Using **Continuous Assurance** to Detect Fraud in E-Commerce Transactions

Stuart Loh
Thesis

Supervisor: A/Prof. Rodger Jamieson

Submitted for the degree of **BSc (BIT) (Honours)**
School of Information Systems, Technology and Management
The University of New South Wales
2002

Abstract

This thesis is concerned with improving our understanding of detecting and preventing fraud in electronic commerce (e-commerce) transactions by using continuous assurance systems. It also seeks to evaluate the usefulness of eSCARF, a continuous assurance system for fraud detection, which is developed in this thesis. The area of electronic fraud was targeted as it is one of the major risks for businesses engaging in the rapidly growing practice of e-commerce today. The ability to mitigate this risk is valuable to businesses and auditors, and continuous assurance systems, which may provide assurance services in real-time, offer such an ability. A conceptual model was constructed to produce a generalised overview of the fraud auditing environment, and the objects and forces influencing the process. This allows us to better understand and visualise the relationships between all these issues.

The second part of this thesis developed a continuous assurance system that may be used to combat electronic fraud. This system is called eSCARF (electronic System Control Audit Review File), designed for the IBM WebSphere Commerce 5.4 e-commerce system. The development of eSCARF is documented and provides insight into the architecture of a continuous assurance system.

The third part of this thesis involves a user evaluation of eSCARF by 15 auditors via an evaluation survey. The evaluation survey assessed the quality and perceived usefulness of the system. The survey discovered that the participants regarded eSCARF as a highly usable system with clear indications of its usefulness in effectively detecting e-commerce fraud. Further input gathered from auditors provided ways eSCARF could be enhanced. With this information and the verification of eSCARF's feasibility and applicability for fraud detection, future avenues for eSCARF's continued development are mapped out.

Statement of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis.

I also declare the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.

Stuart Loh
27 November, 2002

For Mum and Dad

Acknowledgements

This thesis would not have been possible without the knowledgeable and efficacious guidance of my honours supervisor, Dr Rodger Jamieson. His continual insight, perspicacity, feedback and support throughout the duration of this thesis is very much appreciated.

Thanks to Jack Verdins, market manager for IBM WebSphere software, for supporting the academic use of WebSphere in the pursuit of this thesis. Thanks also to Justine Middleton, also of IBM, for giving up much of her time to assist me with the installation of WebSphere Commerce.

I am also grateful to Ken Stevens for his assistance in pilot testing the survey, as well as to all the survey participants who took the time out of their busy schedules to participate. Thanks to Kevin Yeoh and Sophia Lan for their support, advice and help with proofreading.

To my family and friends who have had to involuntarily endure my more irate moods in the stressful moments of this year, I also extend my gratitude. Finally, everlasting thanks to my parents who have always given me their unconditional support.

“Fraud and falsehood only dread examination.
Truth invites it.”

- Thomas Cooper

Contents

List of Figures

List of Tables

Chapter 1. Introduction

The growth of electronic commerce ('e-commerce') in the world has been dramatic over the past few years, with forecasts suggesting that this explosive trend will continue (Pastore 2002). The birth of the dot-coms who have capitalised on the advantages e-commerce provides, such as lower barriers to market entry, as well as the extensive integration of e-commerce systems into incumbent organisations is testament to this. This growth has arisen as the benefits of e-commerce have been realised by businesses and consumers alike. In the 'digital economy', e-commerce has facilitated efficient information exchange, enabled cost reductions, provided new revenue opportunities and increased process efficiencies. Customers also reap these benefits through the reception of better customer service and the greater convenience of paperless online transactions (Turban 2000). E-commerce is ever more becoming intertwined with how organisations do business and are operated.

However, this tight integration with e-commerce has increased the exposure of businesses to a broader range of risks such as security, privacy and reliability concerns (Daigle and Lampe 2000). Actual and perceived security concerns, in particular, are large barriers preventing a more rapid uptake and growth of e-commerce (Elliot and Fowell 2000). Managing these risks becomes of great importance to companies partaking in any e-commerce operation – both in protecting company e-commerce revenue flows from security-related mishaps (such as fraud, theft and systems failure), and in assuring hesitant customers of the safety of engaging in e-commerce.

One of the chief security risks is fraud (Anandarajah and Lek 2000; Cerpa and Jamieson 2001). In any transaction, participants want to ensure proper receipt of payment in exchange for goods or services. Failure by one party to receive what they expect may indicate the occurrence of fraud. As a consequence of the advent of e-commerce, new methods of carrying out financial transactions mean that new methods by which fraud is perpetrated also arise (such as shill bidding in online auctions (Wang, Hidvégi and Whinston 2001b)). The presence of fraud, or even the threat of it is a deterrent to businesses and customers alike, who may choose to resort to more traditional means of performing transactions (Elliot and Fowell 2000). Currently, a variety of tools using a myriad of approaches to detect fraud do exist, but their use is limited, fragmented and their effectiveness is untested (section 3.2.4). The fraud detection solutions that do exist for businesses engaging in e-commerce tend to be proprietary in nature and how they work is unpublicised. The electronic environment therefore has need for effective controls, built around a generalised, tested framework, that will mitigate the risk of fraud that e-commerce poses. It is clear that businesses stand to benefit from the ability to reduce fraud, but the development of these controls is also important for auditors. An essential responsibility of auditors is to plan and conduct audits for irregularities induced by fraud, other illegal acts and errors, that impact upon the financial reports of an entity (AUS210 2002; Baer 2002; AICPA 2002). Therefore, enabling better detection of fraud would facilitate an auditor's job.

Fraud prevention is difficult in the faceless world of the Internet, and any measure designed to respond to it must be able to do so in a timely manner. Continuous assurance (CICA 1999) offers a timely method of assurance where, by monitoring transactions (flows of information, especially payment and order details) in real-time, irregularities that point to illicit behaviour may be promptly detected and dealt with. Continuous assurance systems capitalise on the infrastructure and real-time nature of e-commerce systems. In fact, continuous assurance systems rely on the system being assured to be a quick and reliable source of relevant data, because the assurance system must, in turn, provide its own service of delivering timely assurance and reporting information (Vasarhelyi, Kogan and Sudit 2000). Such a system will be able to detect fraudulent activity in an e-commerce system in an unobtrusive manner. There is a need to develop an assurance system that can be easily integrated into existing systems, be flexible to adapt to different organisations and organisational change, and provide control over the assurance process (Vasarhelyi, Kogan and Sudit 2000).

This thesis has a variety of aims, focusing primarily on improving our understanding of detecting and preventing fraud in e-commerce systems by the use of continuous assurance systems. A conceptual model relating the aspects and concepts associated with the real-time monitoring of e-commerce transactions for fraud will be developed.

Explored along the way will be the various continuous assurance methods for e-commerce fraud detection, before we finally settle on using the SCARF (Systems Control Audit Review File) technique to implement an assurance system that will provide assurance for financial transactions for a business to consumer e-commerce store. This system, called eSCARF (electronic SCARF), will be adapted for the IBM WebSphere Commerce (interchangeably referred to as WebSphere) environment from a prototype eSCARF system developed by Ng and Wong (1999). Following the implementation of eSCARF, a user evaluation of the system will be acquired from professionals with experience in auditing. The evaluation will be obtained by performing an evaluation survey, assessing attributes of the system from an auditor's perspective, looking at its design quality and perceived usefulness. This will aid the future and ongoing development of eSCARF by providing valuable user input, as well as shedding further insight into auditors' requirements for continuous assurance systems.

Chapter 2. Research Objectives and Significance

The main objective of this research is to determine if eSCARF is useful in the continuous assurance of e-commerce systems for the detection of fraud. In doing so, our understanding of detecting fraud in e-commerce transactions by using continuous assurance systems will be improved. The research will investigate using continuous assurance techniques to monitor transactions on e-commerce systems for fraudulent activity. This should contribute to organisations' and auditors' ability to detect the perpetration of fraud in e-commerce systems and hence enable them to reduce its occurrence. The main objective can be translated into an overarching research question that this thesis will address:

> **How is eSCARF useful, as a continuous assurance system, for the detection of fraud in e-commerce transactions?**

This main research question can be divided into five research questions (RQs) which drive this investigation:

RQ1. How can the problem of electronic payment fraud be addressed?

RQ2. Can continuous assurance be applied to e-commerce systems in order to detect fraud?

RQ3. How does a continuous assurance system (namely, eSCARF) function?

RQ4. Do auditors perceive that continuous assurance systems (namely, eSCARF) are useful for assuring e-commerce systems?

RQ5. What factors are important to the design of a continuous assurance system using the SCARF audit technique?

