetical network.

As seen in figure 3.2, there are various hardware devices as well as software to be found. This scenario, however, does by no means contain a comprehensive list of hardware devices and software.

Studying figure 3.2 reveals three parts. The first, Part A, is the original business which was started by the owner. When this business started, it was a great textbook example. The business started with only two computers and a printer. However, as the business progressed, more and more computer equipment were installed for various reasons. Part A has thus grown from a relatively small and easy system to a complex system. One of the main concerns in such a large system is the administration of access control permissions.

IBM Tivoli Access Manager is responsible for the administration of access control permissions in Part A. IBM Tivoli Access Manager is an access man-

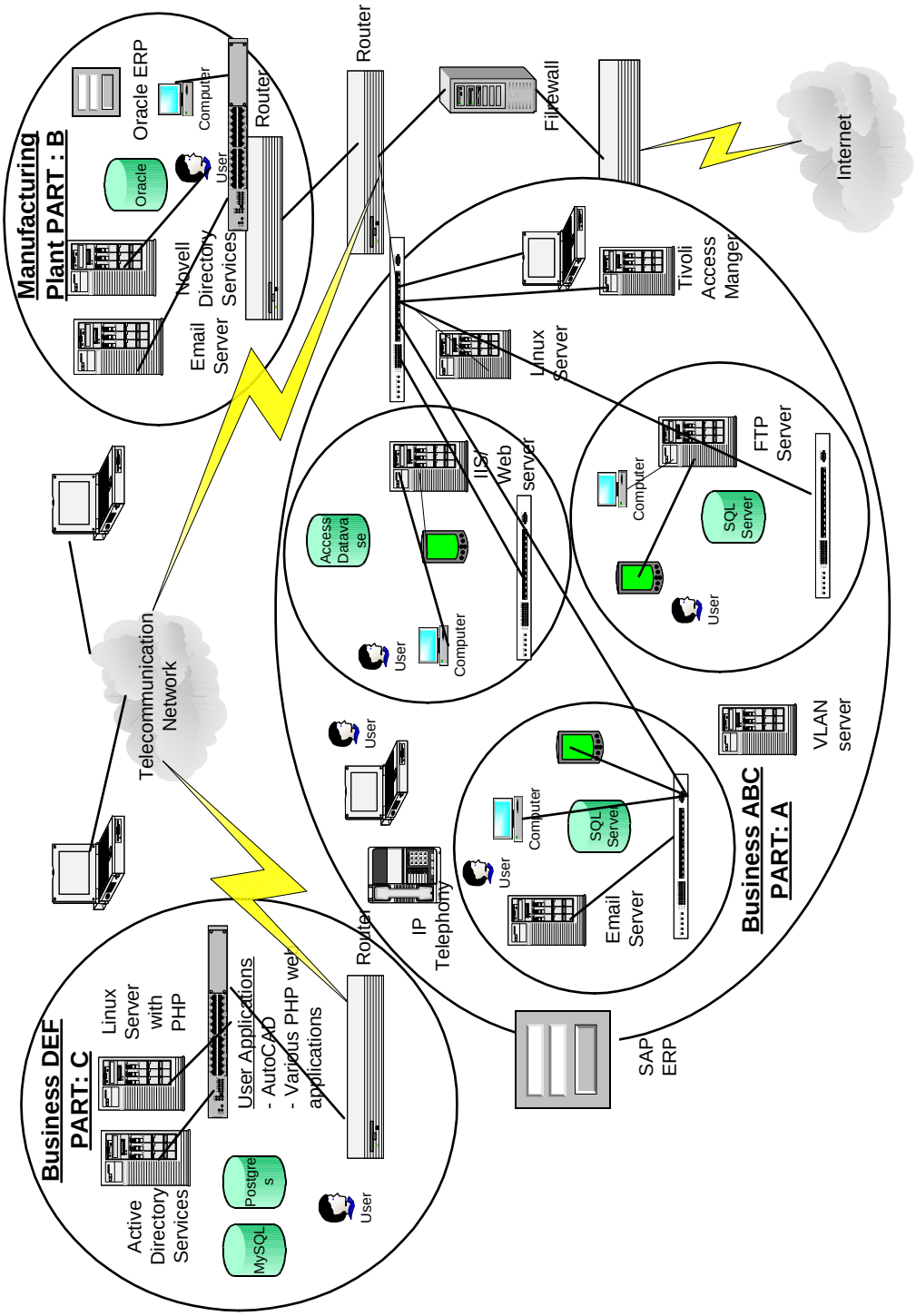

Figure 3.2: Complex Environment

ager which implements authentication as well as authorization on a business (IBM, 2003). IBM Tivoli Access Manager maintains the access control for various systems, including the IIS Web server.

The IIS server is mainly used to host the corporate web site as well as other smaller applications developed by the IT team. Microsoft Access and Microsoft SQL server are databases which are used to store information by these developed applications.

Another system which is commonly used is email. The email server enables employees of the business to send work-related email messages to each other. These email messages are also able to contain attachments, used if employees need to share files of a specific kind. However, if employees need to share files, for example, a plan for review by team members for a new mould being manufactured, employees are able to put the plan on the FTP server. The business can also put various files frequently used by employees onto the FTP server, which would enable employees' access to download these files as needed.

All of these services, including the FTP server, are accessible via the employee notebooks through wireless access points. However, not only wireless notebooks, but also wireless personal digital assistants (PDAs) access the server. Employees equipped with PDAs can also make use of the IP Telephony software which is installed. This enables employees to make long distance phone calls by using the Local Area Network infrastructure installed.

Without a network infrastructure, it is impossible for two systems' hardware or software to communicate with each other. In this scenario the network infrastructure consists, of routers, switches and cabling. The basic function of a router is to route information to the appropriate place in the network. A switch is a hardware device which connects various other hardware devices. Lastly, cabling is necessary to connect the personal computers, printers and servers to the appropriate networking devices. This environment is broken up into various Virtual Local Area Networks (VLANs). The (Cisco Systems, 2003b, p. 303) textbook describes a VLAN as switching technology, whereby switches are used to cluster personal computers and servers into logical groupings. This means that a business is able to logically group different sections within the business together. However, to fully take advantage of this technology, a VLAN server needs to be installed at the business.

Business ABC of our example, was doing very well and decided to start a manufacturing plant. This manufacturing plant is shown as Part B on figure 3.2. Because of the manufacturing equipment and the better support for Novell, it was decided to implement Novell Directory services at the manufacturing plant. Novell's email server, Groupwise, was also installed to enable employees at the manufacturing plant to send emails. Oracle ERP, as well as an Oracle database, were implemented to handle the manufacturing process. This plant enabled Business ABC to save on the manufacturing costs it had endured previously.

Business ABC then decided to acquire a second business, and bought Business DEF. Business DEF already had a set infrastructure, as well as existing systems. Business DEF is shown in Part C of figure 3.2.

Business DEF's user access is maintained by means of Microsoft Active directory. The Microsoft Active directory maintains access for users to AUTOCAD and to their personal computers. Furthermore various web-based applications have been developed, using PHP deployed on a Linux server. These applications mainly use MySQL databases as well as a Postgres database. Linux is responsible for access to the server itself, as well as the MySQL and Postgres databases. In order for Business DEF, in Part C, and Business ABC, in Part A, to communicate with each other, they need to be connected by means of a telecommunications network infrastructure. The provider of the telecommunication network will also be responsible for providing the infrastructure to connect this business to the Internet.

This Internet connection will provide business partners, employees connecting to the business through the Internet, as well as customers, the means to communicate with the business. However, giving such users access to the business intranet opens the door to possible threats from outside the business. Therefore, a firewall was installed to restrain unauthorized Internet traffic.

In a situation such as discussed and shown in figure 3.2, many technologies forced to work together in one environment is not uncommon. Many of these technologies were developed to function completely independently of other systems. Other systems could have been developed to be integrated. The problem, however, is that although these systems environment is complex, access to various systems, as well as to information should be authorized.

In figure 3.3 the black dots depict areas in this hypothetical environment, where access control information might be present. Consider some of the access control issues.

Various networking devices are deployed within such a complex system. These devices should all be configured in order to only provide access to authorized requests. It is possible that requests for access could come from a user, a personal computer, or even another networking device. Various VLANs will be configured, because to communicate with users on another VLAN, the VLAN server should provide the access permissions to do so. The VLAN server contains information regarding the access authority of one's personal computer.

This access authority could be based on the user login into the system or the address of the personal computer. However, the identity of a user on different applications is based on the login information supplied. This login information is based on the username and password supplied by the user, when logging into the particular system.

This identity is built based on the username, which was supplied when logging onto a PC. Based on this combination, either the Novell directory, the Microsoft Active Directory or Tivoli Access Manager would identify the user. For this reason these platforms need to be administered and given the correct usernames and password. However on all three platforms, different permissions must be associated with the username. Various systems use these directory services in order to enable users to make use of the services available.

One such service is email, which makes use of directory services. The directory service gives the user the specific permissions to access the email server. The server, on the other hand, will need to be set up for the user to have an email account. Once a user's PC has been set up with the correct settings, email access is acquired. As mentioned previously, if that user needs to share files with other users, a FTP is the appropriate technology to use.

Such a FTP server also needs to be managed to enable users to gain access. Various types of access can be set on such a FTP server. Access can be given to a user to be able to add files to the FTP server, to read or download files from the FTP server or both. In order to gain access to the FTP site, a web browser is used. These days the browsers have built-in support to be able to

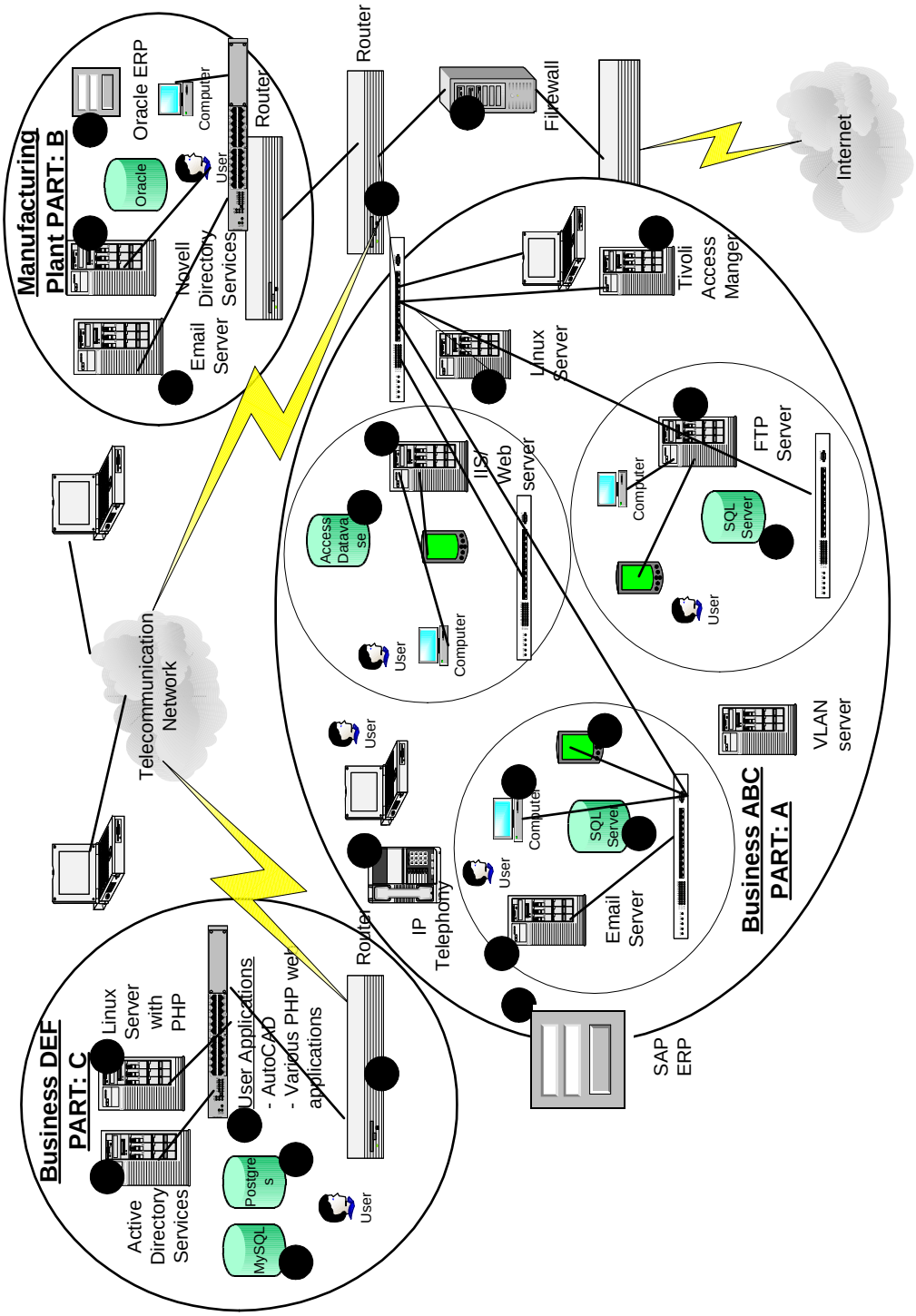

Figure 3.3: Access Control Information

connect to a FTP server. Browsers, because of their flexibility, are also used by many web-based applications to serve as the front-end.

However, to gain access to these web-based applications, the user needs to have the appropriate access control permissions. These permissions are set on the web server providing the web-application services.

One such system Business ABC implemented is an IIS web-based server. IIS also needs to be setup and administered to enable users to gain access to the services provided by the server. The same administration needs to be made on a Linux server, which also provides web-based services. If these web-based applications want to store information, they make use of databases.

Various databases exist, some of which have been used in Business ABC. In order for users or applications to gain access to these databases, access control permission needs to be defined. These can either be set on each database or they can be set on directory services. The directory service could contain access control permission to various other systems.

Many different systems, as well as services, are found within Business ABC. It is not only due to this fact that the system is of such a complex nature. The way that access controls need to work together in order to provide the best possible access control within the business is very important. The way that access control needs to be integrated is, however, very complex because access control is not only found at one level of abstraction.

3.4.2 Different levels of abstraction

Various resources within the environment implement access control in different ways. However, such access control permissions could be administered at different levels of abstraction. To further explain the various levels of abstraction found in the environment, the following example should be considered. Business ABC has various applications, some of which use databases to store information. One such system is the corporate website, which uses an Oracle database to store information. In order for applications or users to gain access to information stored in it, the database needs to be configured to allow access.

This access control could however be used at the application level, not necessarily at the user level. This means that certain access control information is used when an application wants to access the information on the

database. For example, while the user might be logging into the application level with the user identity of James Green, the application level can serve as a proxy and log in as HRAPP into the database.

This can also be seen in the popular SAP system. On the database which stores all the data in SAP, application users do not receive database usernames (Hernandez, 1997, p.349). These user accounts on the database are extremely powerful and are only know to the application server. Users receive usernames which is managed on application level. This allows the SAP system to be portable across many database platforms.

However, various directory services can be found. These directory services add to the profound complexity problem. In Part B of figure 3.3, the manufacturing plant uses Novell's directory service. In the newly acquired business, seen in Part C, Microsoft Active Directory is used. Access control permissions need to be set on both of these directory services. This adds to the complexity of the system because these two systems need to be able to talk to each other to effectively manage access control across the systems environment.

3.4.3 Systems Integration

This existing complex scenario becomes even more complex, and adds more levels to the systems environment, when it is decided to integrate all of these systems. Businesses have realized that their systems should be able to integrate effectively, as well as efficiently, in order to provide not only better functionality to their employees, business partners, but also to their clients. For this reason businesses are putting much effort into integrating their systems. Mische (2002) states that many vendors, as well as consultants, use the word, 'integration'. The problem is that there is no main definition for integration, and therefore, many different types of meanings exist. Many of these definitions are ambiguous. Mische (2002, p.6) mentions that integration is *"The melding of divergent and often incompatible technologies, applications, data, and communications into a uniform information technology architecture and functional working structure"*. This means that many systems being integrated are going to be different from each other. The access control of each system, specifically could possibly be on a different level than that of the other systems.

The system used to integrate these various systems will need a management console to manage the integration of these systems. This management console will need to enable the systems administrator to administer access control permissions. This means that another level, where access control is managed, is added to the already complex environment. It is possible that although access to information could be simplified, the administration burden on the security administration could increase. Questions arise about how integration software manages this process. The reason for this question is because each software system already handles access control. Does the integration software serve as a proxy, meaning that the integration software has all the permissions, and the user only accesses the integration software? This would mean that the user is only able to log onto the system once, and thereafter access is given for each application the user has got access too.

This question relates to the same problem as using only one password on various systems. If an attacker gains knowledge of that password, the attacker can gain access to all the resources which use that password. For this reason, users tend to use different passwords for the various systems they have permissions too.

If the integration software acts as such a proxy, meaning that the user enters a password, and the proxy translates this password into gaining access to the various systems, an attacker can merely attack this integration software to gain access to the various resources within the environment. One such software system which attempts to integrate various systems within the business is called Middleware.