This research is divided into three sections, one theoretical, and two practical in nature. RQ1 and RQ2 will primarily be addressed in the theoretical component. In the literature review, we seek to improve our understanding of the nature and problem of fraud in e-commerce, as well as detailing methods by which it may be combated. The developing field of continuous assurance will be discussed in detail due to its perceived applicability towards detecting electronic fraud.

A conceptual model for continuously assuring e-commerce systems for fraudulent transactions will be developed to also address RQ1 and RQ2. This model is based on Activity Theory and will identify the subjects, instruments, rules and other environmental factors that are involved. How these all interact with each other to achieve the outcome of providing continuous assurance of e-commerce transactions will also be explored. The literature review and conceptual model will then provide a theoretical grounding for the rest of the thesis.

The second component of this research will implement a continuous assurance system that will address RQ3, as well as enabling RQ4 and RQ5 to be answered. It will involve a build and test of a system that will audit e-commerce transactions for payment fraud. A continuous assurance system will be built for the IBM WebSphere Commerce e-commerce system using Java. This system will be adapted from a rudimentary eSCARF prototype software developed by Ng and Wong (1999). The system implementation process will allow us to better understand continuous assurance systems and how they interact with the e-commerce systems they assure.

The third component will consist of a user evaluation survey by professionals with experience in auditing, that will examine the implementation of eSCARF from an auditor's perspective in an effort to answer RQ4 and RQ5. The survey will be partially quantitative (in measuring constructs related to system quality and effectiveness, such as usability and accuracy), as well as partially qualitative (in that it will also gather suggestions for system improvement, and general commentary about the system, from the participants). Results from the survey provide user input into the current design of eSCARF which will aid its future development. This information also improves our understanding of continuous assurance systems by exploring what auditors desire in them.

In summary, the theoretical component is about theory development, while the practical component consists of software implementation and an evaluation survey of the software implemented.

Furthering our understanding of fraud and fraud detection in e-commerce systems is significant in several ways. Business organisations will gain a better understanding of fraud and be able to address it in a more systematic and informed manner than is currently done in industry. For business organisations, the problem of electronic fraud is a large one. If fraud detection methods can be improved, then fraud can be reduced, which produces a few benefits – apart from reducing financial losses sustained from fraud, reducing it will also have the effect of increasing the confidence of e-commerce users and increasing adoption and usage of e-commerce systems (Elliott 2002).

For auditors, the creation of an auditor evaluated continuous assurance system will allow future work in assuring e-commerce transactions to be performed more effectively. This allows them to better meet their responsibilities in detecting irregularities induced by fraud that impact upon businesses (AUS210 2002) by providing them with a tool that can manage fraud detection.

In the academic sphere, this research will provide a better understanding of fraud, of fraud detection methods that have been evaluated by experts in the field of auditing, and will produce a continuous assurance system which can be used as a tool in future research. Developing upon eSCARF will provide insights into the technical architecture of continuous assurance systems, one of the issues Alles, Kogan and Vasarhelyi (2002) cite as an avenue for further research concerning continuous assurance.

Chapter 3. Literature Review

3.1 The Electronic Commerce Environment

Electronic commerce relates to the usage of electronic communication networks to conduct business transactions (Turban 2000). The emergence of e-commerce in society has profoundly impacted upon how people manage and conduct business. It has changed how companies operate internally, whilst also giving them the opportunity to expand into new, previously untappable, markets. The ubiquitous nature of e-commerce has also accelerated globalisation as instantaneous information exchange is possible anywhere on the planet. The smallest of firms employing e-commerce potentially have access to a global market. The largest of firms have redefined or remodelled themselves in response to the advent of e-commerce. Indeed, e-commerce not only affects the way business is conducted, but its nascent influence reverberates through to changing the world economy (Nezu 2000).

Nonetheless, this new dimension of business has problems, barriers and disadvantages that inhibit its expansion. It is a phenomenon undergoing continual, rapid change and maturity. Increasing levels of integration of e-commerce systems into business has led to an increasing level of reliance on these systems. Interorganisational systems and globally distributed data means that ensuring the availability, integrity and confidentiality of the information these systems process is of paramount importance. Unfortunately, it is the pace of e-commerce system development that amplifies the huge challenge of ensuring those same systems are secure. This thesis examines specifically the threat of fraud which is the largest security risk that has direct implications upon the revenue flows and costs of a business.

It is for this reason that e-commerce security should receive collaborative attention from research institutions and commercial organisations, such that security may be able to keep in step with the latest advancements in e-commerce (Anandarajah and Lek 2000). Current approaches tend to be fragmented in nature, due to the wide variety of systems in the marketplace, and the trend of interorganisational systems integration means that unless a more unified approach to suring up security is taken, the rise in number of points a large e-commerce system has that are exploitable will be increasingly detrimental. A system vulnerable to different types of fraud stands to be a large liability over more traditional means of business and undermines the attractiveness of e-commerce. Moreover, customers that perceive that their e-commerce transactions are susceptible to fraud are not encouraged to engage in such business (Elliot and Fowell 2000). Only when security systems are developed that can, with a reasonable degree of effectiveness, detect fraud, will this barrier to e-commerce uptake be assuaged.

3.1.1 The Impact of E-Commerce

There can be no denial that e-commerce has made a definite and significant impact upon the global economy. Its integration into society has affected the ways people manage and conduct business. The spread of e-commerce will continue as organisations use it to increase productivity as well as another avenue for sales and service. In fact, Clarke (1993) predicts that business-to-business (B2B) and business-to-consumer (B2C) e-commerce will become so popular that most businesses will be forced to enter the digital economy in order to retain competitive advantage.

In a study encompassing the first half of 2000, the Internet Economy was, in the United States, found to support more than 3 million workers (CREC 2001). Online businesses numbered 550,000 by mid-2000 (Cerpa and Jamieson 2002), up 30 percent from the previous year. The United States Department of Commerce estimated that retail e-commerce sales for the fourth quarter of 2001 totalled $10 billion (Pastore 2002), up from $5.3 billion in the same period in 1999 (Armstrong 2000). In contrast, total retail sales were $821.2 billion and $860.8 billion in the fourth quarter of 1999 and 2001 respectively. Although e-commerce only accounts for a miniscule portion of all retail sales, e-commerce sales have doubled proportionate to total retail sales in the two year period, reflecting an increasing amount of e-commerce usage. That e-commerce sales only compose about one percent of total retail sales demonstrates there is plenty of room for e-commerce to continue expanding into.

From a worldwide perspective, IDC found that e-commerce spending grew 68 percent between 2000 and 2001 to reach $600 billion. IDC estimates that this will continue increasing to a massive $1 trillion in 2002 (Pastore 2002). The numbers above are primarily in reference to B2C transactions. It is postulated that B2B transactions outstrip B2C ones with the Gartner Group predicting 2004 worldwide B2B revenues at $7.3 trillion. It is this profit potential that has lured venture capitalists into investing into ‘dot com’ companies which are trying to ‘ride the wave’ and establish themselves as profitable businesses.

There are many other statistics that may be cited. However, one thing is clear – that e-commerce’s prominence in business is increasing. In the next few years, this growth is forecasted to continue unabated.

From an organisational and management perspective, the changes e-commerce has wrought have been just as dramatic. Most notably, the restructuring of the ‘Big Five’ multinational accounting firms to separate their e-commerce consulting arms from their auditing arms. The impetus for this is to ensure that their audit work is not compromised as a conflict of interest exists if a firm both consults and audits the same client (Kane 2002). Accenture’s separation from Andersen, as a result, also gave it independence such that when Andersen was shaken by the collapse of Enron and consequential legal proceedings, Accenture was relatively untouched. PricewaterhouseCoopers has spun off its consulting arm which was acquired by IBM, Deloitte Touche Tohmatsu spun off its consulting arm into Braxton, with KPMG likewise turning theirs into BearingPoint.

Apart from sales and marketing, e-commerce systems are also employed for operational and supply purposes, including finance, logistics and procurement. Incumbent firms especially have managed to take advantage of these types of systems, enabling cost reduction and greater process efficiencies (Turban 2000).

3.1.2 Advantages of E-Commerce

E-commerce possesses a variety of attributes that have made it attractive to businesses and to their customers. Implementing e-commerce systems has enabled organisations and their customers to conduct business in ways previously not possible.

A company conducting e-commerce via the Internet has, potentially, a global reach, due to the interconnected, ubiquitous nature of it. This has greatly lowered the barriers of entry for many industries, especially those conducive to electronically-based ventures, as opposed to more traditional and well established markets (Dertouzos 2000). Previously, penetrating into a market, especially a geographically based one, required organisations to have a physical presence and physical assets. E-commerce allows businesses to have market presence without physical presence, and thus less capital is required. Systems are also available 24 hours, 7 days a week, therefore are not limited by time zone constraints (Vacca 1995). As a result, many 'dot com' companies such as Dell have been able to successfully enter markets dominated by traditional incumbent giants like Hewlett-Packard and IBM. Additionally, intangible assets such as technological innovation and strong customer relationships are increasingly important in a business' strategic planning (Anandarajah and Lek 2000). The ability to leverage new technology and quickly establish strong brand equity has allowed Amazon.com to prosper against incumbent Barnes and Noble. Even though Barnes and Noble entered the e-commerce marketplace shortly afterwards, Amazon.com had created enough brand loyalty such that it eventually returned a profit.

In a Forrester Research survey (Martin 2002), over 50 percent of B2C respondents indicated that penetrating new markets is the reason they are involved in e-commerce, followed by the ability to deliver a better quality of service. For B2B firms, over 30 percent cited lower operating costs as their motivation, followed by over 25 percent citing better information delivery. Therefore, e-commerce offers more than just new market opportunities for business.

Performing transactions electronically decreases the cost of creating, process, distributing, storing and retrieving paper-based information. For example, the implementation of an electronic procurement system may reduce a company's administration costs by up to 85 percent (Turban 2000). E-commerce may incite business process re-engineering, as e-commerce systems change the way business processes interact. Supply chain management and procurement systems necessarily remodel and optimise supply chain processes (Turban 2000).

The open nature of e-commerce allows businesses to select from a greater variety of business partners – the opportunity for businesses to build network-based *ad hoc* partnerships (Wang et al. 2000). The formation of strategic alliances with other firms, and achieving knowledge sharing through the integration of their systems, allows firms to diversify, leveraging on each others' expertise. The result is businesses being able to adapt and to respond to shifting market forces. For example, Amazon.com's alliance with Drugstore.com allowed Amazon to diversify into the pharmaceuticals market, while giving Drugstore access to Amazon's large customer base.

The advantages are not restricted to businesses alone, though. For e-commerce to work, incentives for customers must exist as well. Greater information accessibility is enabled via e-commerce. Information that may have taken days to obtain is now available to customers immediately from a single computer terminal, enhancing the level of customer service. Furthermore, because the information is exchanged in electronic format, customers are able to integrate transaction processing into their own e-commerce systems, thereby automating the procurement process and reducing costs. EDI systems have allowed this in the past, although XML-based information is increasingly being used as a standard format for information exchange over the Internet (Yeomans 2001). The advantages of paperless transactions are therefore applicable to customers.

Customers also get more choice in terms of vendors. Compared to traditional retailing, each electronic vendor is just as accessible as another, providing customers with greater convenience. Quick price comparisons between a large number of competing vendors is also available. Online marketplaces allow customers to place requests for tenders, and businesses to bid for them. Finally, quicker delivery of products is possible, especially with digitised products (Turban 2000).

3.1.3 E-Commerce Risks and Weaknesses

Despite the prodigious growth of e-commerce, the new environment is not without its weaknesses. From these weaknesses arise risks that must be noted and accounted for, lest they are exploited by the unscrupulous or cause other unintended disruption. E-commerce is relatively new, and that, combined with the swiftness of its uptake, has meant that there are few formalised methodologies and guidelines in place for developing those systems (Ng and Wong 1999). Systems, as a result, have weaknesses that may be exploited. As businesses and customers become increasingly reliant on their e-commerce systems, they should be aware of the risks they have increased exposure to.

Weaver et al. (2000) discuss four different possible trends in the future growth of e-commerce. Two of the possible trends include a decline in e-commerce sales, both due to flaws in e-commerce (one which the Internet recovers from, and one which is irrecoverable from such as the breaking of commonly used encryption standards). Consequently, it is important to identify these flaws lest e-commerce suffers from one of them.

E-commerce must be regarded as a safe and practical way to do business, by both businesses and customers, to succeed. The feasibility of e-commerce is well proven, although it is clear that many are not convinced that e-commerce is secure. A majority of the insecurities associated with e-commerce fall under the branch of ‘electronic crime’ (ACPR 2001), which is crime perpetrated via information systems. This encompasses both traditional ‘real world’ crimes that have migrated online (such as credit card fraud), as well as new crimes that have arisen alongside the development of new information systems (such as viruses).

Insecure systems are a recurring theme when e-commerce flaws are discussed in literature. Weaver et al. (2000, p. 30) say that *“at least two major research areas will affect the growth–or nongrowth–of Internet businesses over the next three to five years: wireless technology and security.”* Udo’s (2001) survey concludes that security forms a major barrier for the spread e-commerce with regards to consumers: *“Security concern is one of the main reasons Web users give for not purchasing over the Web.”* (Udo 2001, p. 166) An empirical study by Elliot and Fowell (2000) of consumer experiences with e-commerce retailing discovered that 50 percent of transactions rated as unsatisfactory stemmed from security concerns. Other factors forming barriers are privacy issues, censorship concerns, e-mail safety concerns and impersonation/forged identity concerns (which are security related).

For businesses, security is a major issue. The case of the ‘love bug’ virus, which crippled many corporate e-mail systems and costed billions of dollars in lost productivity is widely cited in research (Wang, Hidvégi and Whinston 2000; Udo 2001; Weaver et al. 2000). A distributed denial of service attack occurred in February 2000 which disabled numerous large sites of businesses including eBay, Microsoft and Yahoo. Following the attack, which caused a 22 hour outage on eBay, eBay experienced an 18 percent drop in share price and an immediate 43 percent decline in business volume (O’Brien and Mercer 2002). Other security breaches, such as the Code Red worm, and various hacking attacks which have disclosed databases of customer credit card details (eg: Leyden 2000) all lower customer and business confidence in e-commerce. More traditional crimes, have found their way online. Most prominent of these is fraud, which, in an online environment, can be perpetrated in ways unique to e-commerce (for example, the faceless nature of the Internet means that properly identifying customers is difficult, if not impossible).

Not surprisingly then, e-commerce security and protecting against electronic crime is a significant and fertile field of research (Cerpa and Jamieson 2001). Security controls are necessary to mitigate these risks and remove some of the barriers inhibiting the growth of e-commerce. Accordingly, a myriad of control frameworks and approaches have been devised in the research literature. These include research on introducing security into e-commerce systems at design-time, through secure mechanism design (Wang, Hidvégi and Whinston 2000), the development of architectures for real-time intrusion monitoring (Furnell and Dowland 2000) and the use of continuous auditing (Cerpa and Jamieson 2001).

This thesis addresses one facet of e-commerce security, albeit a crucial one. A central unit of e-commerce is the transaction. Transactions form the lifeblood of e-commerce in which information and money are traded amongst businesses and consumers. Thus, ensuring that these transactions are legitimate in nature (i.e. not fraudulent), and that participating parties receive what they expect from them, is a fundamental issue of e-commerce. If businesses and consumers are unconvinced that performing transactions on the Internet is secure, a lack of trust will develop. This lack of trust acts as an inhibiting factor (Elliot and Fowell 2000). This thesis focuses on transaction security – or more specifically, the fraud auditing of transactions.

3.2 Electronic Fraud

Fraud in e-commerce occurs in many guises. By definition, fraud is where one party in a transaction makes a knowingly false representation of a fact to deceive the other party (Garner 1999). In e-commerce, fraud can apply to any type of transaction where information is electronically exchanged between two parties engaging in business. Intentional misrepresentation of information, as supplied by either party, constitutes fraud.

There are many different categories of fraud. Two of these will be discussed that are particularly pertinent to the domain of e-commerce transactions. These are identity fraud and payment fraud. Identity fraud is where a person pretends to be someone they are not in order to deceive someone else. Given the faceless nature of the online world, identity fraud is often a precursor for committing many other types of crime. By using a fraudulent identity, other types of fraud may be perpetrated, most notably, payment fraud (Doocey 2002). Payment fraud is a more specific type of fraud than identity fraud and concerns a payment transaction – a financial or other type of payment that is made in exchange for goods or services. Some forms of payment fraud, such as credit card fraud, are even construed as being a form of identity fraud as well, since information is used which illegitimately identifies the person committing fraud.

Because identity fraud plays a significant role in acting as a starting point for many instances of payment fraud (and many other types of crime), an understanding of identity fraud will help us also recognise how payment fraud arises and how it is perpetrated. In the next section, we will briefly examine what constitutes identity fraud, and the consequences of it. The continuous assurance system implemented in chapter 6 is aimed at detecting B2C payment fraud as it is one of the most common forms of fraud in e-commerce. Therefore, B2C payment fraud will also be discussed, in order to gain an understanding of how it occurs such that we may address this problem with a continuous assurance system.