Middleware is a new type of system which integrates various other systems. Mondai and Gupta (2000) defines Middleware as: *"A platform to support integration among different applications, where applications differ in their operation environments consisting of hardware, operating system, networking standard, language of implementation, etc."*. The main idea of Middleware is thus that various resources (different hardware and software systems) within a business could be integrated by means of a single control system. This means that there should be one system which handles the management of the integration and provides the user with one interface, which will enable them to gain access to all the resources they require to do their job.

The problem however is that the environment in which Middleware is deployed could integrate various systems, some of which are on their own very complex systems. Many of the access control permissions in the systems environment might well be on various different layers. All of these complex systems, including the access control permissions to those systems, should be managed from one management system. This raises the question: *"Where is security managed?"*. If access control is managed through a single interface on the middleware product, questions that arise are: *"How does middleware integrate access control?"* and *"What is the level of integration?"*. Many different types of access control paradigms have been developed and are currently being used in businesses. This raises the question: *"What supports the access control paradigms?"*. If middleware integrates 10 existing systems within the business systems environment and each system has its own access control system the question is: *"Where is the access control permissions stored?"*

Another question which is raised is whether a user will then only be asked to log onto one system and then be able to access all the application and systems that a user has access to? This is the basic idea of a uniform identity, which enables users to log onto one application, and that login details filter through to all the other applications. Uniform identities is a great idea, however, it is not that simple. When it comes to a complex scenario, uniform identities suffer from the same problems as the Middleware product. Thus the same questions asked about Middleware, can be asked when talking about uniform identities. However, a question which is not asked is: *"How do we know that a user in System A, is the same person in System B?"*. Directory technologies have been developed which attempt to address these issues. The problem is that these technologies are not necessarily being used or implemented. The reason for this scenario is because security administrators are unable to implement a complete set of access control permissions due to the various levels in such a complex environment.

As discussed earlier, much access control permissions need to be implemented in such a large and complex systems environment. This can be seen in the light of the following example.

3.4.4 Sheer volume of administration transactions

The following is an example of the number of access control permissions which need to be implemented in order to restrain unwanted access to resources. It should once again be mentioned that this is by no means a complete set of the access control permissions which could be implemented.

Assume a company currently uses Role-based Access Control, and has 20 different systems running. On average, an employee uses five of the possible 20 systems. Each employee needs, on average, ten access-control permissions on each system. These permissions will provide employees with the correct level of authorization in order to fulfill their tasks. This means that each employee needs 50 (5 x 10) access-control permissions. If that company has 1000 employees, it means that the administrator of the system is going to need to implement 50000 (1000 x 50) permissions on the system. This could be a daunting task.

One can argue that an administrator of the system only needs to implement this number of access controls once. However, if five percent of employees resign each year, then an administrator needs to disable (1000 x 5 /100) x 50 = 2500 access controls and enable the same number for a new employee. This number of access control permissions which need to be changed, will grow even more, as it is possible for employees to change job positions. This has the effect that access control permissions need to added to the various systems, as well as taken away. This administration of access controls adds to the complexity of the systems environment.

The above mentioned scenario proves that there are many access control permissions, which need to be implemented in a systems environment. This, together with the fact that the system environment is complex and it consists of many levels, begs the question: Who is responsible for implementing these access control permissions?

3.4.5 Human capacity

A business' main goal is to fulfill its mission statement and objectives set by top management. Various departments exist within the business, each contributing towards fulfilling this mission. In order for each department to function, it needs employees. Every employee requires many resources in

order to fulfill his/her job function. In order for employees to gain access to systems within the business, they need to get access control permissions. In normal circumstances, a systems administrator or a team of systems administrators will provide the employee with the correct access control permissions.

However, providing the correct access control permissions can be a tedious job. A complete business with all of its systems, as well as resources, has many functions and configuration possibilities. The problem is that it is difficult to understand all of the elements encapsulated in all of these business systems. This can be gathered from Von Solms and Hertenberger (2005), who mention that there are many functions and configuration possibilities within an Enterprise Resource Planning (ERP) system. If there are so many functions and configuration possibilities within an ERP system, how many possibilities exist inside a complete business system environment? It is almost impossible to understand and know all the elements within this type of system.

This problem is furthermore complicated because employees' main focus areas are different within a business. Two different types of employees exist within a business. On the one side, there are the business-orientated employees. They are more focused on how the business operates. On the other side, there are the technical experts and administrators, whose main focuses are the architecture and structure of the system. The focuses of both these types of employees are different and, as a result, there is a lack of a complete understanding by both parties of what the other does.

Furthermore, because the management of users and roles is complex, the administration of access control permissions is usually done by a small team of security administrators (Munawer, 2000). Due to this fact, it is also impossible for each security administrator within the team to have technical knowledge of all the areas within the business. A problem could thus arise with the implementation of access control permissions, if the systems administrator or the team of administrators, do not understand the processes of the business. It is also possible for a security administrator to not understand the technical side of a specific process. The reason for this being that there is a team of security administrators, and a specific security administrator might not have any experience in the processes where he/she is implementing access control permissions. It is thus possible for an administrator to

Table 3.1: Issues with Access Control Administration

Access Control Issues
Evolutionary Nature of Systems
Different levels of abstraction
Systems Integration
Sheer volume of administration transactions
Human capacity

apply an inappropriate set of access-control permissions to a system.

The profound questions to ask are whether administrators are able to effectively control access to businesses resources, and therefore, can they be trusted and relied upon by the business?

3.5 Conclusion

This chapter explored access control and the administration thereof. It was seen that various access control abstractions were found within businesses. Various problems were highlighted which complicated the administration of access control. These problems are listed in table 3.1.

Based on the problems discussed in this chapter, and shown in table 3.1, proper administration of access control is very complex. Therefore businesses are exploring different alternatives to administer access control permissions. One such solution which businesses are exploring is automated provisioning. The author believes that automated provisioning could provide the means to solve the access control administration problem.

Chapter 4

Provisioning

Chapter 2 discussed the drastic changes in the business world over the last three or four decades. In particular a move towards knowledge-based operations was identified. This move requires that employees be empowered to handle these changes. One such way of empowerment is through the ability to access information, thus resulting in better decisions.

Although Chapter 3 discussed a variety of access control paradigms, it is interesting to note that access control remains a fairly "negative" activity: preventing people from gaining access to resources. It would not be unfair to say that, until now, access control has a "keep away" rather than "enablement" feel.

One could therefore argue that access control is currently not geared towards supporting knowledge-based businesses. Even so, some moves to remove the human bottleneck from access control administration are evident and worth noting. One such move, as mentioned in Chapter 3, is the establishment of a uniform identity. This allows employees to sign in only once per session, whereafter their identity will be known throughout that session to the various resources in their business.

Another move by software vendors is the introduction of a new breed of software systems, called provisioning systems. Provisioning systems rely on uniform identity and uses workflow management techniques to assist in the automation of access control administration.

This chapter will introduce provisioning as a concept in the following way. Section 4.1 will briefly introduce "provisioning systems" from a product perspective. Thereafter, the chapter takes a step back by defining the

term "provisioning" more formally in section 4.2. Section 4.3 explores the relationship between access control and provisioning, arguing that provisioning systems may represent a significant move towards a more positive view on access control.

Section 4.4 discusses technologies and standards prevalent to provisioning systems. However, the rise of provisioning systems must first be discussed.

4.1 The rise of provisioning systems

Provisioning systems is about "providing" resources to employees. Several products have been launched and are classified as provisioning systems. This section is not intended to serve as a comprehensive view of the marketplace, but merely points to the emergence of these type of products. It does, however, attempt to identify the commonalities between the various products.

Consider, for example, IBM Tivoli Provisioning Manager (ITPM). Various manual tasks can be automated by using ITPM. According to Aggarwal, Atakan, and Boyce (2005) these tasks include the provisioning, as well as the configuration, of: "servers and virtual servers, operating systems, middleware applications, storage and network devices acting as routers, switches, firewalls, and load balancers". ITPM enables a business to use predefined automation packages for the products of major vendors. For example, it allows integration with Citrix Password Manager, which allows administrators to pre-populate users' secondary credentials in the Password Manager Central Store (Citrix Password Manager, 2006). However, if the packages supplied do not give the functionality needed, customized automation packages can be developed for specific businesses. This enables the business to embed policies and procedures regarding access control privileges into the provisioning manager.

Another product by Courion, which automates user provisioning, is called AccountCourier (User Provisioning, 2006). AccountCourier is able to enforce security policies, and eliminate repetitive user management tasks. Any operating system, application, Web portal, or other IT asset can be instantly granted, revoked or modified by a system administrator.

From the two examples above it can be seen that provisioning systems

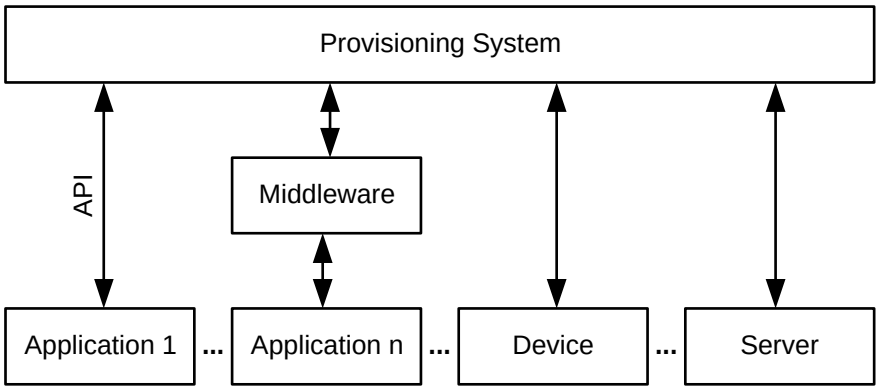

Figure 4.1: Provisioning Systems move Administration function away from application

move administration of access control to a level away from actual application programs. It does this through pre-defined application programming Interfaces (API's) or through additional "middleware". This is depicted in figure 4.1.

In addition to moving administration to a new level, provisioning systems also provide functionality to implement customized workflow to incorporate organization best practices and procedures.

Provisioning systems, obviously, deal with more than access control rights, and incorporates functionality such as software distribution and patch management, that, while related, are not logical access control in the sense that this dissertation views access control.

It is interesting to note that Provisioning is a term which is widely used in various disciplines. This calls for a more in depth investigation into the meaning of the term.

4.2 Provisioning: A definition

We attempted to understand the term "provisioning" better by considering a dictionary definition.

According to the Merriam-Webster Online Dictionary (2004) provision is "the act or process of providing; the fact or state of being prepared beforehand; a measure which is taken beforehand to deal with a need or contingency". Furthermore, the Latin roots of the word eludes to "foresight".

From a linguistic perspective then, we can say that provisioning implies that we should have the foresight to prepare to deal with any need that will occur in future.

"Provisioning", as a term, is also encountered in the financial disciplines. The South African Institute for Chartered Accountants (SAICA's) (SAICA Web Site, 2004) has various standards explaining and defining certain concepts. These statements are collectively known as the AC statements. The concept of provisioning is defined in the IAS37 (International Accounting Standards Board (IASB), 2005, p. 1595) statement, previously known as the AC130.

The IAS37 statement defines provisions as "liabilities of uncertain timing or amount". In order to fully understand the definition of a provision, the definition of a liability needs to be addressed. The International Accounting Standards Board (IASB) (2005, p. 1595) statement defines a liability as "a present obligation of the entity arising from past events, the settlement of which is expected to result in an outflow from the entity of resources embodying economic benefits."

From the accounting perspective, a provision is therefore an obligation one has because of a certain event, that occurred or will occur, which will benefit another party.

On a more technical note, the OASIS Provisioning Services Technical Committee (PSTC) has adopted the following formal definition in order to explain the general term. "Provisioning is the automation of all the steps required to manage (setup, amend and revoke) users or system access entitlements or data relative to electronically published services." (Rolls, 2003a).

Based on these views, for the purposes of this dissertation, provisioning will be defined as:

> *Provisioning is the automated process of managing resource obligations.*

From the perspective of this dissertation referring to "provisioning" therefore implies adherence to the following salient properties:

Automated. Provisioning is done in an automated fashion. No human interaction, or as little as possible, is required.

Process. Provisioning is a process, rather than merely a single action, and has a start and an end.

Managing. Managing involves the actions of giving permissions, changing permissions and revoking permissions.

Obligations. Businesses must empower employees to do their job.

The question which can be asked is what makes the author's definition different? The author's definition specifically focuses on the enablement or empowerment of employees in order for those employees to become knowledge workers. The author believes that this enablement should be prepared beforehand and then automated, thus enabling employees to have the correct set of access control permissions, when they require them. Thus, there is an obligation towards employees to provide them with the information required to do their jobs.

Of course, at the same time, as highlighted in section 2.3.3, the business has a responsibility to government and its shareholders to protect information appropriately.

The four aspects listed above show a close relationship to those found in the cycle of enabling and disabling of access to resources. Therefore provisioning, and its use together with access control, should be investigated.

4.3 Provisioning and access control

A business has many obligations, which affect the community. These obligations include those towards the shareholders of the business, which is to make a profit that will then be divided between the various shareholders. Due to this obligation, shareholders should invest more money into the business, enabling it to grow.

Another obligation of the business is towards the customers. This obligation is to provide the customer with the best possible service. The result of this obligation is that the customer will return to the business in future.

The business also has an obligation towards its employees. This obligation is to provide the employees with the means to effectively and efficiently do their jobs. Finally, the business also has an obligation towards the community. Due to the success of the business, the community is able to grow,

start new businesses, provide new work opportunities and form a better community.

In some instances, the results of these obligations are that the business needs to provide services. For example, in order for an employee of business ABC to deliver an item to a customer, he/she needs to have appropriate transport. The business should therefore supply the employee with transport in order to deliver that item. The business thus has an obligation towards the employee to supply him/her with the necessary transport to make the delivery. However, the business does not only have an obligation towards the employee, but also towards the customer, to deliver the item on time.

If the business has an established knowledge worker culture, it has obligations to these employees who need access to appropriate information. Businesses should see it as an obligation to provide employees with the necessary access to information. This dissertation focuses on the access control obligations which exist within business. Therefore it is necessary to investigate the management of access control obligations within business.

As depicted in figure 4.2 there are many resources which are provided by businesses to their employees. Each of these resources is implemented because of the obligation which the business has towards the business, employees, stakeholders and customers. These resources may include email, Internet access, access to the corporate portal, the finance system, as well as access to a finance-building door, ID badge, an office, a desk, a chair, a phone, a notebook computer.

For example, it is very important that while Sue works for the company, she needs to have access to these provisioned items. However, when Sue resigns or changes jobs within the business, these provisions made when she was appointed should be re-evaluated. If Sue resigns from the business, each and every access to a resource should be de-provisioned. This means she should no longer be able to gain access to any of the information or resources she previously could gain access to.

So it is evident that there are times when the provisioning processes is initiated as well as times when it is terminated.

The provisioning process has various stages. These stages are shown in figure 4.3. Consider the following example. Business ABC has decided to appoint a new accountant, who requires an office, furniture and equipment,

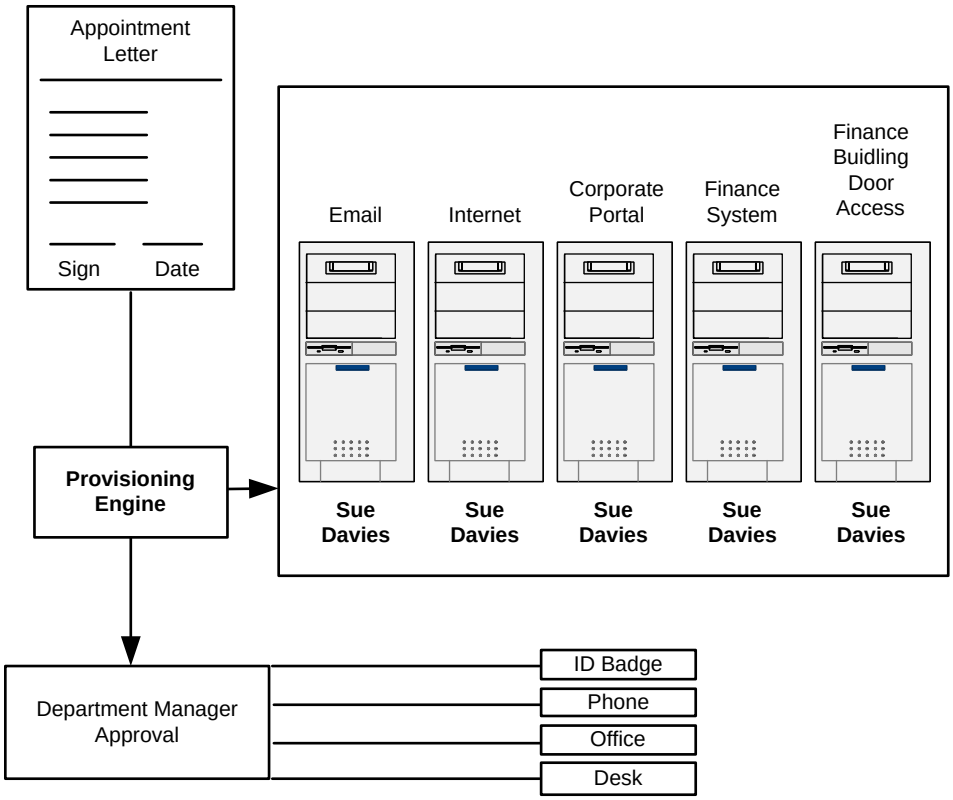

Figure 4.2: A Provisioning Example

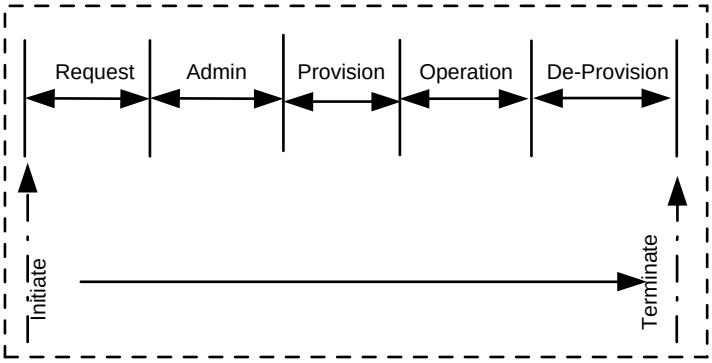

Figure 4.3: The Provisioning Process

including a notebook computer. At the end of the recruitment process, a new accountant is appointed. After the new accountant takes up the position, he occupies his office and takes possession of the equipment. The new accountant has thus been provisioned with the necessary equipment, which he can start using. This is indicated by the operation phase in figure 4.3.

However, it is very important that once the accountant decides to leave the business, all equipment issued to him should be returned to the business. This is the phase where the original provision is de-provisioned. However, on appointment the accountant will not only be supplied with the equipment he requires, but he needs to have access to all the information necessary to do his job.

Figure 4.4 depicts the process of enabling the appointed accountant to gain access to information. Based on the job function which that accountant is doing, he/she will need access to information. Once the accountant has

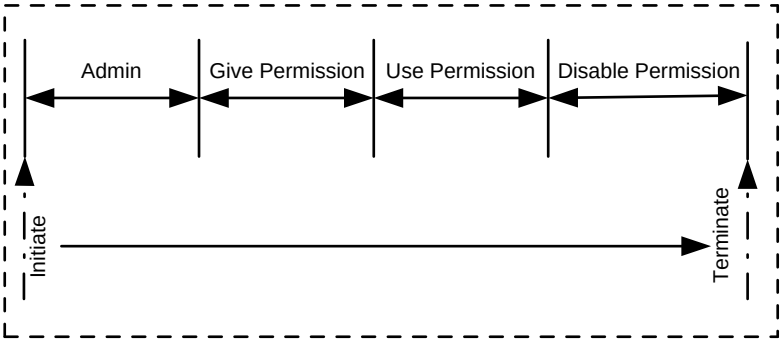

Figure 4.4: The Access Control Process

been appointed, the security administrator will need to identify what access the specific accountant will need. This can be seen as the administration phase within the access control process. Based on this decision, the security administrators will give the accountant the necessary access control permissions by associating the permissions to resources with the specific accountant. This can be seen as the phase where the permission is actually given. Thereafter, the accountant can use those permissions given to him/her.

Once that accountant decides to leave the business, or his/her contract expires, all of those access control permissions need to be disabled. The accountant will no longer have access to any business resources.

If figures 4.3 and 4.4 are examined, it can be seen that provisioning and access control share some commonalities. Considering the first phase of provisioning and access control, both have an administration phase where it is decided when to give what access to whom. If one considers the same example of the accountant, the business had to decide that the accountant needed an office, other equipment, a notebook computer as well as access to specific information. Thereafter, the equipment, as well as access to information, were given to that accountant. This equipment was thus provided to the accountant by the business. Access to information was also provided to the accountant in order to do his/her job. Finally, once the accountant left the business, the equipment needed to be returned to the business. The security administrators have to disable all the access control permissions of resources that the accountant had in the business. The accountant is thus unable to use both the equipment as well as the access he/she had to resources within the business.

Figure 4.5 depicts the access control administration lifecycle. This lifecycle starts when access control permissions are granted and end when these permissions are revoked.

Therefore, there is a direct correlation between provisioning systems and the way access control permissions are granted to users of business resources. This correlation is, however, not a one-to-one mapping. Access control is a kind of provisioning, possibly maintaining logical access control to information, rather than physical access to equipment.

Now that we have established that access control administration and the possible automation thereof indeed falls within the ambit of provisioning

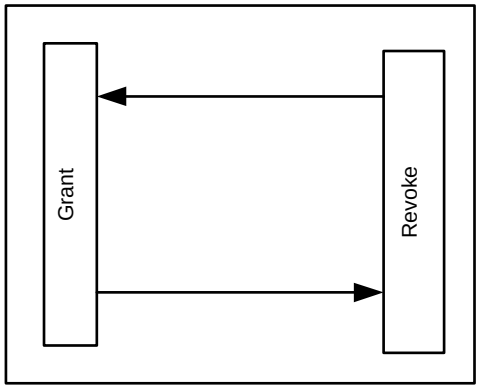

Figure 4.5: Access Control Lifecycle

systems, we consider provisioning from a technological perspective.

4.4 Provisioning System Technologies

While section 4.1 argued that current provisioning system share certain commonalities, implementation details differ. However, for future reference, it may be important to note that various standardization efforts are in place. While some are directly related to provisioning, other fall within the more general ambit of access control and security. In particular three of these standards are:

SPML Service Provisioning Markup Language

SAML Security Assertion Markup Language

XACML Extensible Access Control Markup Language

Each of these technologies and its relevance to provisioning will be briefly explained in the subsections that follows.

4.4.1 SPML

SPML is a XML framework proposed by OASIS (Rolls, 2003b). SPML facilitates the requests for various services made by clients of the provisioning system. A high level figure of such a SPML system, containing all the basic operational components, is depicted in figure 4.6.

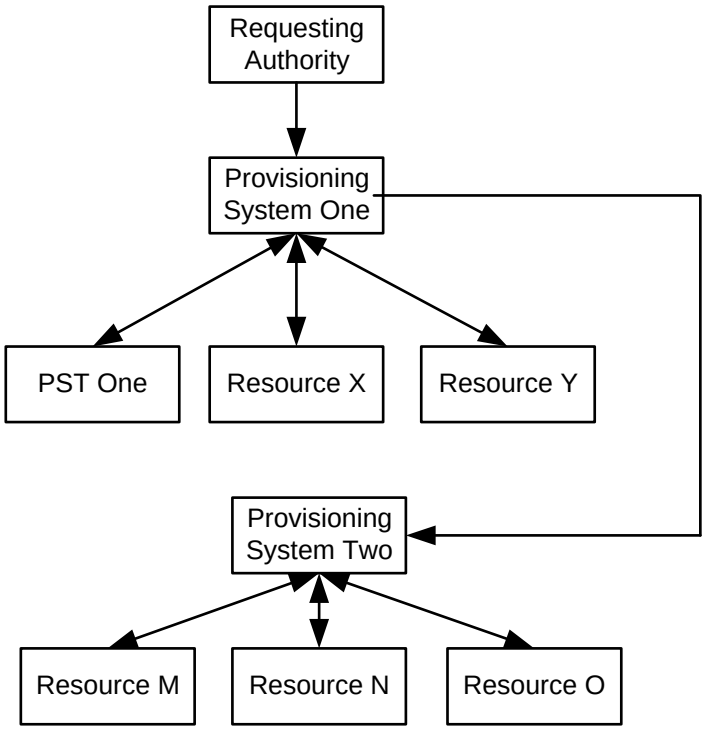

Figure 4.6: SPML Provisioning System

There are various parties or systems which can be identified in figure 4.6. They are the Requesting Authority, Provisioning System One, Provisioning System Two, PST One as well as various resources. (Rolls, 2003b) defines the Requesting Authority as a: "Party or system that is authorized to request a resource for the party."

Firstly the Requesting Authority, the client, generates a request for a service. This request can only be for a service from a predefined list. This request is in the form of a SPML document. This SPML document is then passed to Provisioning System One. The data contained in this SPML document is then used by Provisioning System One to construct another SPML document. This SPML document is then send to PST One, a resource, which provides a service based on SPML requests.

Based on the original request by the Request Authority, Provisioning System One decides that PST One cannot provide all the necessary information. Therefore, Provisioning System One sends a provisioning request to Provisioning System Two. Provisioning System Two also offers provisioning services. One such service offered is the company PBAX service. In this

scenario the Authority Requester requested the telephone line be redirected. Therefore, Provisioning System Two requests this service from a specific resource. However, a simple command string was send to the PBAX, in order to redirect a specific telephone line.

In this example it can be seen that both the Request Authority as well as the Provisioning System generate SPML messages. However, because the PBAX does not provide a SPML interface, another form of connection needs to be established.

SPML has three major systems elements. They are the Request Authority (RA), Provisioning Service Points (PSP) and the Provisioning Service Target (PST).

In order to obtain a service the Request Authority software components issue a well formed SPML request. These requests are issued only to known SPML service points. These service points, called the Provisioning Service Points, listen for these requests and respond to them. However, the PSP will only respond if the request is a well formed SPML request and if the request comes from a known SPML RA.

In order for computing entities to "know" each other it is important that authentication and identification information be exchanged. The next standard, SAML, may insist in this regard.

4.4.2 SAML

SAML is a standard by OASIS, based on XML, which includes a language, as well as a flexible and extensible protocol (Madsen & Maler, 2005). This framework enables computing parties to communicate security, as well as identity (Madsen, 2005). Therefore, SAML enables an organization to share resources, while required authentication and authorization information is exchanged (Ramakrishnan, 2004).

SAML consists of various components. They are Assertions, Protocols, Bindings and Profiles. The first component, assertions, is used to enable authentication and authorization (Cantor, Kemp, Philpott, & Maler, 2005). According to the OASIS Glossary (Hodges, Philpott, & Maler, 2005) an assertion is: "A piece of data produced by a SAML authority regarding either an act of authentication performed on a subject, attribute information about the subject, or authorization data applying to the subject with respect

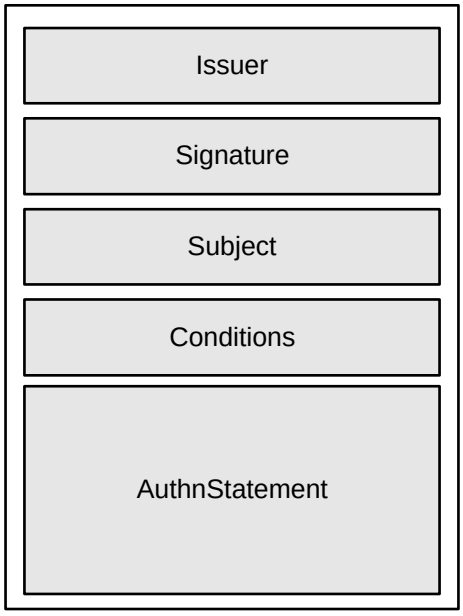

Figure 4.7: SAML assertion (Madsen & Maler, 2005)

to a specified resource" Therefore, SAML authorities generate or exchange assertions with subjects. A subject is defined as: "A principal in the context of a security domain." Where a principle is defined as: "A system entity whose identity can be authenticated."

Therefore, a SAML assertion is merely messages, or statements which flow between an authenticated system and a SAML authority. A high-level structure of such a typical SAML authentication assertion is depicted in figure 4.7.

There are three types of assertion statements defined in the SAML specification: They are Authentication, Attribute, and Authorization Decision.

An authentication statement defines how and when an assertion subject was authenticated. In order to associate the assertion subject with the given attributes, an attribute statement is used. An authorization decision is the result of the request by an assertion subject for access to a resource. Various requests as well as responses are defined in SAML by means of protocols. Bindings are the components which are responsible for mapping SAML request/response messages to standard messages.

The final component, profiles, is responsible for defining constraints and/or extensions to support the SAML when a specific application is used.

SAML provides various advantages. Because the security framework is no longer based on a specific vendor or platform, SAML provides *Platform neutrality*. No user information needs to be stored or maintained. Therefore information does not need to be synchronized to any directories. The browsing experience by users will also improve based on the fact that they only need to sign on once.

Since provisioning systems would encounter many users and many other heterogeneous systems, a platform neutral standard can be used very effectively to convey authentication and authorization information between various systems.

From a perspective of heterogeneity, the expression of access control policies also are challenging. Next, we discuss XACML, an OASIS standard which facilitates the expression of access control policies in heterogenous environments.

4.4.3 XACML

An organization's security policy outlines and discusses the elements and enforcement points which exist within that organization (Moses, 2005). However, the setup and maintenance of such a security policy, especially in a large organization, can become complex. Therefore, a need exists for a common language to articulate this security policy. One such policy language which has been standardized by OASIS is the eXtensible Access Control Markup Language (XACML), based on XML (Godik, 2003). XACML provides a mechanism that enables various applications and environments to gain access to many policies through a single interface (Lorch, Proctor, Lepro, Kafura, & Shah, 2003). In order to respond to authorization decision requests, XACML also specifies a request and response format (Ramakrishnan, 2004). XACML, therefore, provides a foundation on which organizations can build their own solutions. However, XACML cannot be seen as a complete authorization solution.

Figure 4.8 depicts the elements of XACML. The figure also shows all the various steps XACML follows. These steps will now be discussed in more detail.

Firstly, policies and procedure sets need to be compiled. These policies and procedure sets are then made available to the Policy Decision Point

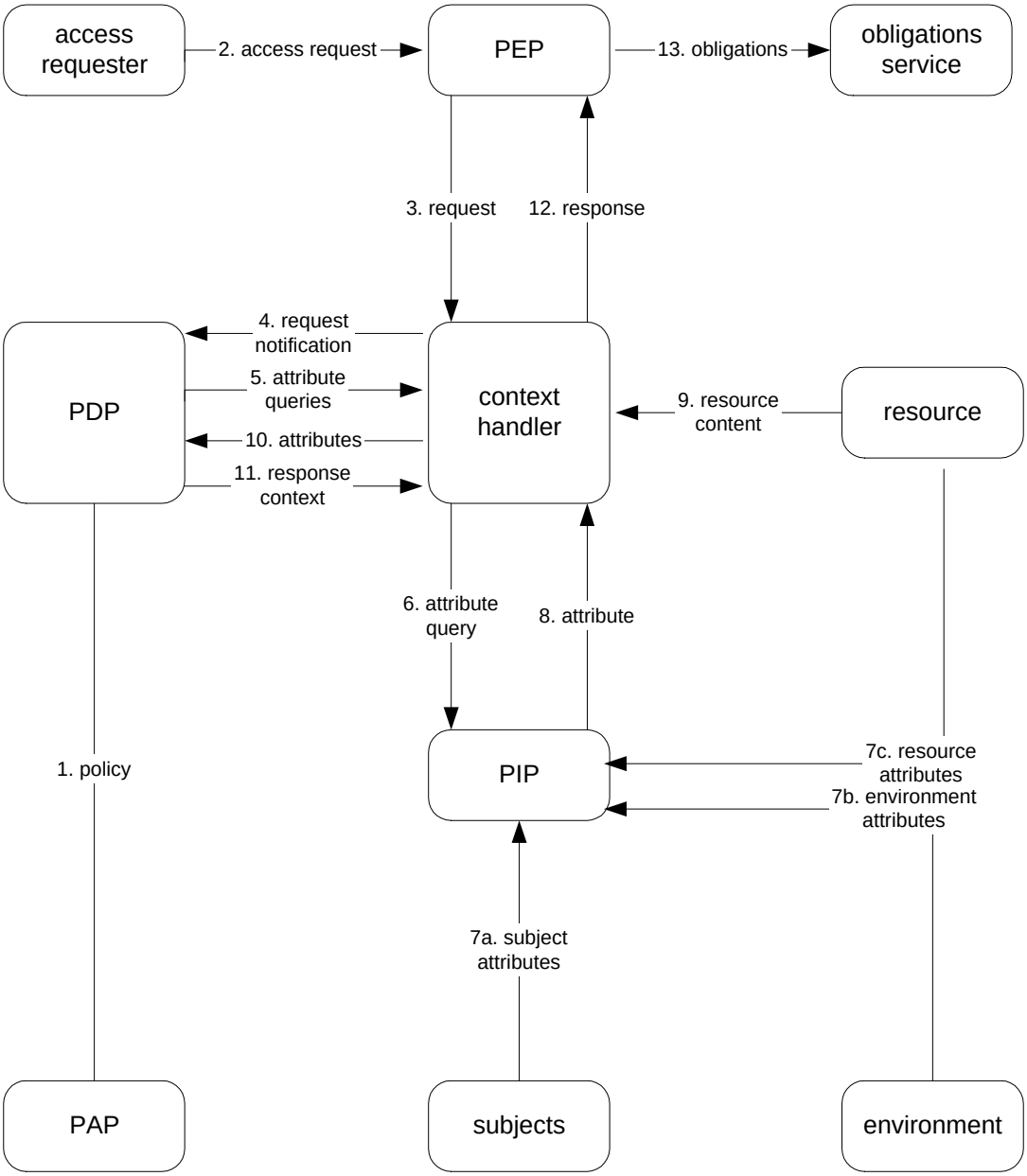

Figure 4.8: XACML

(Moses, 2005)

(PDP) by the Policy Administration Point (PAP).