3.2.1 Identity Fraud

Identity fraud can be defined as, *"The possession and/or use, or intent to use, fraudulent and/or stolen documentation and/or identity information to deceive a third party for a benefit."* (AUSTRAC 2001) Identity fraud basically involves a party pretending to be someone they are not, in order to deceive another party for some sort of benefit.

There are many aspects that make up a person's identity. Identity fraud involves fabricating, or stealing from someone else, any number of these aspects. A UK Cabinet Report (Cabinet 2002) delineated three elements of personal identity: biometric, attributed and biographical identity. Biometric identity includes biological traits unique to an individual, such as DNA profile, fingerprints and facial structure. Attributed identity includes components of a person's identity imparted to them at birth, such as name, date of birth and parents' names. Biographical identity refers to details of a person's life built up over time. This includes things like employment history, registration of marriage, credit histories and so on.

Identity fraud is a relatively new concept in society and research, and as a result, currently lacks a standardised definition in literature (Matejkovic and Lahey 2001). This means that determining what activities may be classified under identity fraud is not always clear, as identity fraud itself overlaps with many other areas of crime. For instance, credit card fraud is considered by some (but not by all) to come under the umbrella of identity fraud as it involves misrepresentation of credit details, which in turn form part of a person's biographical identity. The United States Federal Trade Commission released a report detailing the most common types of identity fraud committed (FTC 2000). Credit card fraud (50%) was the most prominent means by which identity theft occurred, followed by unauthorised usage of phone or utility services (25%), bank fraud, which includes drawing fraudulent cheques and opening bank accounts under false identities (16%), fraudulent loans (9%) and fraud used to obtain government documents or benefits (8%).

Regardless of what activities are classified under identity fraud, the repercussions of it are widespread and serious. Nguyen (2002) notes that identity fraud flows on to financial crime (payment fraud), electronic crime (where criminals use the Internet to maintain anonymity), immigration offences, drug trafficking and a host of other types of crime – both organised and non-organised – occurring at all scales from local to global.

The market for the theft of identities is huge, and growing. In November 1998, the US Immigration and Naturalization Service seized almost 2 million counterfeit documents, including social security and residency cards (Airports 2002). 'Cyberbazaars' in Russia sell off thousands of illegally obtained credit card details and identities to the highest bidder (Doocey 2002).

The methods by which identity fraud is perpetrated are numerous. Federal Agent Gordon Williamson, from the Australia Federal Police postulated that the concerns about false identity, *"essentially revolve around the ease of availability of some documents which can then be used to prove identity and the ease with which technology permits the falsification of documents."* (ANAO 2000, p. 66) Fraudulent identities can either be fabricated, or stolen from others. With regards to fabricating new identities, fraudsters can do things such as forge documents (forgeries which now are more convincing of their legitimacy through advances in printing technology) or register false businesses. Examples of how identity information can be stolen include simply looking through trash where disused utility bills and statements may be found (documents which may be used for proof of identity purposes), stealing of proof of identity documents such as passports and driver's licences, 'skimming' (counterfeiting of valid credit cards) (Donnelly 2002) or hacking databases filled with personal data.

Although much identity fraud occurs in the offline environment, the arrival of the Internet has exacerbated the problem. As previously noted, e-commerce systems have increasingly been integrated into organisations, meaning more identity information is being stored and handled electronically. The online world has allowed identity data, now no longer needed to be physically accessible, to be both globally distributed, and globally used, and in the case of identity fraud, globally misused. The US Federal Trade Commission summarised the situation comprehensively at a congressional hearing:

> *"The Internet has dramatically altered the potential occurrence and impact of identity theft. First, the Internet provides access to identifying information through both illicit and legal means. The global publication of identifying details that previously were available only to a select few, increases the potential for misuse of that information. Second, the ability of the identity thief to purchase goods and services from innumerable e-merchants expands the potential harm to the victim through numerous purchases. The explosion of financial services offered on-line, such as mortgages, credit cards, bank accounts and loans, provides a sense of anonymity to those potential identity thieves who would not risk committing identity theft in a face-to-face transaction."* (FTC 2000b)

Hence, as identity fraud has such far ranging repercussions and as it is becoming increasingly common with the growth of the Internet, it is clear that detecting such fraud becomes extremely important. The effects of electronic identity fraud often materialise in the form of payment fraud. The system implementation in this thesis focuses on the process of detecting fraudulent transactions in a payment context, which will now be discussed.

3.2.2 Payment Fraud

When a commercial transaction is undertaken with one party never intending to accurately and truthfully fulfil their part of the deal, payment fraud is committed. For example, if someone intentionally deceives another person by providing invalid or false payment details such that payment can not be collected. Payment fraud ultimately either leaves one party without payment, or the other party without the good or service paid for. On the Internet, this process occurs in many activities, with either the vending party or purchasing party being the victim.

The most common types of payment fraud found online include online auction fraud, general merchandise sales and various e-mail scams such as 'work-at-home' opportunities and Nigerian money offers are all such cases (Internet Fraud Watch 2000; Internet Fraud Watch 2001). Online auction fraud is particularly prominent: Shill bidding, where vendors pose as buyers to drive up bid prices (also known as phantom bidding), and bid-shielding, where two people (or one person with two aliases) collude to prevent other legitimate buyers from bidding, are two forms of fraud affecting auction-based transactions (Duh, Jamal and Sunder 2001). However, a set payment in exchange for goods or services remains the most common type of financial transaction.

The exposure businesses (vendors), which may process large numbers of transactions on a daily basis, have to fraud far exceeds the exposure customers have. Customers can minimise the risk that they will become victims to fraud, and have a certain measure of protection against it. For instance, online credit card purchases are normally insured against fraud (see section 3.2.3). eBay and its recently acquired payment intermediary, PayPal, both provide customers with insurance from auction fraud[1]. Customers can also exercise prudency, and screen vendors before undertaking transactions.

On the other hand, businesses can not pay the same level of attention to the transactions they handle. Businesses often employ automation to process large volumes of transactions. Analysing these transactions manually with human labour would be an incredibly complex and time consuming task, and in many cases, not remotely feasible. It is clear that this deficiency must be addressed, as it is vital that businesses should be able to screen their transactions in an effective, timely manner, in an effort to prevent fraud. The system developed in this thesis focuses on preventing instances of electronic fraud where the business is the victim of the customer, who perpetrates the fraud.

3.2.2.1 How Payment Fraud Occurs Online

Payment fraud commonly occurs on e-commerce systems when a customer attempts to procure goods. A fraudulent transaction is undertaken in exactly the same manner as a normal one, with one exception – payment details provided by the purchasing customer are false or invalid. The most popular and widespread method of payment online is via credit card (Schreft 2002). See figure 1 for a flow of events regarding purchasing from an e-commerce system via credit card. Note that although the exact steps involved in processing electronic payments may differ in various circumstances,

[1] PayPal press release [http://www.paypal.com/html/pr-110300.html] (last accessed: 21 November 2002)

for example, secure transactions using public key infrastructure have been mapped out to fourteen distinct steps (van Krugten and Hoogenboom 2000), the essential flow of information remains the same.

In most cases, all that is needed to purchase goods online is a credit card number and its corresponding expiry date. No signature can be provided, as happens with real world credit card transactions. While most e-commerce systems run a preliminary check against a simple mathematical algorithm (called a ‘mod 10’ check) to see if the credit card number supplied is valid, there are often no further background checks to ensure if the customer is also the cardholder, or the customer is authorised to utilise the card whose card number has been provided, or in some cases, the card has sufficient credit.

Customer		Business		Bank
1. Compiles Order				
2. Submits Order Details	→	3. Processes Order		
		4. Processes Payment	→	5. Verifies Payment Request
				6. Debits Customer Account
9. Receipt of goods/service	←	8. Ships Goods or Performs Service	←	7. Credits Business Account
Transaction complete. If the transaction was fraudulent (the payment details provided were a third party’s credit card number – the use of which was not authorised), the following scenario may occur:				
10. Third party receives statement and detects an unauthorised transaction				
11. Lodges Fraud Complaint with bank	→		→	12. Bank verifies validity of claim
				13. Bank reverses transactions
				14. Credits Customer Account
17. Fraudulent transaction removed from statement	←	16. Writes off loss to fraud	←	15. Debits Business Account

Figure 1: E-Commerce Credit Card Transaction – Flow of Events

The customer submits to the vendor the payment details along with other pertinent information such as shipping address, the actual order details and pricing. These details are passed to the vendor's relevant systems. For example, the order details may be passed to a procurement system. How payment is processed differs depending on the e-commerce system employed. Payment details in highly automated systems are normally passed immediately through to a merchant gateway connected to a financial institution (Marlin 1999; Camtech 1999). In less automated systems, details may be stored in a database for a human to manually process payments at a later time. In the majority of operations, the processing of the order will not proceed until payment is received. However, there are other cases such as where payment processing will not occur until the items are shipped.