After policies and procedures have been compiled, it is possible for the Policy Enforcement Point (PEP) to receive requests for access. The PEP will then send this request for access to the context handler. A XACML request context is constructed by the context handler, and sent to the PDP. If the PDP requires any additional information; it will request that information from the context handler. In this case, the context handler will request the information from the Policy Information Point (PIP). The PIP will acquire the requested information and return that information to the context handler.

The context handler also has the option to include the resource with the information that was requested. The information requested, with the optional resource, is then returned to the PDP. The PDP will then evaluate the policy.

Based on this evaluation, the PDP will return a response context to the context handler. This response context will include the authorization decision by the PDP.

Next, the response context is translated to a native response format of the PEP by the context handler, and sent to the PEP. This response will include the obligations the PEP has. These obligations are the actions which need to be performed by the PEP in conjunction with the enforcement of an authorization decision.

It is then the PEP's responsibility to see that the obligations are executed. This means that if the response was that access is permitted, that access is given to the requester. Otherwise the PEP will deny the access to the resource.

In order for a provisioning system to function properly it needs to make use of policies. Therefore XACML is the perfect technology to facilitate the setting up of policies within the provisioning system. This will restrain unwanted access to resources, but even more important allow access where applicable.

All of these technologies help developers to implement provisioning into the business. However, in order to provision for access control a complete list of access controls, which could possibly be used, needs to be implemented. As mentioned earlier, analyzing the business and setting up a list of the access control rights that need to be implemented can prove to be a problem.

4.5 Conclusion

This chapter upheld provisioning as an automation facilitator for access control administration. While it is clear that logical access control is only one aspect of provisioning systems, it is argued that it plays an important part.

Since, in an environment where access control requirements may change regularly (such as when dealing with knowledge workers), automation seem to be the only way out, it is predicted that provisioning systems will play an important part in businesses in future. A brief look at some of the standardization issues confirmed that work in the general area of provisioning and access control is indeed happening and has drawn the attention of standardization bodies.

This chapter has then completed our look at the environment in which our research question is set. Chapter 2 has confirmed that a need for an "enablement" perspective on access control exists, while cautioning that regularity and legal responsibilities also constrain the "openness" of access control. Chapter 3 then explored access control paradigms and the administration of access control. It identified the direct need for automation of access control administration. The current chapter placed our bets on provisioning systems as being the platform for this automation.

However, the problem exists that there is no guidance that specifically assists with analyzing access control requirements within the context of provisioning systems. With this background, the next chapter, Chapter 5, will introduce the agreement abstraction, which is a vehicle that will facilitate the analysis of access control requirements within the context of provisioning systems.

Chapter 5

The Agreement Abstraction

Chapter 2 stated that the business world has changed significantly over the last couple of decades. These changes are characterized by an increased dependency on information. More and more employees are becoming knowledge workers.

However, Chapter 3 argued that access control mechanisms are not very supportive of such knowledge-based environments. These environments are characterized by an extremely high administrative load with respect to access control activities. The situation is exasperated by the complexities of real-world systems.

Chapter 4, subsequently, introduced the concept of "provisioning". Essentially provisioning aims to automate some of the access control administration. However, no work was done on how such environments should be administered, specifically how access control requirements translate to provisioning system configuration.

In a bid to assist in this regard, this chapter introduces the agreements abstraction as a basic building block for access control analysis The chapter commences with section 5.1 discussing the required traits of the agreement abstraction. Thereafter the term "agreement" is discussed from various perspectives. Finally section 5.3 formally defines the agreement abstraction.

We begin by discussing essential traits of the new agreement abstraction.

5.1 Required traits of the new abstraction.

Existing access control abstractions force the security administrators to manually grant and revoke access to a resource. In Chapter 3 it could be seen that the administration of access control becomes more of a problem as the size of the organization increases. The workload of the security administrator thus also increases. As the workload of the security administrator increases, so does the likelihood of errors occurring.

In this dissertation the premise is that access control paradigms administration per se is not the problem. Many systems exist, using a variety of access control paradigms; these cannot easily and realistically be replaced. In the development of the abstraction we see access control administration therefore as a process on its own. Hence the process must be managed in its own right. The agreement abstraction therefore aims to peacefully co-exist with existing access control mechanisms and is independent of the underlying access control paradigm. Note that this view of seeing access control administration as a process is fundamental also to provisioning systems as discussed in Chapter 4.

To set the scene further, existing paradigms will now be discussed, specifically with a view to extract the commonalities between them and to enumerate the required traits of a new abstraction.

Fundamentally problems exist because businesses tend to rely on fixed "structures" to enable or restrain access to resources. In order to gain access or restrain access, that access is linked to a resource or a user. Chapter 3 discussed various access control paradigms. Consider the notion of "fixed structures" in terms of three widely recognized access control paradigms. Firstly, consider access control based on labels. Within this paradigm, a user is able to gain access to information based on the "label" associated with him/her and the "label" associated with the information.

These labels are static, i.e. they are associated with users and objects and rarely changed. Similarly when access control based on ownership is used, access is based on the fact that the user is either the owner of the document, or access is granted by the owner of the document. The access to the document is thus attributed to the existence of the document and who was responsible for creating the document. Again, fairly "fixed" information.

Table 5.1: Existing Access Control focuses fairly fixed information

Access Control Paradigm	Main focus
Labels	Based on fixed lattice of labels
Ownership	Based on creator not actual owner
Roles	Based on the business structure

Within role-based access control there is a strong focus on the role concept, which relates to the businesses' organizational structure. Again, organizational structures are fairly fixed; although they may change, this is rare.

Table 5.1 summarizes these access control paradigms and shows how they share a focus on relatively fixed information.

While reliance on relatively fixed information has the advantage that it requires "less" administration[1], it has the disadvantage that where access rights are difficult to predict, such as with knowledge workers, the administrative load would grow tremendously.

While searching for a suitable abstraction, one of the guiding principles was the assumption that we can programmatically manipulate the access control administration tasks and need not concern ourselves with the fixed structures underlying the access control mechanism. Our thinking rather concentrated on the goal of any given access control request. In other words, why do the user need access? This led to a positive empowerment view of access control.

This supported the notion from Chapter 2 that businesses that want to be successful must create knowledge. To this end, a culture of knowledge working must be established whereby workers are empowered to create knowledge within the business. Should an employee be unable to gain access to information, he/she will never truly become a knowledge worker. It is thus imperative that businesses trust knowledge workers with the information they have. Of course, this trust does not imply that employees should not be audited on their behavior and actions.

Two forces which work against each other can be identified. The one force

[1]Granting and revoking rights will happen only when the "fixed" nature of the information changes. For example, in RBAC when organizational structure changes or staff move between jobs, changes will be made. However note that "less" does not mean "little".

Table 5.2: Traits required from a new abstraction

Traits of Existing Paradigms	Required Traits
Static task (admin)	Process in its own right
Focus on Structure	Focus on goal
Preventing unauthorized access	Enabling required access
Human Activity	Automated activity

is the status quo: restrain all possible access to resources. The other force: enable employees of a business to have sufficient information at hand. It is of great importance that a business seeks a balance between these two forces. This balance should be between giving access to employees, thereby empowering that employee, and restraining access to resources, thereby protecting the business.

However, in order to have this balance, a great deal of effort needs to be placed on the administration of access control. Chapter 3 discussed some of the problems involved. This included problems associated with human involvement. The more humans are involved in the administration of access control, the more likely it will be that an inefficient set of access control permission could be implemented. Similarly, the more repetitive the task, the higher the chance of human error. Therefore an effort should be made to try and automate this administrative process.

Table 5.2 summarizes the required traits of a new abstraction as compared to the traits of existing access control paradigms.

The following section introduces the new abstraction by answering the question: "What is an agreement?".

5.2 What is an agreement?

The term "agreement" is widely used in society. In some cases it is used as an informal term, whereas in others, such as the legal world, it is seen as a very formal term. According to the Merriam-Webster Online Dictionary (2004) an agreement is:

1. the act or fact of agreeing

2. an arrangement as to a course of action

3. a contract duly executed and legally binding

Implied in this definition is the involvement of more than one party. Furthermore, words such as "course of action" strongly suggests activity, in other words, something that will or should happen. Since the notion of "contract" has a legal feel to it, we investigated what an agreement is in the legal discipline.

In the legal discipline, Sharrock (2002, p. 59) defines an agreement as "a meeting of mind on all aspects of a transaction". A transaction, typically, is between two parties. The first individual, called the offerer, will make an offer. The second individual, called the offeree, needs to accept this offer in order for an agreement to exist. In order to complete the agreement, both parties, the offerer, as well as the offeree, needs to specify certain terms and conditions. These terms and conditions are then known as the offer.

When the offeree accepts this offer, the offeree agrees to all the terms and conditions. Acceptance of the offer constitutes the establishment of an agreement. When all the objectives of the agreement, as stated by the terms and conditions are met, the agreement is finalized and ceases to exist.

It is interesting to note that if we consider legal agreements, they correspond in many ways to the provision of access controls of a business to its employees. Consider therefore the legal constraint (in italics) and a brief discussion of how these relate to access controls in business.

An offer needs to be made by one of the parties involved.

In order for access to be given to an employee, either the business should instruct the employee to do a specific task, or that employee should indicate that he/she needs to do a specific task.

This offer needs to be communicated to the other party.

It is necessary that a form of communication should have taken place between the business and the employee. This communication could be the worklist of a workflow management system, a memo, or an email which was received by either party. It could also be a formal contract such as appointment contract.

The offer may not have already finished or been canceled in the past.

Employees should not be allowed access to resource for tasks which they don't need. This implies that access to information should be revoked when not necessary anymore.

The offerer needs to accept the offer as stated by the offeree.

In certain cases the employee accepts an offer by signing a contract such as an employment contract. In other cases an employee will accept the offer based on a button which is pressed or a task selected from a worklist. Regardless of form, it is important that this acceptance can be audited specially when considering the next guidance.

The offeree must have notified the offerer of accepting the offer. This should be done as arranged by the offerer.

The business communicates with the employee by providing him/her the access he/she requires. However, the employee should only be able to access the resources while it is necessary to do so. After that the permission should be revoked.

The offer is specified in terms of terms and conditions. The question "What is terms and conditions?" thus begs to be asked.

According to Merriam-Webster Online Dictionary (2004) a *term* is: "provisions that determine the nature and scope of an agreement" and a *condition* is "a premise upon which the fulfillment of an agreement depends". In other words, the terms and conditions of an agreement is where the commitment for *both* parties is determined before that agreement is made. This commitment could include what needs to be done, the time frame, the responsibilities of the parties involved, as well as the complications if either party does not perform under the agreement. The terms and conditions thus protect each of the parties involved in the agreement.

From an access control perspective a term identifies what resource specific access rights are to be given to the user. The conditions will specify the circumstances under which the agreement will cease to exist.

For an example of terms and conditions consider the following. Tom, a section head, has asked William to help on a special project. Tom explains to William that he is only needed during the planning phase and that they should be finished by a certain date from now. The terms of this agreement is that William will be getting access to project specific information not generally divulged. The conditions specify that William only take part up to the stage where the planning of the project is completed and that the planning phase should not pass the specified date.

In the agreement abstraction we are thus going to state agreements in terms of terms and conditions. The offer will specify the terms and conditions of the agreement. From an access control perspective the required permissions are the terms. The condition specify when the agreement is finished[2].

From an access control perspective we therefore see agreements to have the following salient properties.

1. An agreement involves the business as the offerer.
2. An agreement involves the employee as the offeree.
3. The terms of an agreement specifically define the scope, i.e. what the access control significance is.
4. The conditions of an agreement specify when the agreement is fulfilled, that is when the access control changes should revert to their previous state.
5. The establishment and terminations of agreements can be audited.

The next section will now formally define the agreement abstraction and its components.

5.3 Agreement Abstraction Defined

In the business systems environment various events can be found. The number as well as the type of events which is found is dependant on the resources found in the environment. We abstractly represent these events as:

$$E = \{e_1, \cdots e_n\}$$

In a similar way, agreements can be enumerated by a set A.

$$A = \{a_1, a_2, \cdots a_m\}$$

[2]Although this dissertation does not formally model obligations (as discussed in Chapter 3), it may be worthy of future work to consider the relationship between obligations and terms and conditions more formally.

However, only some events will correspond to the instantiation of an agreement. Not all events result in agreements. We can represent this relationship through the relation *AE*.

$$AE \subset A \times E$$

As learnt from the legal discipline, agreements are often dependent on other agreements. For example, you cannot go into an agreement about additional project responsibilities, unless there is another agreement which governs the contract of the employee. Therefore, we define

$$AH \subseteq A \times A$$

a partial order on A called the agreement hierarchy. For convenience, we also adopt the shorthand notation $\succeq$. Therefore, if $a_i \succeq a_j$ we imply that a_i is a parent agreement of a_j. In other words, a_j must be evaluated in terms of a_i.

For example, in order for a *LeaveAgreement* agreement to be formed between an employee and his/her employer, a *WorkAgreement* agreement needs to exist between the two parties. Therefore the *LeaveAgreement* agreement is a child agreement of the *WorkAgreement* agreement, and the *WorkAgreement* is the parent agreement. Or, in mathematical shorthand, *WorkAgreement* $\succeq$ *LeaveAgreement*.

If we consider a specific agreement a_i, then a_i can be represented as a four tuple

$$a_i = (x, y, T_i, C_i)$$

where x is the offerer and y the offeree, T_i is the set of terms and C_i the set of conditions.

$$\begin{aligned} T_i &= \{t_{i1}, t_{i2}, \cdots t_{ik}\} \\ C_i &= \{c_{i1}, c_{i2}, \cdots c_{ip}\} \end{aligned}$$

The following subsections will formalize terms and conditions.

5.3.1 Agreement Terms

A term indicates the provision made, which determines the nature and scope of an agreement (Merriam-Webster Online Dictionary, 2004). Agreement a_i

has an associated set of terms T_i, defines as:

$$T_i = \{t_{i1}, t_{i2}, \cdots t_{ik}\}$$

For agreements we identified three basic terms: *grant*, *revoke* and *amend*. Therefore,

$$t_i = \begin{cases} grant(s, R, u) \\ revoke(s, R, u) \\ amend(a, T', C') \end{cases}$$

A *grant* implies that access permissions are given to a user, enabling that user to gain access to a specific resource. The grant term is defined in terms of parameters. These are: the system which is affected (s), the set of permissions which needs to be granted (R) and the user (u) which is granted the permissions.

The permissions which can be granted to, or revoked from, a user is abstractly represented by the set R.

$$R = \{r_1, r_2, \cdots r_n\}$$

The semantics of the underlying system will determine the practical implementation of the access rights. A *revoke* implies that access control permissions is removed, thereby restraining a user's access to a specific resource. As with the grant term the revoke term is identified in terms of the system (s) that is affected, the set of permissions which needs to be revoked (R), and the user(u) whose permissions is revoked.

An *amend* refers to changes made to an existing agreement. Therefore the amend term is defined in terms of the existing agreement a, that must be amended. Furthermore the new set of terms(T') and conditions(C') for the agreement being amended is provided.

By its very nature an amend assumes the existence of a hierarchy of agreements. This relating can be formalized by stating:

$$t_i = amend(a, T', C') \Rightarrow a \succeq a_i$$

Consider agreement conditions in further detail.

5.3.2 Agreement Conditions

Conditions specify when an agreement ceases to exist. For agreement a_i these conditions are represented as:

$$C_i = \{c_{i1}, c_{i2}, \cdots c_{ip}\}$$

Any condition, c_{ij} is a statement in first order predicate logic. Each condition will therefore return either a true or a false. Or, more formally:

$$\{0, 1\} \leftarrow c_{ij}$$

Therefore, an agreement will define one or more conditions. Since this abstraction will eventually be implemented in systems, the condition is typically that an event has happened. Therefore, we define the function *hashappend(e)* to return true when event e has happened. A condition is of the form:

$$c_{ij} = hashappend(e)$$

We assume that we can track when a specific event occurs. Now that we have formally defined the agreement abstraction the following subsection reviews an example of an agreement.

5.3.3 Example

In hypothetical business XYZ various agreements have been identified. Some of these agreements are shown in A.

$$\begin{aligned} A &= \{a_1, a_2, a_3, \cdots\} \\ &= \{\text{WorkAgreement}, \text{TakeLeave}, \text{RecallLeave}, \cdots\} \end{aligned}$$

In the above, three agreements are identified as being elements in the set of agreements A. The first agreement, *WorkAgreement* is formed when the business hires employees[3]. Agreement *TakeLeave* occurs when employees take leave. Lastly, agreement *RecallLeave* exists when an employee's boss asks the employee to come back to work earlier.

[3]Note that the naming in this example is to illustrate the concept. In real business many different types of WorkAgreement will exist. The examples furtheron in this dissertation use easy to read names rather than implying that, for example, there will only be one type of WorkAgreement

In describing these agreements we must identify events that will trigger their existence. These events are shown to be part of the set E.

$$E = \{\text{DateReached}(d), \text{LeaveFormApproved}(f), \text{LeaveRecall}(r), \cdots\}$$

In general events are abstract concepts, but in this example they relate to anecdotal examples. Note that since agreements deal with "classes" of events, we indicate some events in a parameterized manner. In this case *DateReached(d)* would be dependant on date d and *LeaveFormApproved(f)* will depend on the actual form *f*. The parameter value will only become clear when an actual agreement is instantiated.

The relation AE would associate events and agreements. In this case for example:

$$\begin{aligned} AE = \{&(\text{LeaveRecall}, \text{LeaveRecallForm}(r)), \\ &(\text{TakeLeave}, \text{LeaveFormApproved}(f))\} \end{aligned}$$

Many agreements will be identified. However, not all agreements identified will have access control significance. The focus of the agreement abstraction in this dissertation is those that have access control significance.

Consider, in more detail, the *TakeLeave* agreement. This agreement has various elements.

$$\begin{aligned} a_2 &= \text{TakeLeave} \\ &= (x, y, T_2, C_2) \end{aligned}$$

Note that in practice there may very well exist a relation between the parameters in parameterized events and values in the agreement. For example, in this case x, y and some of the information used in specifically T_2 and C_2 may in fact be derived from the f parameter in *LeaveFromApproved(f)*. For discussion purposes here we use a dot notation to show such relationships. This notation is adopted for its explanatory value and does not formalize this possible relationship.

The first elements in the agreement is the offerer (x) and the second is the offeree (y). We can therefore define:

$$\begin{aligned} x &= f.\text{approver} \\ y &= f.\text{applicant} \end{aligned}$$

The terms of the agreement T_2 will be the third element of the agreement, and finally, the conditions C_2 which governs the agreement, the fourth element.

T_2 and C_2 can be expanded as follows:

$$\begin{aligned} T_2 &= \{grant(\text{finsys}, R_2, f.\text{substitute}), \\ &\quad\ revoke(\text{finsys}, R_2, f.\text{applicant})\} \\ C_2 &= \{hashappend(\text{DateReached}(f.\text{returndate}))\} \end{aligned}$$

Two terms are defined in T_2. The *grant* term indicates the system which will be effected *(finsys)*, the set of permissions (R_2) which are granted, and who should receive these grants, in this case *f.substitute*, the substitute indicated on the form. The set of permissions will depend on the semantics of the system. In this case it may be:

$$R_2 = \{\text{ApproveRequisition}(), \text{ViewSalaries}()\}$$

However, there may be conditions which could lead to the termination of the specific agreement. In this example, the condition to be met to terminate the agreement are the condition set C_2.

$$C_2 = \{hashappend(\text{DateReached}(f.\text{returndate}))\}$$

Therefore, the set of conditions C_2 contains only one element, *hashappend*. If a specific time is reached the condition is met, and the agreement ceases to exist.

To explain how the agreement hierarchy comes into play, consider the *RecallLeave* agreement. According to *AE* the *RecallLeave* agreement is created when *RecallLeave(r)* event takes place, possibly on completing a specific form. Details of how leave recalls will happen obviously will be different. Nevertheless, we can continue in an explanatory manner, referring to a case based on figure 5.1.

$$\begin{aligned} a_3 &= \text{RecallLeave} \\ &= (r.\text{requester}, r.\text{employeeonleave}, T_3, C_3) \end{aligned}$$

with the terms of the agreement

$$T_3 = \{\text{amend}(\text{TakeLeave}(a_2), T', C')\}$$

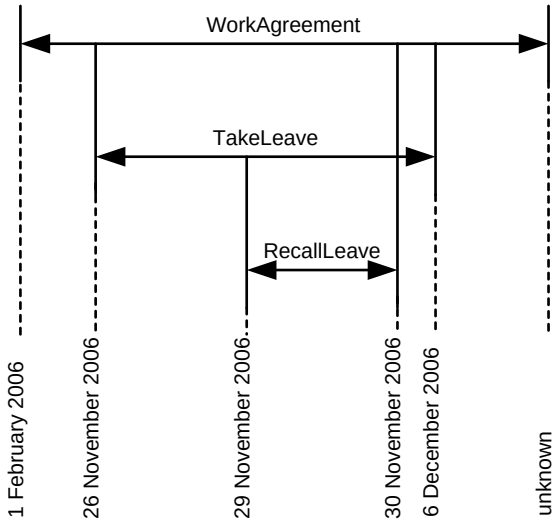

Figure 5.1: An Agreement Example

where

$$T' = T_2$$

that is no immediate change in terms for agreement a_2, but

$$C' \quad = \quad \{hashappend(\text{dateReached}(r.\text{newreturndate}))\}$$

To define C_3 we define a new event *amenddone*, therefore:

$$C_3 \quad = \quad \{hashappend(\text{amenddone})\}$$

Note, however that for this amend to be valid $TakeLeave \succeq RecallLeave$, which implies that:

$$(\text{TakeLeave}, \text{RecallLeave}) \quad \in \quad AH$$

Note that this latter implication is not always the case since there may be intermediary agreements in the role hierarchy. For example, we would also know that an employee cannot take leave unless he/she is employed. In a more complex example we have now:

$$\begin{aligned} AH \quad = \quad & \{(\text{WorkAgreement}, \text{Promotion}), \\ & (\text{WorkAgreement}, \text{TakeLeave}), \\ & (\text{TakeLeave}, \text{RecallLeave})\} \end{aligned}$$

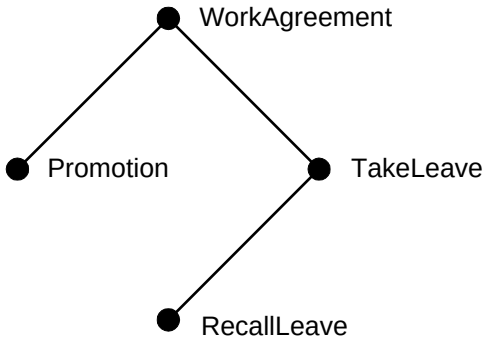

Figure 5.2: An example agreement hierarchy

which is depicted in 5.2. In this case:

$$\text{amend}(\text{WorkAgreement}, T_3, C_3)$$

would be a valid statement in *RecallLeave* since *WorkAgreement* $\succeq$ *RecallLeave*. This presents the essence of the agreement abstraction. Although many open issues exist in implementation, the author believes that the abstraction is useful in analyzing access control requirements and extensible and abstract enough to serve as a basis for further refinement.

5.4 Conclusion

This chapter introduced the agreements abstraction. The first section noted the various differences between existing access control thinking and the agreement abstraction.

One such difference is that the administration of access control should be a process in its own right, and that the focus should be on the processes within a business, not the structure of the business. This will support the empowerment of employees, lead to the creation of new knowledge, and ultimately benefit the business.

The second section explained the conceptual background to the term "agreement". Thereafter, the third section proceeded to formally define the agreement abstraction.

The use of events to identify agreements and incorporate it in an abstract way will allow for the agreement abstraction to be useful in automating

the granting and revoking of access control rights. It was shown that it was possible to look for triggering events, which provide businesses with the means to automate certain access control administrative processes.

Chapter 6 will now show how the agreement abstraction can be put to use. It does so by discussing various examples of how to identify and use agreements.

Chapter 6

Using the Agreement Abstraction

Chapter 5 defined the agreement abstraction. This chapter will explain how the agreement abstraction can be used to identify possible opportunities for the automation of access control administration.

The current chapter firstly describes a possible methodology for using the agreement abstraction. In the spirit of the four kinds of IT artifacts, identified in Chapter 1, it should be noted that this methodology is not seen as the main artifact designed in this dissertation. Instead, it serves as a mechanism to show the potential of the principal IT artifact produced: the agreement abstraction. No claim is therefore made as to the efficacy of the methodology; it is merely produced to show the potential usefulness of the underlying agreement abstraction.

After describing the possible methodology, it is shown in detail how the methodology can be used in analysing processes at a strongly customer-focused business. Thereafter the chapter briefly and anecdotally discusses three further examples. All examples are specifically chosen with a view to show that the abstraction

- is useful in environments where employees could be seen as knowledge workers
- may even be useful in more traditionally non-knowledge-based processes
- can support analysis across organizational boundaries.

- could lead to the identification of opportunities which would otherwise be difficult to identify.

Consider, firstly, the high-level overview of the methodology based on the agreement abstraction.

6.1 The high-level methodology

An abstraction, by its very nature, is only sensible if it can be used. In general, abstractions can assist us to understand complex issues better by highlighting the essence of the issues. As such, an abstraction must meet the needs for which it was developed.

In the case of the agreement abstraction, the abstraction was developed to identify potential opportunities for automating access control administration activities. Therefore, when considering the agreement abstraction this would imply identifying agreements and their corresponding events with the aim of automatically detecting the event in the systems environment and acting thereon as specified by the agreement.

This chapter shows that the agreement abstraction can indeed be used to achieve this goal. Applying it according to a high-level methodological approach will aid us in identifying events that we could possibly detect within the business' systems infrastructure. These events would be associated with agreements that embody the access control administration tasks to be performed at the occurrence of the identified event.

The high-level methodology proposed here consists of five distinct steps. These steps are depicted in Figure 6.1. Briefly consider these steps.

Step *one* is to identify the process that is the target of analysis. Typically this would be a problematic or high pay-off business process. This target process could have been identified due to security breaches or because of the importance of the business process to the success of the business.

The *second* step identifies all the related business processes. A process is deemed related if the process being analyzed depends on this process to have taken place, or it is a process which may take place when the process that is being analyzed have been completed. The dependent processes can become a target for analysis at a later stage.

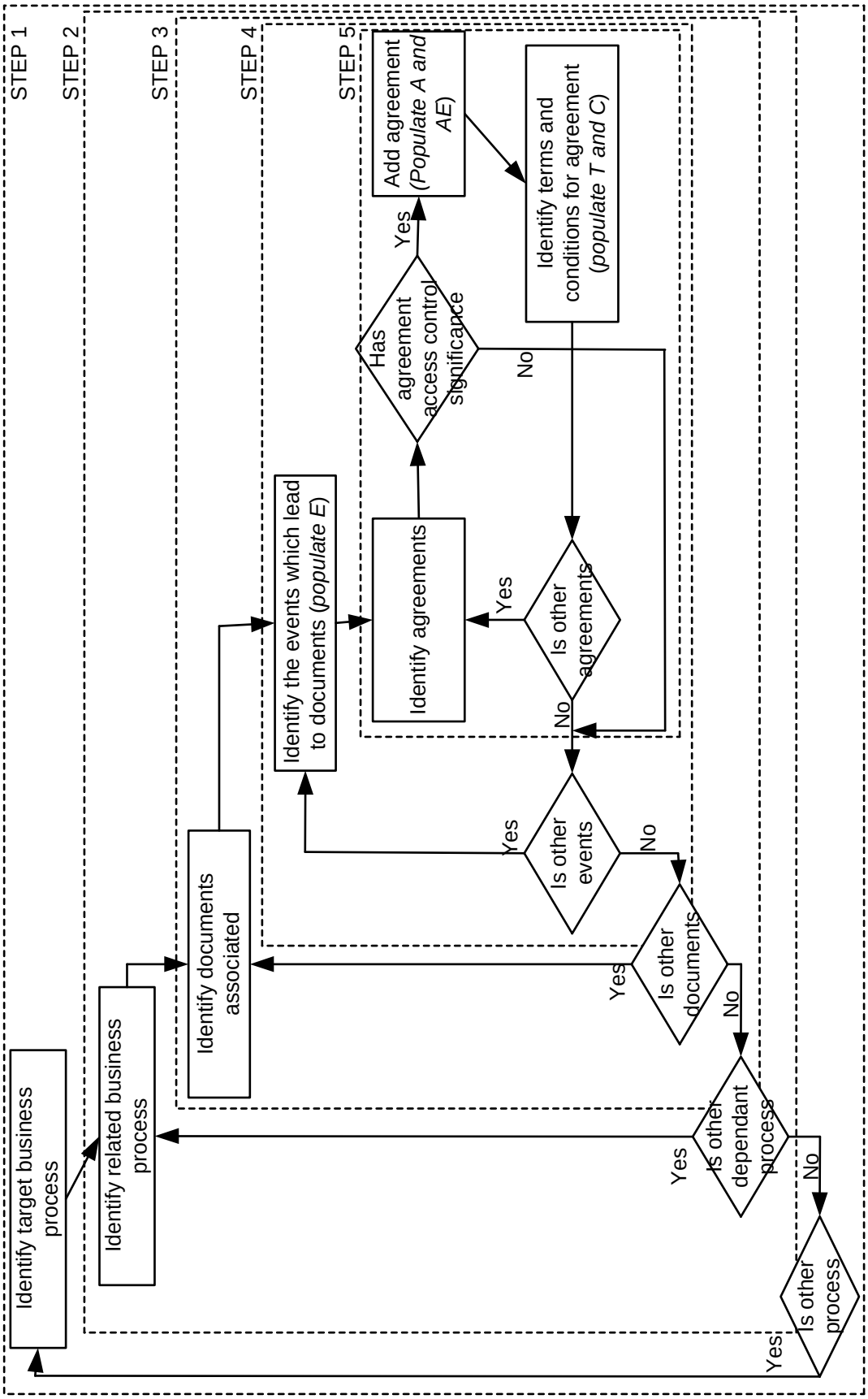

Figure 6.1: High-level methodology

Step *three* identifies all "documents" which form part of the analyzed process or are used as a communication means between the analyzed process and related processes. The word "document" will be used here in the widest sense of the word, indicating a record of communication independent from its technological implementation.

The identification of documents enables step *four*, the identification of significant events. In particular we see the events that lead to the creation and completion of the documents as possibly significant.

As documents could imply that there was a meeting of minds on some aspect agreements, may have been formed. Therefore, step *five* finally identifies agreements. Many of the agreements will have no access control significance; however, for those that indeed have access control implications we identify the terms and conditions.

The remainder of this chapter focuses on explaining each of these steps in more detail. The examples used are illustrative and have been generalized from meetings held with various businesses, but do not represent specific instances of the processes. We also recognize that the steps may in practice not be as linear as presented here. There may be much more back and forth between steps as it essentially is a discovery process.

6.2 A case study in customer focus

Chapter 2 acknowledged that the world in which we operate changed significantly over the last number of decades. The success of many businesses is no longer measured in the "manufacturing" paradigm associated with Taylorism. Instead, it is recognized that other measures, such a customer satisfaction, could have a significant impact on the sustainability of the business. Chapter 3 furthermore argued that current access administration is problematic, specifically when dealing with knowledge-based processes.

For this case study we therefore chose processes which are in line with the ideas behind knowledge workers. In their book "Re-engineering the Corporation" Hammer and Champy (2003, p. 66) argue that customers want a single point of contact. Many organizations therefore have implemented help desk systems.

This case study therefore will look into these type of businesses. We

commence by identifying the target business process, in this case the "handle a help desk call" process.

6.2.1 Identify the target business process

When dealing with a customer focused organization a possible target process could be the "handle new help desk call" process. Consider why this may be an appropriate target process, by assessing the salient properties of good candidate processes. A good candidate process

- *is important to the business.* In a true customer-focused business, the customer is king. Since the help desk can then be seen as the customer's point of contact, the importance of processes around the help desk is paramount.

- *is often identified as problematic in some way or the other.* In the case of help desks customers are often complaining that it does not provide help at all (Cena & Torre, 2006). Often these complaints may point to insufficiently empowered help desk consultants, which may be due to the the variety of information sources.

- *possibly requires access to a wide variety of information sources.* As customer's concerns are varied, it may be difficult to predict exactly what information the help desk consultant may be requiring (Richardson & Howcroft, 2006).

- *has potential privacy implications.* Help desk consultants may be privy to sensitive information. We therefore want to ensure that they only have access to information that is absolutely necessary. We also want to be able to track who accessed the information.

This process, can be argued, is knowledge-based. Why? The help desk consultant may need to access a wide variety of sources to deal with the customer. In the process the help desk consultant may build the knowledge base of the organization by adding knowledge about specific situations, specific problem or solutions to customer concerns. At worst, the help desk consultant collects information from the customer that could assist the organization to build a better view of the customer.

The "handle new help desk call" process therefore makes an interesting example to consider in terms of the agreement abstraction. Hence, consider the process as depicted in Figure 6.2.

The process of handling a help desk call starts when the help desk consultant's telephone rings. The help desk consultant picks up the telephone, and asks the user how he/she may be of assistance. The call is logged. Based on the answer the caller gives, the help desk consultant decides whether it is a problem that can be resolved by the help desk, or not. If the problem is of such a nature that it cannot be resolved by the help desk, the call is logged by taking down the details and giving the caller a reference number. This may be the case if, for example, it is obvious that a technician must be dispatched to a customer's site.

If the caller's problem is one which can be solved by the help desk, the help desk consultant attends to the problem. Of course, here we now assume that the help desk consultant should be empowered in order to assist the caller with whatever problem the customer might have. Therefore, it is imperative that the help desk consultant automatically receives the correct set of access control permissions in order to assist the customer.