The credit card details ultimately end up at the financial institution who handles the relevant transfer of funds. With invalid details, the transaction may be immediately rejected and the vendor alerted. However, with false details, the details are still legitimate, and thus the transfer of funds occurs. These 'false details' consist of the credit card details of an innocent third party (the real cardholder)[2]. Only when the credit card statement arrives for this third party, do things start to go awry. The third party is likely to complain if something they did not purchase, or authorise a purchase for, appears on their statement. As a result, they would contest the transactions in question, and if sufficient proof is provided to the credit card company, then the company can rescind the transaction and reverse the flow of funds (subject to certain conditions, depending on the merchant).

The major problem here is that the company has shipped the goods, but now finds itself unable to collect payment for them (loss of revenue). The innocent third party is also affected in terms of inconvenience and time wasted following up the matter.

[2] Obtaining another person's card number can happen in many ways: theft of a card, overseeing/overhearing card details, intercepting online transactions and extracting payment details from there, hacking into a database, etc. A random card number generator also produces a card number that matches the mathematical algorithm all legitimate card numbers conform to. (The problem with this method, for those intending to commit fraud using it, is that the number may not belong to a real or activated account, causing instant rejection of the transaction by the card merchant.) (Mesmer 2000)

Payments also are charged to accounts. For example, Amazon stores customer credit card numbers on its system. Customers who have shopped with Amazon before log on to the system with a user name and password. Amazon has a patented '1-click' system whereby a logged on user can order an item without having to provide further details (Smith 2001). These 'further details' include the provision of a credit card number, as the system can be told to use a card previously used. Fraud in this circumstance may occur if someone logs on to another person's account, and purchases goods on that person's account without their consent or knowledge.

Fraud may occur with services and both physical and digital goods. Fraud tends to be worse for vendors with digital products, though, as the convenience of being able to deliver a digital product immediately becomes a liability. In many cases, the goods are shipped before actual payment is assured.

3.2.3 The Impact of Online Fraud

Fraud is regarded as one of the foremost and widespread problems in e-commerce. The Gartner Group survey of web retailers shows that internet-based card fraud is *"at least 10 times the rate for the physical world"* (i.e. offline fraud), making it the *"'No. 1 problem' in e-commerce"* (Mesmer 2000).

While credit card companies are affected by fraud (Visa loses 6 cents in the dollar to fraud (Visa 2002)), the main impact of fraud rests against businesses. This is because of how many credit card companies work. Presently, many card companies protect their consumers (by law, in the United States[3]) from fraud by not making them liable for any unauthorised usage of their card, up to the first $50 of the fraudulent transaction. Some companies, like American Express waive this fee entirely[4]. A study by Jupiter Media Metric found that 59 percent of online shoppers were afraid that their credit card details would be stolen (Geralds 2002). This measure promotes online consumer spending, giving a safety net to that hesitant 59 percent of shoppers.

[3] This legislation is found in the Federal *Truth in Lending Act*, Volume 15, United States Code, Section 1601

[4] American Express Online Fraud Protection Guarantee [http://www10.americanexpress.com/sif/cda/page/0,1641,5962,00.asp?CCNR=OZ4] (last accessed: 21 November 2002)

On the other hand, businesses are not so fortunate. Businesses must apply for a merchant account which allows them to accept credit card payments. If a transaction is discovered to be fraudulent, the merchants have to pay for the fraud (the rationale being it was them who accepted the card as a payment method initially). The fee for credit card transactions is also higher if they are processed via the Internet (Kennedy 2000). Credit card companies claim that this is to offset the liabilities and administrative costs of the rates of online fraud being so much higher than that of the physical world. This measure, however, also cancels out the benefit to business of being able to lower costs by being able to process transactions online. Thus, it is also in the interests of business to clamp down on fraud.

Worse yet, if fraud rates get too high for a particular business, the card company can cancel the merchant account. Naturally, the person who committed the fraud is ultimately liable, but given the nature of Internet, it is difficult to track perpetrators down (as well as being potentially costly).

Electronic fraud is common. A KPMG (1999) survey found that, between 1997 and 1999, there was a 71 percent increase in the number of companies who reported computer-related fraud. A domestic joint study by the Victorian Police Force and Deloitte Touche Tohmatsu (Deloitte 1999) run in 1998 showed 33 percent of respondents reporting unauthorised computer usage in the last year, a quarter of which were motivated by financial gain (Anandarajah and Lek 2000).

Electronic fraud is serious. On occasion, it has shut down online businesses (Gengler 2002). Gartner reports that the e-business fraud rate is 1.3-2.6 percent. With the introduction of anti-fraud measures, payment intermediary PayPal has found that it has reduced its fraud rates by half a percent – significant given that it processes over 180,000 transactions totalling US$8 million per day. Expedia, an online travel agent, processes credit card transactions for airline tickets and hotel reservations. It reported that in 2000, card fraud costed them $4.1 million.

The Internet Fraud Complaint Centre is a US organisation that addresses Internet fraud by referring reported cases on to the relevant law enforcement agencies. Through 2001, the IFCC (2001) claims that the 16,775 complaints it referred combined to produce a loss of US$17.8 million. It also found that the average amounts lost by businesses exceeded that of individuals, and that 76 percent of the alleged perpetrators were individuals (as opposed to businesses). Clearly, fraud is widespread and costly. Businesses recognised this too, spending an estimated $6.4 billion on computer security in 1999 (Mertl 2000). A major motivator of this is the possibility of fraud. The Gartner group predicts that *"money spent on private hacker protection will increase from $US720 million in 2000 to $US2.2 billion by 2005"* (Maher 2002, p. 64) thereby showing the importance in business of securing sensitive information such as credit card details.

Ultimately, detecting and preventing fraud is a major issue for businesses undertaking e-commerce. Because e-commerce systems are so tightly interwoven in some businesses, the impacts of fraud are potentially devastating. Therefore, addressing the problem of electronic fraud is also a *strategic* issue that must be managed by both information systems staff, as well as senior management.

3.2.3.1 Impact upon Auditors

The benefits of being able to reduce fraud are clearly evident for business owners. However, auditors also stand to significantly benefit from measures that may detect and prevent fraud, due to their prescribed responsibilities as auditors. Australian Auditing Standard 210 was created:

> *"to establish standards and provide guidance on the auditor's responsibility to consider fraud and error in an audit of a financial report. While this AUS focuses on the auditor's responsibilities with respect to fraud and error, the primary responsibility for the prevention and detection of fraud and error rests with both those charged with governance and the management of an entity."* (AUS210 2002, p. 3)

In the United States, auditors are charged with a similar responsibility in a new Statement on Auditing Standards (SAS) exposure draft released by the AICPA (2002): *Consideration of Fraud in a Financial Audit.* Although primary responsibility lies with business management, the auditing standards above show that internal auditors also play a vital part in detecting fraud, and require effective tools in order to do so:

> *"The problem of misstatements through fraud is neither a rare event nor unique to the Enron case. The bankruptcy of Enron is just one of many examples of how financial shenanigans can quickly escalate and result in the failure of a business and losses to all investors. The auditors in these cases did not use effective tools to discern fraudulent transactions."* (Baer 2002)

It is therefore necessary to determine how fraud can be prevented in order to aid businesses *and* auditors.

3.2.4 Preventing Fraud

Fraud can come from anywhere at any time and, as such, is hard to prevent. Expedia's marketing director, Suzi LeVine observed that, *"fraud was committed by professional criminals who obtained the card numbers, not from Expedia or Expedia customers, but from elsewhere."* (Mesmer 2000) The major issue with fraud is that detection of it is difficult. E-commerce is particularly vulnerable to it because the transactions are remote (and the exact physical location of the other party is often unknown) and the ability to identify the legitimacy of customers is difficult (Ng and Wong 1999).

As a further barrier to the detection of fraud, companies have been reluctant to report on security breaches happening on their systems, even if they are detected in time and prevented from causing damage (Smith 1999). This is due to the fear that their commercial reputation would be damaged, discouraging potential customers. Furthermore, admission of security breaches may attract further attacks due to perpetrators perceiving weak e-commerce system security. This unfortunate view is an impedance to companies moving to implement strategies which may combat fraud. Because fraud attacks are not publicised, the lack of availability of information about the type and frequency of fraud has hampered efforts to deal with it. This only perpetuates the increasingly common incidents of criminals anonymously perpetrating fraudulent activities (Anandarajah and Lek 2000).

In section 3.2.3 it was observed that businesses are often liable for fraudulent transactions. A merchant can prosecute a criminal to recover the liability, but this rarely happens. *"Less than 10 percent of the cases are prosecuted to the point where a* *merchant received restitution. A higher percentage is prosecuted, and conviction of the perpetrator may even occur, but unless the merchant received restitution, it isn't really 'successful'."* (MacVittie 2002) Thus, correcting the crime after it has occurred is not an attractive option for businesses. The other measure left is detection and the subsequent prevention of fraud *while it occurs*: the timely examination of e-commerce transactions for fraud. The next section addresses how fraud may be detected.

3.2.5 Fraud Detection Methods

MacVittie (2002) offers some general, stopgap measures for organisations to cut down on e-commerce fraud:

- Use an Address Verification System (AVS). An AVS cross checks a customer's billing address with the address connected with a credit card number to ensure validity.
- Verify e-mail addresses. As 97.3 percent of fraudulent orders comes from free e-mail accounts (like Hotmail), some companies choose to disallow such accounts.
- Monitor discrepancies between billing and shipping addresses. If the two addresses are international relative to each other, the transaction may be suspicious.

In many cases, more elaborate solutions are needed and many companies have implemented their own in-house systems to deal with fraud. Various industry alliances, notably amongst credit card companies, have been formed to produce schemes to reduce fraud as well. Each approach is different from the other, utilising different techniques and tools.

Virtually all e-commerce systems today employ SSL (secure socket layer) technology, a de facto standard which ensures that data communicated between customer and business is encrypted (Comer 2000). Although this protects customer card details from being intercepted by third parties, it does not perform any function that allows business to detect fraud. Nonetheless, it is an indirect measure that reduces the availability of credit details that fraud perpetrators may utilise to engage in fraud. The main benefit of SSL is that it is universally employed, something not true of the methods below.

Certain schemes aim at ensuring that the identity of each party in an e-commerce transaction (the cardholder, business and credit card company) is authenticated, thereby ensuring for the business that the payment details are coming from someone who authorised them. Visa and MasterCard developed SET (Secure Electronic Transactions), a specification involving the use of digital signatures and public key encryption as a means of identity authentication[5]. Unfortunately, the uptake of SET has not been widespread and Clarke (1996) points out various weaknesses in the scheme. Among these are complexity and the need for many participants to adhere to the specification, that nothing is mentioned about managing participants' private keys, and that nothing is said about the apportionment of responsibility for losses. A customer's private keys could be stolen, meaning that a fraud perpetrator may thwart the reliable authentication SET is meant to provide. If credit card details can be obtained, so too could SET private keys.

Visa's 'Verified by Visa' scheme provides assurance to merchants by adding on a plug-in to a merchant's payment processor. The plug-in adds a layer of security in which the user is prompted for a password as an added step to authenticate his or her identity. Unfortunately, this suffers the same weakness of SET in that if the password is compromised, then the verification process becomes inaccurate, opening the door for fraud. Furthermore, utilising SET and 'Verified by Visa' also slows down the time to process credit card transactions – the latter *"on average…adds 10-20 seconds to the total transaction time"* (Visa 2002). This negatively affects the customer's purchasing experience which is influenced by expediency (Elliot and Fowell 2000).

Another approach is one that is deployed solely on the business' side, therefore not relying on the customer to do anything special. These approaches use rule-based expert systems that compare each transaction with a predefined set of rules before approving, denying, or flagging the transaction for manual review. The rule set used by the system must be determined by the business according to the context of the e-

[5] SET LLC, How Set Works [http://www.setco.org/how_set_works.html] (last accessed: 10 October, 2002)

commerce system such that anomalous transactions are detected (Gengler 2002; MacVittie 2002).

A further enhancement to a rule-based expert system is a neural network. These systems are also called predictive statistical modellers or fraud scorers (MacVittie 2002). They employ statistical modelling and data mining techniques on accurate historical data to assess current transaction patterns and determine if a transaction is likely to be fraudulent. While neural networks have been known to be effective, they are complex to set up and rely on a large, accurate source of historical transaction data which may not always be available to a company. Neural networks also require customisation, as an uncustomised system will flag too many transactions as fraudulent. Additionally, neural networks do not easily adapt to systems which change a lot (as may be the case with e-commerce systems), as new patterns have to be programmed back into the system.

Visa and CyberSource's Internet Fraud Screen utilises a hybrid expert system and neural net model to analyse each transaction processed by a business' web site (MacVittie 2002). Each transaction is assigned a score that ranks how suspicious it is based on data validation, artificial intelligence pattern matching, network data aggregation and negative file checks. The Fraud Screen may then block a transaction from progressing if it hits a certain threshold (InternetNews 1999).

Many other systems employ neural network related techniques to discover fraudulent transactions on their system. ClearCommerce Corp.[6] and Fair Isaac[7] offer neural-network-based fraud-detection systems. Pure Commerce offers a fraud detection solution that uses a mishmash of methods including: *"neural networks, feature detectors, pattern transaction orientation, non-linear relationships and rule integration."* (Pure Commerce 2002) While these fraud detection solutions exist, some companies have found it more effective to develop their own, proprietary system. PayPal, which provides a payment systems solution for business and individuals, being one example (Gengler 2000).

[6] ClearCommerce Corp. [http://www.clearcommerce.com/] (last accessed: 22 November 2002)

[7] Fair Isaac [http://www.fairisaac.com/page.cfm/section=sub_cat/id=389/id3=389/id1=46/id2=157] (last accessed: 22 November 2002)

Although a myriad of schemes, ready-made systems and proprietary solutions exist, this says nothing about the effectiveness of them in detecting and preventing fraud. eBay has software which statistically analyses bidding records in order to detect the presence of shill bidding. However, it has been criticised that eBay's solution is very limited and thus ineffective due to eBay only keeping limited bidding records, extending back one month.

> *"In addition, some [sic] insider of eBay has indicated that eBay was not willing to spend substantially on advanced data mining software. Even eBay itself admits that its screening system is not effective enough to detect all shill bids and bidding rings. Detection is hard because shill bidders try to remain undetected and hence anonymous. The Internet makes the hiding of true identities a much easier job and consequently finding one's true identity an extremely difficult task."* (Wang, Hidvégi and Whinston 2001b, p. 6)

Furthermore, an MSNBC investigation found that eBay's lack of transparency with revealing how it deals with fraud was cause for much consumer concern (Brunker 2002).

Companies have staged numerous efforts, all using different techniques, in order to alleviate the problem of fraud. However, the introduction of all these has done little to lift customer and business confidence. These efforts occur independently, producing proprietary solutions, so overall, the industry response to fraud detection has been extremely fragmented. How and how well these solutions work is little publicised, due to the fact that these systems offer businesses a competitive advantage over other businesses without them. Unfortunately, this also means that the effectiveness of current fraud detection solutions is questionable. That none of these systems or methods have been utilised on a wide scale reflects upon these facts, and a collaborative, widespread framework by which fraud detection can be implemented is yet to be developed. Organisations require a systematic method of detecting fraud which can be easily integrated into an e-commerce system, providing an effective, flexible and maintainable mechanism by which transaction validity may be assured. In the next section, we will look at Continuous Assurance, a field of research that is aimed at allowing the timely detection and prevention of anomalies in data. Many of the above techniques and systems discussed fall under this general area, and understanding continuous assurance will help establish a framework by which fraud detection can be systematically implemented and then integrated into e-commerce systems, independent of the nature of the transaction (eg: payment method chosen) and maintainable by the business.

3.3 Continuous Assurance

Continuous assurance[8] is a rapidly developing field of research which extends upon the traditional accounting practice of auditing. A report by the American Institute of Chartered Public Accountants (AICPA) and the Canadian Institute of Chartered Accountants (CICA) defined continuous auditing as:

> *"a methodology that enables independent auditors to provide written assurance on a subject matter, for which an entity's management is responsible, using a series of auditors' reports issued virtually simultaneously with, or a short period of time after, the occurrence of events underlying the subject matter"* (CICA 1999, p. xiii)

In other words, continuous auditing is a type of auditing that produces audit results either instantaneously, or a very short time after the occurrence of the events being audited. The terms continuous auditing and continuous assurance are sometimes freely interchanged in the literature, but a distinction between them does exist, a view shared by Alles, Kogan and Vasarhelyi (2002, p. 126).

This distinction lies behind the nature of continuous assurance and how it differs from traditional auditing. Traditional auditing concerns itself with reporting on the relevance and reliability of financial and accounting information on a regular, but long term, basis. It has long been realised that any type of information can be audited. However, due to the time lags inherent in traditional auditing techniques (traditional auditing occurs *ex post*), it has not been practical to audit all types of information, hence the confinement to financial information (CICA 1999).

[8] The term "concurrent auditing" is also used, in reference to when auditing occurs concurrently with the system's operation.