It should not be necessary for the customer to contact any other employee in the business to assist him/her with the problem.

While investigating the problem, the help desk consultant must identify possible sources of relevant information. This may, for example, involve accessing the caller's customer records and delving into detail of certain transactions. By calling the help desk the customer implicitly allows access to his/her transaction records[1]. As the help desk consultant needs to access information he/she can "ask" permission by giving the help desk call as reference.

Once the investigation leads to a solution, this solution to the caller's problem must be communicated to the caller. This may involve further actions that the help desk consultant must perform on behalf of the caller. It may, for example, be required that the customer register for a different service or change the level of service he/she is subscribed to. The help desk consultant will get an explicit go-ahead from the caller.

[1]Where the information is extremely sensitive this permission may indeed also be explicit

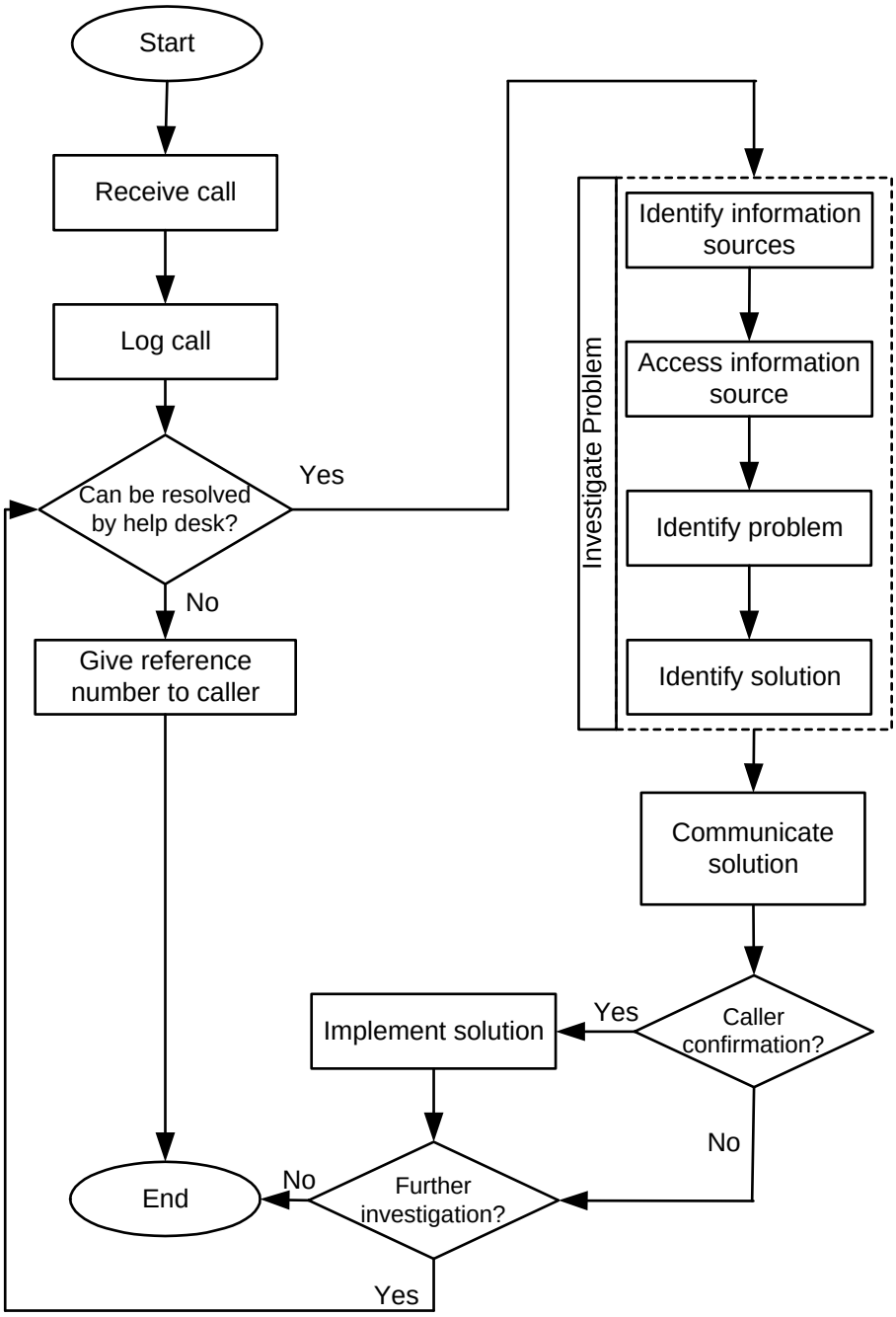

Figure 6.2: "Handle new help desk call" process

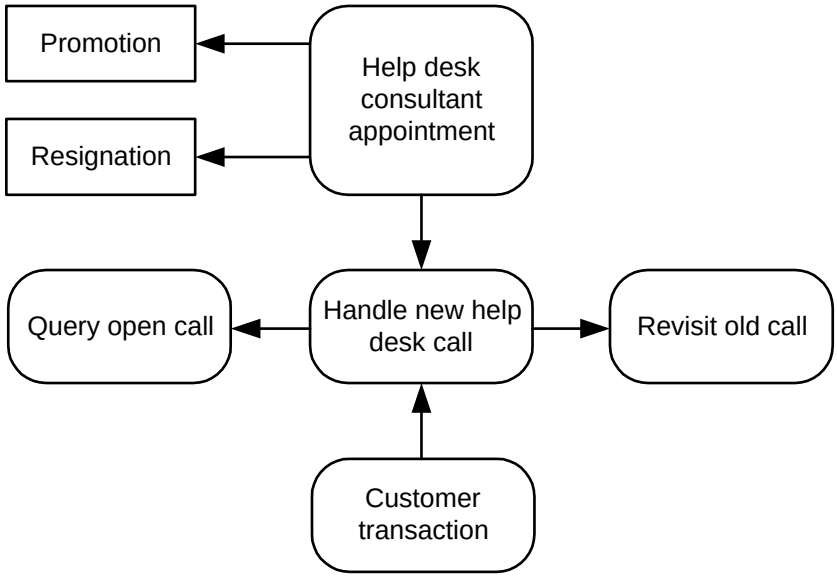

Figure 6.3: Some processes related to "Handle new help desk call" process

Refer to Figure 6.2 for more details on this process. Having identified and understood the process we can commence to the next step.

6.2.2 Identify dependant business processes

In order to understand the business process better, the context of the business process must be understood. One way to do this is by identifying related business processes. Related business processes include

- processes which happened in the past, making the analysed process possible and
- processes that can happen now that the analyzed process has taken place.

Doing this exercise for the process under the spotlight will lead to a dependency diagram as given in Figure 6.3. Consider what this diagram depicts.

First note that "handle a new help desk call" implies that the help desk consultant has a work arrangement with the employee. This is established in the *Appoint Help Desk Consultant* process. In this example, the help desk deals with existing customers and we can therefore state that some *Customer transaction* process has taken place in the past.

Similarly we can argue that since a new call has now been handled, it opens opportunities for *Query open call* and *Revisit old call* processes to exist. A help desk call that cannot be resolved by the help desk consultant can be referred to a technician; this can be done in a *Assign Technician* process.

6.2.3 Identify applicable documents

The identification of documents happens in the widest sense of the word. While it includes what might be described as traditional business documents (contracts, letters, memos, purchase orders, and so forth), it could also include a variety of "other" documents.

These "other" documents could include voice recordings, system event logs and other electronic representations that could serve as evidence of an agreement. In fact, the author believes that having an open mind as far as this step is concerned is crucial for the successful application of the agreement abstraction. Specifically thinking not only about the communication between people, but also between people and systems and between business processes could aid in identifying applicable documents.

As an example, consider the "handle new help desk call" process. Possible documents include:

- the recorded telephone conversation or even parts thereof,
- system events logs that show activity of the help desk consultant,
- electronic forms being completed by the help desk consultant, and
- the electronic rendition of the proposed solution.

Considering the possible documents provides excellent cues for identifying events.

6.2.4 Identify events

By considering the process description (depicted in Figure 6.2) and the documents together, the possible events can be identified. Table 6.1 describes the events that can be identified. Note that for brevity in the mathematical notation the table labels each event with a unique code.

Table 6.1: Identified events in "handle new help desk call" process

Code	Event description
rec_call	Receive the call
prb_cat	Problem categorized (solvable by help desk or not)
cust_go_invest	Customer go-ahead to start investigation
hdc_info_req	Help desk consultant requests additional info
sol_offer	Solution offered to caller
cust_acc_sol	Customer accepts solution
ref_give	Reference number given
call_cancel	Call cancelled by caller or caller indicates satisfaction

Note that some of these events will be parameterized in terms of the relevant call record cr, while others will use the help desk consultant's request record req. We, therefore, have:

$$\begin{aligned} E \;=\; & \{\text{rec\_call}(cr), \text{prb\_cat}(cr), \text{cust\_go\_invest}(cr), \text{hdc\_info\_req}(req), \\ & \text{sol\_offer}(cr), \text{cust\_acc\_sol}(cr), \text{ref\_give}(cr), \text{call\_cancel}(cr)\} \end{aligned}$$

Once the events are identified, we can further explore the events to determine whether they lead to an access control relevant agreement.

6.2.5 Identifying agreements

To identify possible agreements one can start by considering the access control relevance (if any) of events. Table 6.2 summarizes the potential access control relevance of the events identified in the previous section.

Table 6.2: Access control relevance of events

Event Code	Access Control relevance
rec_call	basic customer information required and new call record
prb_cat	update call record
cust_go_invest	access granted to read customer records
hdc_info_req	requested information is accessed
sol_offer	none
cust_acc_sol	may require permission to change records
ref_give	no further access to records required
call_cancel	no further access to records required

By considering the access control implications of the various events, we can, in this case, identify several possible agreements. These include:

- there is an agreement between the help desk consultant and the caller to investigate the problem
- there could be an agreement between help desk consultant and the caller to go ahead with solution
- there is an agreement between the business and the help desk consultant to answer help desk calls.

Using descriptive identifiers, we therefore have the set:

$$A = \{\text{InvestigateCustProb}, \text{ImplementSolution}, \text{AppointHDC}\}$$

We can formally identify the relation between agreements and the events that triggered them by populating the relation AE as

$$\begin{aligned} AE = \{&(\text{InvestigateCustProb}, \text{cust\_go\_invest}(cr)),\\ &(\text{ImplementSolution}, \text{cust\_acc\_sol}(cr)),\\ &(\text{AppointHDC}, \text{rec\_call}(cr))\} \end{aligned}$$

Having identified the agreements that appear to have access control significance, we can formally flesh out their definitions by considering them as specified in Chapter 5

$$a_i = (x, y, T_i, C_i)$$

Therefore the agreement $a_1 = \text{InvestigateCustProb}$ can be represented as:

$$\begin{aligned} a_1 &= (cr.\text{HelpDesckConsultant}, cr.\text{Customer}, T_1, C_1)\\ T_1 &= \{grant(\text{crm\_system}, R_1, cr.\text{HelpDeskConsultant})\}\\ R_1 &= \{(\text{view\_customer\_record}, \text{remote\_desktop\_view\_access}, \text{access\_to\_wan})\}\\ C_1 &= (\text{hashappened}(\text{call\_cancel or cust\_acc\_sol}) \end{aligned}$$

Similarly, the agreement $a_2 = \text{ImplementSolution}$ can be represented as:

$$\begin{aligned} a_2 &= (cr.\text{HelpDesckConsultant}, cr.\text{Customer}, T_2, C_2)\\ T_2 &= \{grant(\text{crm\_system}, R_2, cr.\text{HelpDeskConsultant})\}\\ R_2 &= \{(\text{update\_customer\_record}, \text{remote\_desktop\_write\_access}, \text{access\_to\_wan})\}\\ C_2 &= (\text{hashappened}(\text{call\_cancel or cust\_acc\_sol}) \end{aligned}$$

When fleshing out the third possible agreement we realize that the basic rights are given to the help desk consultant because of the "Appoint Help Desk Consultant" process as identified in Section 6.2.2. This, therefore, urges us to consider that process in more depth. However, for the purposes of this explanation we will not proceed with that.

Consider therefore how the agreement abstraction has helped us to identify opportunities for access control automation.

Firstly, we have identified events that must be monitored within the relevant systems. This provides us with the opportunity to develop trigger mechanisms to programmatically detect these events.

Secondly, these events were associated with agreements. These agreements clearly spelled out the access control administration activities that must take place. Furthermore it identified the conditions of fulfillment, that is when the access privileges should be revoked again.

To strengthen the case for agreements, briefly consider some further examples.

6.3 Further examples

The agreement abstraction proposed in this dissertation was primarily aimed at analyzing access control requirements in environments where knowledge working is valued. The previous example was then indeed based on a knowledge intensive process. This section will continue the quest to show that the agreement abstraction is indeed useful. It does so by briefly discussing its applicability in three different scenarios.

First, consider a more traditional process, the requisition process.

6.3.1 Requisitions: a more traditional case

Dealing with requisitions can certainly be described as a "traditional" process in most businesses. Although access control issues within this kind of environment are generally well understood, this section argues that the agreement abstraction could also be used in more "traditional" environments. Consider each of the phases in the high-level methodology and how it is applicable to this case.

The *target business process* here is the requisition process. As it effectively enables the spending of money, it is certainly a process worth considering.

Several *related business processes* can be identified. These include, but are not necessarily limited to, domain specific business processes that may be responsible for depletion of stock, financial processes associated with the payment of delivered goods or services and store-related processes associated with the fulfillment of orders. From a Human Resources perspective appointment, promotion and resignation processes could impact on user's access rights.

The *applicable documents* in this scenario are traditional "requisitions". "Quotations" may also be encountered and orders could be generated or filled out.

Several *events* can be identified. Examples include: requisition completed, requisition approved, requisition rejected, order placed and stock threshold reached.

Potential *agreements* can be identified. For example approvers of requisitions are given their permissions based on the agreement activated when they took on a position of responsibility; typically specified through an employment contract representing an "appointment" agreement. The question is then how promotions can be seen. This is also a type of agreement that amends the original employment agreement. Similarly resignations can be seen as an agreement that amends the employment contract. While promotions would amend the terms of the "appointment" agreement, resignations effectively amend the conditions.

Requisitions can be issued manually, as a specific need arises or automatically based on a certain stock level threshold being reached. In the latter case the "stock level threshold reached" event would cause a "top-up stock level" agreement to be established. This demonstrates an interesting extension to the agreement abstraction although, as specified here, it has no access control relevance; in other words, nobody specifically needs different access control rights. What is interesting is that on the surface it would appear as if the agreement abstraction can also be employed to identify automation opportunities other than those for access control administration. This idea, however, falls outside the scope of this dissertation.

Also note how the association of the appropriate event is extremely im-

portant to achieving an appropriate level of granularity with respect to access control admistration events. In the previous discussion essentially the same agreement could come into effect at different events. If Person X may order non-stock items at any point in time the agreement is established by an event associated with the relevant appointment process. However, if we want to implement the business rule that stock items can only be ordered by Person X when a certain threshold is passed, the agreement would only be made once the stock-level reaches that threshold level. Note that this is somewhat different from the automation opportunities mentioned above since a user is still involved.

Although these examples are only a selection of ideas in the traditional requisition process, it clearly shows that the agreement abstraction can be useful in non knowledge-based processes. However, as expected, some shortcomings also exist in this environment. For example, in these financial type processes, concepts such as separation of duties typically play an extremely important role. The agreement abstraction, as formalized in this dissertation, does not address that type of business rules. While this may restrict application of the agreement abstraction somewhat in specific environments, it is believed that the abstraction is fundamental enough to allow for such extensions when required.

Next, we consider how the agreement abstraction could assist us in web-based processes. For this purpose we have a look at an online book store.

6.3.2 Online book store

With the proliferation of access control it is useful to evaluate new design and analysis techniques in this environment. This subsection, therefore, explores the usefulness of the agreement abstraction in the context of an e-commerce example: an online book store.

XYZ.com sells books on the Internet. Customer X visits XYZ.com online to view books that he/she is interested in. Customer X can also read book reviews that previous customers have written on the particular book. If Customer X decides to buy the book he/she will proceed through the payment process and the book will be shipped to the Customer X.

However, XYZ.com accepts only reviews from customers that bought the

book[2]. In essence an agreement exists between Customer X and XYZ.com, we will refer to this as the "reviewbook" agreement. The "reviewbook" agreement comes into existence immediately after the "customer received book" event.

The terms of the"reviewbook" agreement stipulates that the customer now gets access to write to the review area of the book's record. Since the customer stays an owner of the book, this agreement will never expire and no condition needs to be specified.

Conceptually this scenario is simple. It may however, from a possible implementation perspective require some notable integration between systems to identify "customer received book" events. This said, for the purposes here the agreement abstraction is abstract enough not to be influenced by technology related issues. It can thus be argued that the agreement abstraction could be used in cases which stretch across organizational boundaries.

As a third example we will consider a problem of smaller scope and shows how it can help with access control administration and automation in what the author believes is non-obvious cases. Therefore, consider the example of managing personal time.