However, because continuous auditing has the definitional characteristic of being able to immediately produce audit reports on information (*ex ante*), any type of information can now be practically audited, not just financial data. This is useful because organisation decision-makers often employ both financial and non-financial information in their bid to reduce uncertainty in decision making (Daigle and Lampe 2000). As opposed to financial information, non-financial information is often not attested by auditors, even though it may be just as relevant to the decision making process. Non-financial information, which may be of a more time critical nature and which may not occur in regular time intervals as financial information, can now be audited and reported on.

This vital difference expands the scope of continuous auditing away from the traditional role of guarding capital markets to virtually all markets (Wang, Hidvégi and Whinston 2001). In an attempt to account for this audit technique being more encompassing than traditional auditing, the term 'continuous *assurance*' is used when auditing data which may be of a non-financial nature. Continuous audit techniques may be applied to any form of information to provide assurance and trust to information users on its correctness and relevance.

Continuous assurance could thus apply to assuring e-commerce transactions. Indeed, Shields (1998, p. 40) suggests that continuous assurance may be able to aid guaranteeing *"the authenticity, integrity and non-repudiation of electronic commerce transactions"*.

3.3.1 Requirements for Continuous Assurance

Continuous assurance is not a new idea. In fact, as early as 1980, John Kearns wrote an article asking *"Are we ready for continuous process auditing?"* (Shields 1998, p. 39) Clearly, there is a set of requisites that must be met before continuous assurance may be successfully and feasibly employed. These requisites have been identified as:

1. identifying the subject matter to be audited and ensuring it has suitable characteristics
2. having reliable systems which provide the subject data

3. having highly automated procedures which provide the audit evidence
4. having a reliable means of obtaining instant or timely access to audit results
5. the auditor being knowledgeable about the subject matter as well as on the information system providing the continuous assurance (CICA 1999; Vasarhelyi, Kogan and Sudit 2000)

Naturally, these requirements present a variety of technical hurdles. The first of these is that the information to be assured is generated by systems that are reliable. With continuous assurance, there is much less time available for errors to be dealt with, and errors that occur with any frequency would greatly diminish the advantages continuous assurance offers. Furthermore, these systems may be complex and contain multiple sub-systems, thus any components in a system must be properly integrated with networking infrastructure and sharing of common data (Shields 1998).

A high degree of automation is required in order for continuous assurance to deliver results on a timely basis. Automating processes would ensure that minimal time elapses between the occurrence of the events being audited, and the audit reporting that takes place on those events. This also implies that assurance systems would have to be tightly integrated with the systems being audited because of the level of communication between them. As a part of ensuring tight integration, adequate network connectivity between the two systems is important as it allows the automated processes to be carried out quickly, and also allows auditors to obtain audit results on a timely basis.

Finally, the tightness of integration between the systems mean that auditors now must be proficient in both the subject matter (which may be financial or non-financial) and the systems associated with processing and auditing it. Hence, the importance of information systems auditors within the audit team.

The prominence of information systems as used in business has grown immensely throughout the last decade. Vasarhelyi, Kogan and Sudit assert that *"accounting information is now almost always recorded and stored in electronic form"* (Vasarhelyi, Kogan and Sudit 2000). Indeed, much commercial activity today is carried out over electronic systems. The increasing introduction of these information systems brings a greater dependency on them, as they progressively process greater and greater volumes of data (Daigle and Lampe 2000). Management has also recognised that the implementation of stringent internal controls must occur to mitigate the risks associated with this increased dependency. Continuous assurance systems provide an appealing risk management mechanism. In fact, *"the advantages of electronic business reporting will provide a market for – indeed, the necessity of – continuous assurance."* (Elliott 2002, p. 141)

E-commerce systems especially form major supply and distribution channels for many organisations (and between organisations), processing large quantities of transactions on a daily basis. That these information systems have begun to assume such vital roles in business also attests to their reliability, and as such, imply that the requisites 2 and 3 above are able to be met by today's business environment. Network connectivity, especially via the Internet, provides accessibility to these systems without the need for physical presence (requisite 4). Not only is the infrastructure available, but declining costs of hardware, software and network connectivity also facilitate the uptake of technology necessary to implement an assurance system.

Elliot (2002) notes another requisite for continuous assurance, that information should be disseminated in compliance with standardised information formats that facilitate the acquisition and analysis of that information. Progress is being made in this field, notably by XBRL (eXtensible Business Reporting Language) which is an initiative of the AICPA. XBRL assists the preparation, exchange, publication, acquisition and analysation of all types of data (Elliot 2002).

In addition to these requisites, further barriers do exist that impede the uptake of continuous assurance. The first is that continuous assurance systems place additional burden upon operational systems. This may impact negatively upon system performance. Continuous assurance systems themselves will also handle a significant amount of data, and thus it is also important that capacity planning occurs for them as well (Rezaee, Sharbatoghlie, Elam and McMickle 2002). However, it is felt this has been partially catered for by a large increase in computing power that is also affordable. A further barrier is that continuous assurance is a complex process that requires a great deal of systems expertise, resulting in high costs associated with implementing such a system. This issue is one that will have to be considered on a case by case basis, whether the gains to be made from continuous assurance can offset the cost of resources invested into continuous assurance systems development and maintenance.

Therefore, it would seem that the time and conditions are right for continuous assurance to be introduced. E-commerce systems satisfy the requirements that must be met for assurance to occur continuously. A growing demand exists in business for continuous assurance (Alles, Kogan and Vasarhelyi 2002), especially as providing assurance is akin to adding value to any transaction. The AICPA's Special Committee on Assurance Services holds the opinion that in future, there will be a need for assurance in any generic exchange of goods or services. Indeed, assurance systems exist today such as those jointly developed by the AICPA and CICA – SysTrust[9] and WebTrust[10]. These two systems are not currently of a continuous nature, but Elliot (2002, p. 142) notes that, *"We are likely to arrive at continuous assurance as these services progress further along the spectrum from periodic, after-the-fact assurance to continuous assurance."*

Continuous assurance systems certainly have a place in today's digital economy and are feasible. This motivates the need to develop actual assurance systems themselves, that is, the auditing systems that monitor operational systems currently in place, ensuring they are flexible, maintainable, and tightly integrated with the operational system. Due to the importance of e-commerce transactions, the subject matter for this thesis has been identified as payment transactions which occur over e-commerce systems.

3.3.2 The Benefits of Continuous Assurance

As already mentioned, continuous assurance offers business the ability to verify all types of information, both financial and non-financial. This gives decision-makers increased trust in the information they use, reduces uncertainty and increases transparency in reporting:

[9] SysTrust assures the security, integrity, availability and maintainability of information systems.
[10] WebTrust assures the security, availability, business practices and transaction integrity of e-commerce web sites.

> *"...a motive for fraudulent financial reporting would be removed by real-time disclosure. The attorney Michael Young held that frauds caused by attempts to match analysts' quarterly earnings estimates would be eliminated."* (Elliott 2001, p. 1)

Perhaps the collapses of Enron and WorldCom may have been prevented if reporting had been more transparent (Baer 2002). In any event, continuous assurance should allow the more timely detection of and response to anomalies in financial transactions and reports, picking up the occurrence of fraud before it festers.

Furthermore, the nature of information systems can be leveraged through the fact that continuous remote access is available for auditors (either through the Internet or other methods of connectivity). Assurance can be provided from afar as auditors are not required to have a physical presence on the site of the system being audited, saving on the time and cost of travel.

The act of businesses transitioning to storing information digitally has also changed the audit trail – the documents created and recorded as transactions exist from initiation to final posting. The elimination of the 'paper trail' has meant that traditional audit methods can no longer be employed in a paperless environment. Kanter (2001) postulates that *"when an audit trail does not exist or the auditor does not want to rely on the audit trail that exists within the application system to be included in the audit, concurrent auditing techniques may be used."* Because all data is also recorded in electronic form, the need for manual procedures is eliminated, thereby producing the attribute of timeliness associated with continuous assurance. In fact, continuous assurance is the only option for some systems which are too complex to be audited by traditional means (Weber 1999, p. 755).

Continuous assurance has a lot to offer in terms of detecting fraud. As online e-commerce transactions occur in real-time, so too can these transactions be assured in real-time. This real-time nature is significant because the timely detection of fraud is important – it is easier to prevent fraud if it is detected immediately, rather than later. Moreover, the presence of internal controls within a system not only provides the facility to detect fraud, but also helps to deter it (Glover and Schleifer 1995). Section

3.2 discussed the impact of fraud and the having a tool to reduce this would be invaluable for businesses.