6.3.3 Personal time management

To illuminate the message of this section consider a typical office scene. Busy managers running from meeting to meeting, some in their offices, others in other places. Now step into the shoes of one of the employees, call him Z. Z schedules all his meetings in his electronic calendar. However, he has a problem in that many meetings take place in his office and he has no personal assistant to screen his calls. Let us consider how analyzing his problem by means of the agreement abstraction can assist.

We have just identified *the target process*, being scheduling meetings. Being such a low-level process no clear *related processes* are immediately evident.

Since Z schedules meetings electronically the meeting record is an important *applicable document*. Several *events* can be identified. Example events

[2]This is probably bad business sense in today's day and age, but serves to illustrate a point here.

include meeting scheduled, meeting accepted, meeting started and meeting ended.

We can now identify that an agreement between Z and the meeting participants is formed when the "meeting started" event has taken place. Next we question the access control relevance. Here it is necessary to think a bit outside the box. One of the issues that Z has is that his telephone often rings while in meetings. If we see people's ability to make a call to his number as an access right, the term of the agreement is to revoke calling privileges to this number from everybody or alternatively reroute calls (that is grant access to a different device based on the same identifier). The condition of this agreement is again time-based: once the end time of the meeting is reached rights can be returned to normal.

Of course it may also be possible that other agreements can be in place in this process. For example, accepting the invitation to a meeting may result in an agreement coming into play where all participants would receive access to a shared workspace.

This example showed that the agreement abstraction could be helpful in discovering interesting non-obvious opportunities for access control administration.

Having discussed four different examples of using the agreement abstraction we can now conclude.

6.4 Conclusion

This chapter set out to show how the agreement abstraction can be used to solicit access control administration requirements. Several examples were considered.

A knowledge-based process, the "handle a new help desk call" was considered and detailed access control requirements derived. This demonstrated that the agreement abstraction is indeed useful for analyzing the access control requirements of knowledge workers.

In addition three other scenarios were briefly explored. The first, a requisition process, showed that the agreement abstraction could be useful even in more "traditional" processes. The second considered an e-commerce scenario and argued that the abstraction is not influenced by organizational

boundaries and the Internet. The third example, personal time management, demonstrated that the abstraction could stimulate innovative thinking and can allow for non-obvious processes to be considered.

The author believes that this chapter has shown that the abstraction can be usefully employed by analysts. This supports the claim that the agreement abstraction contributes to knowledge in the domain of discourse.

Chapter 7 will conclude this dissertation by summarizing the results of this research and present ideas for future work.

Chapter 7

Conclusion

In Chapter 1 the realization was made that controlling access to information did not necessarily mean keeping information away from users. However, it was noted that the current focus of access control administration is to restrain access rather than to enable access to information. To restrain users from gaining unauthorized access to information requires a lot of work, usually done manually by a human administrator. In addition the size of a business and its dynamic nature causes the administrative burden to increase.

For these reasons, the automation of access control administration was suggested. It was argued that access control administrative tasks are repetitive and thus suitable for automation. This need for automation is already recognized by provisioning systems, but it was pointed out that administration within provisioning systems happens in an ad hoc manner.

Therefore, the research question "How can access control requirements within the context of a provisioning system be determined?" was identified in Chapter 1 as a pressing need in this dissertation.

The three chapters following Chapter 1, set out to investigate the theoretical foundations of the research question. Chapter 2 identified that the business world has changed. Among other things, the way businesses are measured have changed. This supports the move towards knowledge working, which, in turn, highlights the need for more access. If this is the case, administration of access control will need to occur on a more frequent basis.

From a corporate governance perspective, the business is responsible to protect its information. Although knowledge workers require more access, proper governance requires control. Therefore, a tension exists between the

need to share and the need to protect information.

Chapter 3 discussed the status quo of access control. It was noted that due to the widespread implementation of access control, it is necessary to make use of existing access control paradigms. We, therefore, aimed to develop an abstraction which is independent of existing access control paradigms.

Furthermore, the idea to automate the administration of access control is supported due to the human factor, as well as the size and complexity of businesses. In the study of access control, the concept of provisioning was identified.

Chapter 4 introduced the concept of provisioning. It was shown that it shares our philosophical decision of dynamically assigning access to users. Provisioning systems were upheld as valuable building blocks when providing access to knowledge workers in an ad hoc, dynamic manner.

Chapter 5 defined an agreement abstraction in a bid to contribute to answering the research question. The agreement abstraction is strongly based on the fact that access to information needs to occur in a dynamic way. Therefore, it is evident that we would rather empower the knowledge worker by means of providing access to information and keep an audit trail, than prevent access to information.

Agreements are formed between two parties and is specified through various terms and conditions. The terms define what access is going to be granted, whereas the conditions will point to when the agreement will cease to exist.

Chapter 6 proposed the steps to be followed in order to make use of the agreement abstraction. These steps were explained in one detailed case study. However, various other examples of possible uses for the agreement abstraction were also discussed. This chapter managed to show that the agreement abstraction can indeed be useful in analysing the access control administration requirements.

Having contributed the agreement abstraction to the knowledge base concerning access control administration, the next section ponders on future extensions to this work.

7.1 Future Work

In order to comprehensively answer the research question from a design science perspective, all levels as depicted in figure 1.2 on page 6 can be considered. This dissertation proposed an abstraction at the model layer. A method was shown to demonstrate the usefulness of the proposed agreement abstraction. To establish a future research agenda consider each of the levels in turn.

The agreement abstraction was based on existing constructs. The constructs originated from other fields, for example, access control, accounting and law.

Other fields may contain constructs to be explored which can also make contributions towards the agreement abstraction, or even replace the proposed abstraction. Further constructs searches of an interdisciplinary nature is thus a possible avenue for future researchers.

The agreement abstraction, as proposed in this dissertation, made a contribution on the model category. For future work the agreement abstraction model can possible be refined. In the dissertation mention was made to investigating the relationship between terms and conditions in the agreement abstraction and the obligation concept recently introduced to the access control discourse. Furthermore, the formalization of the agreement abstraction can be extended. Of particular interest here would be formalizing the relation between document semantics and terms and conditions.

Of course this assumes that the agreement abstraction is indeed the best alternative. While it proved to be feasible and useful in this dissertation, alternatives were not thoroughly explored. Alternatives to the agreement abstraction can be proposed at the model level of IT artifacts.

Similarly, different methods of using the agreement abstraction can be explored. The method presented here could be refined or alternatives proposed and compared. Finally future contributions could be at the instantiation level.

These instantiations could include the development of prototypes in order to verify whether it is in fact possible to build a generic engine to automate access control administration based on the agreement abstraction. In building such prototypes it will be possible to furthermore evolve the processes

of identifying agreements as well as automating the agreements. Practical matters, such as integration into existing systems will also be challenging. Various questions of a technical nature would have to be answered.

Still on the instantiation front, the design and building of a design tool to support the agreement abstraction would be useful. Such a tool will facilitate the processes of analyzing the business, identifying agreements, and maybe even assist with the implementation of access controls.

Ideally the agreement abstraction can be tested through a real life case study. This will provide the ultimate proof that the agreement abstraction is useful in the administration of access control.

7.2 Final Word

The fact that access control administration becomes a problem in a heterogeneous and dynamic environment has been noted in this dissertation. This dissertation contributed to this problem's solutions by suggesting that the agreement abstraction could aid in analyzing access control administration requirements. The main contribution of this dissertation, therefore, is the agreement abstraction.

However as is clear from the previous section, this only begins to answer the research question. Much further research is necessary. The author hopes that this dissertation will stimulate further research in this dynamic environment and that it will bring a more positive outlook to the administration of access control.

Appendix A

Access control in commercial systems

This appendix discusses various state of the art applications that are commercially available. This investigation was done in order to try and understand access control in the business world and thus provide some evidence regarding practical investigations.

In this investigation it was identified that the administration of access control is typically done on three levels, depending on the type of system. These systems are : databases, portal software and Middleware.

Section A.1 discusses access control administration within databases. Hereafter, section A.2 investigates the administration of access control in portal software. Finally, section A.3 discusses middleware.

A.1 Databases

Databases have been used for the last few years as data repositories for various applications. There are a large variety of databases available in the market. These range from databases used for smaller applications to databases used in the larger corporate. These databases are installed on platforms. One such database used in the corporate world is Oracle.

A.1.1 Oracle

Oracle Corporation is known worldwide for its Oracle database. In a large business the administration of an Oracle database can be a difficult task. Oracle developed software which helps with this administration. Oracle User Management is an application introduced with the 11.5.10 release of Oracle Applications (ORACLE, 2004).

Oracle User Management is an administrative system which enables businesses to define administrative functions, as well as manage users. In order to manage these users, Oracle User Management enables the creation of decentralized administrators. It is thus possible to create local administrators, give them the necessary privileges, and they will be able to manage certain users of the system.

A Role-based Access Control (RBAC) model has been introduced by Oracle User Management, which strongly resembles the RBAC model proposed by the National Institute of Standards & Technology. The difference between these two RBAC models is that the RBAC model introduced by Oracle User Management has added some extra methods. These methods help with the organization of data security policies and existing function security.

Build upon the RBAC model of Oracle User Management, three main features have been included into Oracle User Management. They are: Delegate Administration, Registration Processes, and Self-Service Request and Approvals.

Another database, which is widely used in businesses, is Microsoft's SQL server.

A.1.2 Microsoft SQL Server

Microsoft's SQL server 2000 is also one of the well known databases used by businesses. SQL server 2000 manages security to the database itself in two ways (Microsoft, 2003). The first of which is by means of SQL server authentication, and the second is windows authentication. In order to manage this security, a utility, supplied with SQL server 2000, needs to be used. This utility helps administrators manage the users of the database. This is done by assigning users to specific groups. Each group is then assigned specific read, write, or read and write access to specific databases, tables or views.

These administrative tasks are handled by the business security administrator or an employee(s) responsible for enforcing access control. There are various types of administrators which can be configured on the database itself.

However, although Oracle and Microsoft's SQL server are widely used, the situation exists, that other databases are also being used. Therefore, the administration of access control can become even more problematic. Portal software have been developed to not only address the administration of access to database problems, but also the administration of access to other resources.

A.2 Portal Software

mySAP Enterprise Portal (mySAP EP) is a portal system that uses "state-of-the-art security technologies" to control the access to all of the business resources (SAP AG, 2002). This access control is made possible by means of the J2EE connector architecture (JCA) (SAP AG, 2003).

Although only a predefined list of resources is integrated by default in mySAP EP, a Portal Development Kit (PDK) is supplied. This Kit includes all the necessary tools to develop and connect to SAP EP, and therefore, mySAP EP is able to support a high degree of heterogeneity. However, although many systems are able to connect to mySAP EP, this access needs to be configured.

A simple interface is used to facilitate the process of assigning application and information level access. This process highlights the assignment of roles to specific employees, which closely resembles the process found in role-based access control (SAP AG, 2004). These assignments will enable those employees to gain access to personalized gateways. These gateways could include certain information pages, working sets, services, and even interfaces to applications that particular employees should be able to use.

There are various administration tools offered by SAP Enterprise Portal to develop roles (SAP AG, 2004, p. 18). One such system is called the SAP User Management Engine (UME).

There are various administrators profiles which can be assigned to users. These include super administrator, content administrator, system adminis-

trator and user administration (SAP AG, 2004, p. 20).

The problem is that there are many such systems, which integrate security, as well as various databases, as discussed earlier, plus many other resources which exist within the business. Middleware is control software which integrates various resources within the business and provides a single interface to all of the information.

A.3 Middleware

Middleware is software products which integrates various resources within the business. One such middleware product is IBM's Websphere.

A.3.1 IBM's Websphere

IBM's Websphere is middleware which enables a business to integrate its systems (IBM, 2005). This type of integration enables a business to pursue many extra capabilities. These capabilities include: federation, replication, content integration, enterprise searches and event publishing (IBM, 2005a). The aforementioned can enable a business to gain a competitive advantage over its competitors.

There are a number of predefined resources which can be integrated. However, API's are provided to develop resource adapters to integrate with non predefined resources (Fontes, Nordstrom, & Sutter, 2004).

In order for WebSphere to integrate with other software systems, resource adapters as well as mediators are used. These resource adapters and mediators are used to provide an interface between WebSphere and the software system (Bhaskaran & Schmidt, 2004). Therefore, a high level of resources heterogeneity is supported (Bhaskaran & Schmidt, 2004).

When WebSphere Information Integrator integrates various systems, it appears that information sources are federated (IBM, 2005a). Employees will be able to access information as if that information was stored on one database (IBM, 2005b). However, information security has become one of the main concerns of businesses. Integrating all of the different systems within a business raises some security concerns. The reason for this being that each and every system within a business incorporates its own security.

Two methods exist to manage access control. The first is using a native WebSphere Application Server authentication (IBM, 2002). A request for WebSphere Application server security is made by means of a servlet. Such a request occurs in order to validate a user's authentication data. This process occurs when the login form posts user's authentication data.

The second option is that a Trust Associate Interceptor (TAI) interface is provided by WebSphere Application server which establishes co-operation with trusted authentication proxies.

If the native WebSphere Application is used, the access control will be done on an application supplied by WebSphere Application server. If authentication is done by means of an external access control program, it depends on that access control management system where access control is managed.

IBM Websphere can be configured to use external access control systems (IBM, 2002). These systems include: Tivoli Access Manager, Netegrity SiteMinder and Entrust GetAccess.

Following section discusses IBM Tivoli's Access Manager.

A.3.2 IBM Tivoli Access Manager

IBM Tivoli Access Manager is a centralized control system which implements authentication and authorization into the corporate Web, client/server and existing applications (IBM, 2003). Access Manager provides applications with authorization services (Karjoth, 2003). In order to make use of these services, an application needs to be part of the Access manager family.

If the application is not part of the Access manager family, they are provided with authorization API's, which enables them to gain access to the Access Manager's authorization services.

Therefore, it is also possible to secure Network-based applications as well as e-business infrastructure, when using Tivoli Access Manager. Due to the fact the IBM Tivoli Access Manager is only a product which handles access to information, it seems to completely replace any access control mechanism which might exist.

If a user tries to gain access to a resource, the reference monitor intercepts this request. This request, together with the ID of the user requesting the information, the name of the resource, and the set of permissions required to execute this request, are passed by the reference monitor to the Authorization

server. The ability of the user to gain access to the requested resource is then determined by the Authorization server. External services provide, as well as maintain, security attributes of principles. These external services are relied upon by the Access Manager Authorization services.

Only selected employees, the security administrators, can change authorization states. Because of the fact that employees with similar security properties are collected into groups, and permissions are then granted to employees, as well as groups, it closely resembles that of the role-base access control paradigm.

If Tivoli Access Manager is used in conjunction with an other standard internet-based application, secure and well-managed intranets can be built.

As mentioned in section A.3.1, it is not only Tivoli Access Manager which provides a access control solution. Entrust GetAccess is another external access control system, which can be configured by IBM Websphere.

A.3.3 Entrust GetAccess

Entrust GetAccess is a system that provides Web portal applications with an access point for user authentications and authorization (Entrust, 2003). With Entrust GetAccess a user's experience of a Web Portal is enhanced with the following features: security, flexibility and performance. Standards, like the Security Assertion Markup Language (SAML), are responsible for authentication interoperability, whereas Authorization is handled by a powerful Role-based Access Control (RBAC) model.

In order to provide this security across various platforms and systems, Entrust GetAccess must integrate with other applications. Two mechanisms have been included to handle integration (Entrust, 2003). The first is via API's which manage the way data travel to and from Entrust GetAccess. The second mechanism is that a system's behavior can be programmatically customized with given events.

Therefore, Entrust GetAccess is a complete access control administration solution. The main idea is to create a scenario where the administration originates, and is maintained, in centralized space. Although access control is managed on the Entrust GetAccess server by means of role-based access control, many other authentication schemes are also supported.

Entrust GetAccess provides the administrator with a comprehensive browser-

based administration tool. Initially administrators can define the resources which must be protected. The roles that are needed to access these resources can then be defined, whereafter specific users can be created and assigned to the specific roles. Specific information regarding the type of authentication that is used, and the time of the day specific resources may be accessed, can also be defined.

Administration is done by specific administrators of the system. These administrators can delegate administration tasks to other administrators on specific levels of administrative privileges. For example: Susan was delegated administrative duties for access control to the Sales department. When Susan wants to change access permissions, she will only be able to see information regarding the sales department, when accessing the administration tool of Entrust GetAccess.

References

About SAICA. (2004). (`http://www.saica.co.za` (last cited: 10 August 2004))

Abramson, N. M. (1985). Development of the ALOHANET. *IEEE Transactions on Information Theory*, *31*(2), 119-123.

Aggarwal, M., Atakan, O., & Boyce, C. (2005). *Operating System and Application Provisioning Using IBM Tivoli Provisioning Manager 3.1.* `http://www-306.ibm.com/software/tivoli/products/prov-mgr`: IBM. (Last cited: 3 Feb 2007)

Atluri, V., & Huang, W. kuang. (1996). An Authorization Model for Workflows. In *ESORICS '96: Proceedings of the 4th European Symposium on Research in Computer Security* (pp. 44–64). London, UK: Springer-Verlag.

Bertino, E., de Capitani di Vimercati, S., Ferrari, E., & Samarati, P. (1998). Exception-based Information Flow Control in Object-Oriented Systems. *ACM Transactions on Information and System Security*, *1*(1), 26–65.

Bertino, E., Ferrari, E., & Atluri, V. (1999). Specification and Enforcement of Authorization Constraints in Workflow Management Systems. *ACM Transactions on Information and System Security*, *23*(1), 65–104.

Bertino, E., Samarati, P., & Jajodia, S. (1993). High assurance discretionary access control for object bases. In *CCS '93: Proceedings of the 1st ACM conference on Computer and Communications Security* (pp. 140–150). Fairfax, Virginia, United States: ACM Press.

Bhaskaran, K., & Schmidt, M. (2004). WebSphere Business Integration: An architectural overview. *IBM Systems Journal*, *43*(2), 238–254.

Boiko, B. (2002). *Why Content Management – A Content Management Domain White Paper* (Tech. Rep.). www.metatorial.com: Metatorial Services Inc. and HungryMinds Inc.

Botha, R. (2001). *CoSAWoE : A Model for Context-sensitive Access Control in Workflow Environments.* Unpublished doctoral dissertation, Rand Afrikaans University, Pretoria, South Africa.

Breaux, T. D., Vail, M. W., & Anton, A. I. (2006). Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations. In *14th IEEE International Requirements Engineering Conference (RE'06)* (pp. 49–58). Delft, The Netherlands: IEEE Computer Security.

Cantor, S., Kemp, J., Philpott, R., & Maler, E. (Eds.). (2005). *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0 (SPML) Version 1.0.* Available from: `http://docs.oasis-open.org/security/saml/v2.0`: OASIS Specification. (Last cited: 4 Jan 2007)

Cappelli, P., & Hamori, M. (2005). The New Road to the Top. *Harvard Business Review*, *83*(1), 25–32.

Cena, F., & Torre, I. (2006). Adapting the interaction in a call centre system. *Interacting with Computers*, *18*(3), 478–506.

Chen, J. J., & Adams, C. (2004). Short-range wireless technologies with mobile payments systems. In *ICEC '04: Proceedings of the 6th International Conference on Electronic Commerce* (pp. 649–656). New York, NY, USA: ACM Press.

Cisco Systems. (2003a). *Cisco Networking Academy Program CCNA 1 and 2 Companion Guide* (3rd ed.). Cisco Press.

Cisco Systems. (2003b). *Cisco Networking Academy Program CCNA 3 and 4 Companion Guide* (3rd ed.). Cisco Press.

Citrix Password Manager. (2006). (`http://www.citrix.com` (last cited: 10 October 2006))

Computer Staff. (1996). Timeline of Computing History. *Computer*, *29*(10), TL1–Tl34.

Davenport, T., & Beck, J. (2000). Getting the Attention You Need. *Harvard Business Review*, *78*(5), 118–126.

Drucker, P. (1973). *Management: Tasks, Responsibilities & Practices.* New York: Harper and Row.

Eliasson, G. (2005). The nature of economic change and management in a new knowledge based information economy. *Information Economics and Policy*, *17*(4), 428–456.

Entrust. (2003). *Entrust GetAccess: Secure identity and access management - Technical Overview* (Tech. Rep.). `www.entrust.com`: Entrust.

Fontes, S., Nordstrom, C., & Sutter, K. (2004). WebSphere connector architecture evolution. *IBM Systems Journal*, *43*(2), 316–326.

Foster, H., Uchitel, S., Magee, J., & Kramer, J. (2006). Model-Based Analysis of Obligations in Web Service Choreography. In *Found in: Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT-ICIW'06)* (pp. 149–156). Washington, DC, USA: IEEE Computer Society.

Fraser, A. G. (1996). Future WAN Telecommunications. *IEEE Micro*, *16*(1), 53–57.

Gama, P., & Ferreira, P. (2005). Obligation Policies: An Enforcement Platform. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'05)* (pp. 203–212). Stockholm: IEEE Computer Society.

Gerwig, K. (2001). Business: The 8th layer: disruptive behavior. *netWorker*, *5*(3), 9–12.

Godik, S. (Ed.). (2003). *eXtensible Access Control Markup Language (XACML) Version 1.0.* `http://www.oasis-open.org`: OASIS.

Hammer, M., & Champy, J. (2003). *Reengineering the Corporation (Originally published: 1993).* HarperBusiness Essentials.

Hernandez, J. (1997). *The SAP R/3 Handbook.* 11 West 19th Street, New York, NY 10011: McGraw-Hill.

Hevner, A. R., March, S. T., & Park, J. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75 – 105.

Higson, C., Zimmermann, J., & Itter, C. (1997). Accounting [Module]. In T. Dickson (Ed.), (pp. 15–56). IMD International, London Business School and Warton School of University of Pennsylvania.

Hodges, J., Philpott, R., & Maler, E. (Eds.). (2005). *Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0.* Available from: `http://docs.oasis-open.org/security/saml/v2.0`: OASIS Specification. (Last cited: 4 Jan 2007)

Hollingsworth, D. (1995). *The workflow reference model* (Tech. Rep. No. TC-00-1003). www.wfmc.org: Workflow Management Coalition.

Hwang, Y., & Leitch, R. (2005). Balanced Scorecard: Evening the Odds of Successfull BPR. *IT Pro*, *7*(6), 24–20.

IBM. (2002). *Integrating WebSphere Portal with your security infrastructure.* (Tech. Rep.). `www.ibm.com/websphere`: IBM Corporation.

IBM. (2003). *IBM Tivoli Access Manager: Base Administration Guide - Ver5.1* (Tech. Rep.). `www.ibm.com`: IBM.

IBM. (2005a). *Information infrastructure: Delivering the advantage of information integration today* (Tech. Rep.). `www.ibm.com/websphere`: IBM Corporation.

IBM. (2005b). *Integrating information for the on demand business* (Tech. Rep.). `www.ibm.com/websphere`: IBM Corporation.

International Accounting Standards Board (IASB). (2005). *International Financial Reporting Standards (IFRSs) Volume 1B.* International Accounting Standards Committee Foundation (IASCF).

Internet Usage Statistics - The big picture. (2006). (`http://www.internetworldstats.com` (last cited: 3 October 2006))

Irwin, K., Yu, T., & Winsborough, W. H. (2006). On the modelling and analysis of obligations. In *CCS '06: Conference on Computer and Communications Security* (pp. 134–143). Alexandria, Virginia, USA.

ISO (Ed.). (1989). *Information processing systems Open Systems Interconnection Basic Reference Model Part 2: Security Architecture.* http://www.iso.org: ISO.

Kaplan, R., & Norton, D. (1996). *Using the balanced scorecard as a strategic management system.* Harvard Business School Press.

Karjoth, G. (2003). Access Control with IBM Tivoli Access Manager. *ACM Transactions on Information and Systems Security*, *6*(2), 232–257.

Knight, S. (1999). The role of ADSL in Internet access. *Computer*, *32*(7), 103–106.

Kroeber, A., & Parsons, T. (1958). The Concept of Culture and of Social System. *American Sociological Review*, *5*(23), 582–583.

Lacoble, K. (1987). Designing user services on a PC LAN. In *SIGUCCS '87: Proceedings of the 15th annual ACM SIGUCCS conference on User Services* (pp. 367–376). New York, NY, USA: ACM Press.

Leavitt, H., & Whisler, T. (1958). Management in the 1980's. *Harvard Business Review*, *36*(6), 41–48.

Lorch, M., Proctor, S., Lepro, R., Kafura, D., & Shah, S. (2003). First experiences using XACML for access control in distributed systems. In *XMLSEC '03: Proceedings of the 2003 ACM workshop on XML security* (pp. 25–37). Fairfax, Virginia.

Madsen, P. (2005). *SAML 2: The building Blocks of Federated Identity.* `http://www.xml.com/pub/a/2005/01/12/saml2.html`: XML.com.

Madsen, P., & Maler, E. (Eds.). (2005). *SAML V2.0 Excecutive Overview.* http://www.oasis-open.org/committees/download.php/13525/sstc-s%aml-exec-overview-2.0-cd-01-2col.pdf: OASIS.

Marchand, D., Kettinger, W., & Rollins, J. (2000). Information Orientation: People, Technology and the Bottom Line. *Sloan Management Review*, *41*(4), 69–80.

Marnet, O. (2007). History repeats itself: The failure of rational choice models in corporate governance. *Critical Perspectives on Accounting*, *19*(2), 191–210.

Massie, J. (1979). Development of Management Thought. In S. Robbins (Ed.), *Essentials of management* (pp. 13–26). Prentence-Hall, INC., Englewood Cliffs, New Jersey 07632.

Merriam-Webster Online Dictionary. (2004). (http://www.m-w.com (last cited: 13 August 2004))

Meyers, M. (2002). *Rethinking Performance Measurement: Beyond the Balanced Scorecard.* Cambridge University Press.

Mische, M. (2002). Defining Systems Integration. In J. Myerson (Ed.), *Enterprise systems integration* (pp. 3–10). Auerbach Publications.

Mollick, E. (2006). Establishing Moores's Law. *IEEE Annals of History of Computing*, *28*(3), 62–75.

Mondai, S., & Gupta, K. (2000). Choosing a Middleware for Web-Integration of a Legacy Application. *Software Engineering Notes*, *25*(3), 50–53.

Moses, T. (Ed.). (2005). *eXxtensible Access Control Markup Language (XACML) version 2.0.* Available from: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec%-os.pdf: OASIS Standard. (Last cited: 20 Nov 2006)

Munawer, Q. (2000). *Administrative models for role-based access control.* Unpublished doctoral dissertation, George Mason University, Fairfax, Virginia.

Nadler, D., & Tushman, M. (1997). A Congruence Model for Organization Problem Solving. In M. Tushman & P. Anderson (Eds.), *Managing Strategic Innovation and Change* (pp. 159–171). Oxford University Press.

Niven, P. (2005). *Balanced Scorecard Diagnostics.* John and Sons, Inc., Hoboken, New Jersey.

Nohria, T. (1991). Note on Organization Structure. *Harvard Business School Case, DOI: 10.1225/491083.*

Nomura, T. (2001). How the Knowledge Dynamics of Individuals, Communities, and Ba Drive the Knowledge Management. In *Proceedings of the 10th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE '01)* (p. 251-256). IEEE Computer Society.

Oppliger, R. (2001). *Security Technologies for the World Wide Web.* 685 Canton Street Norwood, MA 02062: Artech House, INC.

ORACLE. (2004). *Oracle User Management 11i datasheet* (Tech. Rep.). `www.Oracle.com`: Oracle Corporation.

Palmer, D., & Morris, B. (1980). *Computing Science.* Edward Arnold Publishers.

Paulson, L. (2005). Will Hard Drives Finally Stop Shrinking. *Computing*, *38*(5), 14–16.

Peden, M., & Young, G. (2001). From voice-band modems to DSL technologies. *International Journal of Network Management*(11), 265–276.

Ramachandran, S., & Rao, S. V. (2006). An effort towards identifying occupational culture among information systems professionals. In *SIGMIS CPR '06: Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research* (pp. 198–204). Claremont, California, USA.

Ramakrishnan, L. (2004). Securing Next-Genration Grids. *IT Pro*, *6*(2), 34–39.

Richardson, H., & Howcroft, D. (2006). The contradictions of CRM - a critical lens on call centres. *Information and Organization*, *16*(2), 143–168.

Roberts, L. (1986). The ARPANET and computer networks. In *Proceedings of the ACM Conference on The history of personal workstations* (pp. 51–58). Palo Alto, California, United States.

Robinson, T. (2005). Data security in the age of compliance. *netWorker*, *9*(3), 24–30.

Rolls, D. (Ed.). (2003a). *Service Provisioning Markup Language (SPML) Version 1.0.* `http://www.oasis-open.org`: OASIS.

Rolls, D. (Ed.). (2003b). *Service Provisioning Markup Language (SPML) Version 1.0.* Available from: `http://www.oasis-open.org/committees/provision/docs/cs-pstc-spml-core-1%.0.doc`: OASIS Specification. (Last cited: 20 Nov 2006)

Sadler, P. (1997). Designing Organizations for Employee Commitment. In *The Seamless Organization: Building the Company of Tomorrow* (pp. 84–104). Kogan Page.

Saltzer, J. H., & Schroeder, M. D. (1975). The protection of information in computer systems. *Proceedings of IEEE*, *63*(9), 1278–1308.

Samarati, P. (2002). Enriching Access Control to Support Credential-Based Specifications. In *Informatik bewegt: Informatik 2002 - 32. Jahrestagung der Gesellschaft fr Informatik e.v. (GI)* (pp. 114–119). GI.

Sandhu, R., Coyne, E., Feinstein, H., & Youman, C. (1996). Role-Based Access Control Models. *IEEE Computer*, *29*(2), 38 – 47.

Sandhu, R., & Munawer, Q. (1999). The ARBAC99 Model for Administration of Roles. In *ACSAC '99: Proceedings of the 15th Annual Computer Security Applications Conference* (p. 229). Washington, DC, USA: IEEE Computer Society.

Sandhu, R., & Samarati, P. (1997). Authentication, access control and intrusion detection. In *The Computer Science and Engineering Handbook* (pp. 1929–1948). CRC Press.

Sandhu, R. S. (1993). Lattice-based access control models. *IEEE Computer*, *26*(11), 9–19.

SAP AG. (2002). *Security in the mySAP Enterprise Portal* (Tech. Rep.). `www.sap.com`: SAP AG.

SAP AG. (2003). *SAP Enterprise Portal - Portal Content* (Tech. Rep.). www.sdn.sap.com: SAP AG.

SAP AG. (2004). *Role Concept in SAP Enterprise Portal 5.0 and 6.0* (Tech. Rep.). www.sdn.sap.com: SAP AG.

Sharrock, R. (2002). *Business Transactions Law* (6th ed.). Juta and Co, Ltd.

Sherburne, P., & Fitzgerald, C. (2004). You Don't Know Jack About VoIP. *Queue*, *2*(6), 30–38.

Sherif, K., & Xing, B. (2006). Adaptive processes for knowledge creation in complex systems: the case of a global IT consulting firm. *Information and Management*, *43*(4), 530–540.

Shin, D., Ahn, G.-J., Cho, S., & Jin, S. (2003). On modeling system-centric information for role engineering. In *SACMAT '03: Proceedings of the eighth ACM symposium on Access control models and technologies* (pp. 169–178). Como, Italy.

SQL Server 2000 SP3 Security Features and Best Practices: SQL Server 2000 Security Model. (2003). (`http://www.microsoft.com/technet/prodtechnol/sql/2000/maintai%n/sp3sec01.mspx` (last cited: 26 August 2005))

Takahashi, K., & Yana, E. (2000). A Hypermedia Environment for Global Collaboration. *IEEE MultiMedia*, *7*(4), 36–47.

Thomas, B. (1997). E-mail Attachments: Finding the Right Fit. *IEEE Internet Computing*, *1*(3), 78–79.

Thomas, R., & Sandhu, R. (1993). Towards a task-based paradigm for flexible and adaptable access control in distributed applications. In *Proceedings*

of 1992-1993 ACM SIGSAC New Security Paradigms Workshop (pp. 138–142). New York, NY, USA: ACM Press.

Thomas, R., & Sandhu, R. (1994). Conceptual foundations for a model of task-based authorizations. In *Proceedings of the 7th IEEE Computer Security Foundations Workshop* (pp. 66–79). Franconia,NH.

Thomas, R., & Sandhu, R. (1997). Task-based Authorization Controls (TBAC): A Family of Models for Active and Enterprise-oriented Authorization Management. In T. Lin & S. Qian (Eds.), *Database Secutity, XI: Status and Prospects – Results of the IFIP WG11.3 Workshop on Database Security* (pp. 166–181). Chapman and Hall.

Townsend, A. M. (2002). Mobile and wireless technologies: emerging opportunities for digital government. In *dg.o '02: Proceedings of the 2002 annual national conference on Digital Government Research* (pp. 1–5). Digital Government Research Center.

User Provisioning. (2006). (`http://www.courion.com/products/acc/index.asp` (last cited: 10 October 2006))

Von Solms, R., & Von Solms, S. H. (2006). Information Security Governance: A model based on the Direct-Control Cycle. *Computers & Security, 25*(6), 408–412.

Von Solms, S. (2005). Information Security Governance - Compliance management vs operational management. *Computers & Security, 24*(1), 443–447.

Von Solms, S., & Hertenberger, M. (2005). ERPSEC - A reference framework to enhance security in ERP systems. In *20th IFIP International Information Security Conference* (pp. 79–94). Makuhari Messe, Chiba, JAPAN.

Watts, P. (2006). Compliance management in the corporate world. *Computer Fraud and Security, 2006*(12), 19–20.

Websphere Home on IBM website. (2005). (`http://www.ibm.com/websphere` (last cited: 26 August 2005))

Wolff, E. (2005). The Growth of Information Workers in the U.S. Economy. *Communications of the ACM*, *48*(10), 37–42.