3.3.3 Implementing Continuous Assurance

An assurance system traditionally provides three services:

- Capturing information related to the transactions being assured.
- Analysing the captured information (the assurance process).
- Communicating the outcome of this process to the assuror.
  (Alles, Kogan and Vasarhelyi 2002)

A continuous assurance system must perform the same services, but in a more timely manner. Therefore, a continuous assurance system requires new techniques by which to provide timely assurance. So, moving on from the aims and feasibility of continuous assurance, this section presents an overview of techniques with which continuous assurance may be implemented. This field is a rapidly growing one, especially as, *"Ever improving technology suggests that the real-time exchange of financial data will place constant pressure on auditors to update audit techniques."* (Rezaee et al. 2002, p. 147)

3.3.3.1 Computer Assisted Auditing Techniques

There are three general approaches to performing computer-based audits which have traditionally applied to auditing computerised accounting systems. They form the basis of how a computerised system will approach the task of auditing another computerised system. Although not directly related to continuous assurance, looking at these will help us see the types of computer assisted auditing techniques (CAATs) are in use, and where continuous assurance fits into the scheme of things. These CAATs are: auditing around, with and through the computer.

Auditing Around the Computer

This audit approach is a 'black box' approach which asserts that if, given a set of inputs, the outputs from a system are accurate, then it may be concluded that the internal operations of the system are correct (Garceau 1998). As this approach does not directly concern itself with any internal processes occurring, this form of auditing is simple to implement. Little technical knowledge is required, and as the client system is not modified, corrupting it is not a risk. However, the assurance this method provides of system reliability is only limited – this technique is good for familiarising an auditor with a system, and is often used in conjunction with another method (Gay and Simnett 2000).

Auditing With the Computer

As noted in section 3.3.1, most business data today is stored on electronic media and thus requires a computer to access it. Especially if the data kept is voluminous (which is not uncommon given the growing ease and cost effectiveness of electronic data storage), it may be beneficial for the auditor to use a computer to automate some of the auditing procedures (Ng and Wong 1999).

Normally auditing with the computer involves the auditor obtaining a copy of client transaction or master files, and then passing them to the computerised audit system. The audit system may be classified as generalised audit software, specialised audit software or even a utility program designed to perform a routine such as sort, summarise or perform statistical analysis on data (Gay and Simnett 2000).

Auditing Through the Computer

Auditing through the computer involves testing a client's program – both the programmed accounting procedures which calculate and summarise data, and the programmed control procedures which authorise data changes and validate that they are complete and accurate (Gay and Simnett 2000). Auditing through the computer techniques aim to test the controls of a system and are well suited to providing assurance for complex systems that involve real-time, automated data processing. This fact supports this audit technique's applicability to e-commerce systems which have this quality of handling large numbers of real-time transactions.

Methods for implementing continuous assurance fall under this category of CAATs. The following section explores these various methods.

3.3.3.2 Continuous Assurance Methods

Continuous assurance may be implemented using a variety of methods, but Mohrweis (1988) classifies these methods into three main categories:

- Those that assure systems by using test data while systems undertake production processing.
- Those that selectively take transactions from systems undertaking production processing, and auditing those.
- Those that trace or map state changes in application systems as they undertake production processing.

Although there are many specific methods that have been developed, we will explore only a few of the major ones including the Integrated Test Facility (ITF), Snapshot/extended records, Continuous and intermittent simulation (CIS), audit hooks parallel processing and the System Control Audit Review File (SCARF). These methods all provide continuous assurance by monitoring transactions as they pass through systems in real time.

Integrated Test Facility

The ITF method involves introducing dummy entities into the system. Test data is then processed against these entities to verify processing authenticity, accuracy and comprehensiveness. The two main issues in implementing the ITF method lie in how the test data will be entered into the system. The other issue is how these entities may be removed, or at least disabled such that they do not affect the results of live, legitimate production data on the system (Weber 1999).

Test data can be processed against an ITF entity using two methods. The first is tagging incoming transactions so that in addition to being processed against real entities, they are processed against the dummy ones as well. This tagging process can be achieved in various ways – by manually tagging source transactions, embedding audit software modules which automatically tag transactions matching certain criteria, or using sampling routines which tag transactions according to a sampling plan.

The second method involves creating a series of test transactions and entering them into the system along with the real transactions. This allows auditors to comprehensively tailor tests to test all the possibilities. Naturally, development of this test data is more time consuming than the former method.

Finally, as both ways of entering test data above modify the system in different ways, care must be taken to ensure that these IFT entities or test transactions can be removed or reversed. Documentation of testing procedures is therefore essential in ITF tests.

Snapshot and Extended Records

The snapshot method takes a 'snapshot' of a transaction at an instant in time, at points of the system where processing of data occurs (especially where the data may undergo transformation). A before and after image is captured, and assurance occurs by comparing the two images and seeing if the transformation that occurred at that processing checkpoint is correct and complete.

With this method, three decisions must be considered. Firstly, deciding where to locate the snapshot points is important as taking snapshots increases the burden on systems and thus impacts upon system performance. Enough snapshot points must be taken to audit the system, but at the same time, too many should not be used. Secondly, deciding which transactions to track at each snapshot may be done via the tagging methods similar to ITF. Lastly, a reporting system must be defined so that auditors can extract meaningful information out of the snapshots taken – for example, at the minimum, snapshots should be timestamped, as well as recording at which processing point the snapshot was taken.

An extended record is merely an extension on the idea of snapshot records. Instead of one record per snapshot, an extended record progressively builds the various snapshots into one record. The advantage of this is that all the data captured is stored in one location (Weber 1999).

Parallel Processing

Parallel processing is also known as parallel simulation (Simnett and Gay 2000, p. 539). In this method, an auditor's own program is used to process input data that is also processed by the audited system. The two sets of results are then compared to see if there are any differences in what should be identical outcomes. This method is more commonly used to calculate key results than duplicate all system calculations, and test more the processing accuracy of systems as opposed to testing programmed control procedures.

Audit Hooks

An audit hook is an exit point placed in a system that allows auditors to place in commands for special processing. An example is placing an audit hook in a DBMS that allows the auditor to insert additional coding which passes the transaction to a parallel processing system. This permits the auditor to obtain, for example, independent control totals as a result of normal processing (Simnett and Gay 2000).

System Control Audit Review File

The SCARF method involves embedding software audit modules within a system that function to continuously monitor transactions. These audit modules are inserted at various key points in the system and collect data which is outputted to a SCARF audit file. Examination of this audit file (and reports generated from it) is the responsibility of the auditor, who can then follow up any discrepancies that arise.

The routines in the embedded modules contain code which monitors the transactions and code which dictates what actions the module will take in response to different types of transactions. These routines may get quite complex, and as they are also crucial to the operation of the audit system, must be protected. It is advisable to separate the code of the embedded modules completely from the system being audited. The system should only contain extra code that calls these modules – not any of the code for the modules themselves. Hence, the modules may be maintained and managed separately from the application system. Combining SCARF with the audit hooks method facilitates the calling of external modules from certain points within the application system.

With SCARF, there are three primary considerations: determining how the SCARF file will be updated, defining the format of reports to be produced and choosing when the reports will be generated (eg: on demand, or on a scheduled basis).

The method by which SCARF files are updated becomes important when there are multiple systems to audit. For example, transactions submitted to a load balanced server array may end up being processed on any individual server. Embedded audit modules must reside on each machine, but when logging to the SCARF file, several systems attempting to access that file simultaneously may cause problems. One option is to get each system to write to its own temporary SCARF file, before consolidating the temporary files into one master file when permitted. The disadvantage to this solution, however, is data fragmentation. Weber (1999, p. 766) explains that *"data fragmentation might undermine SCARF's effectiveness as an audit technique because it prevents timely analysis of SCARF data across application systems."* Ultimately, the most effective solution may be to employ a database management system (DBMS) to manage the SCARF file. The DBMS takes care of all transaction integrity and concurrent access problems that may otherwise arise.

The reports which are generated from the SCARF file are important as it is these reports that auditors examine. SCARF data must be properly sorted and formatted in order to draw meaning from them and facilitate the interpretation of results. Obfuscated reports may cause auditors to miss errors and irregularities, so it is important that reports display all the information relevant to auditors.

As SCARF operates in real time, reports can be generated instantly. However, when they are generated, and the method by which they are generated, depends upon the costs of report generation and the urgency of the report. In auditing e-commerce credit card transactions, for instance, a SCARF module may flag a severe exception indicating high possibility of fraud. In this case, the system may be programmed to generate an immediate alert to the auditors, notifying that a potential fraudulent transaction has just been submitted. The alert may be in the form of an e-mail, or a 'red flag' on a web page which the auditor periodically checks. Even wireless solutions, such as SMS notifications are possible. For transactions that are less suspect, the reporting action could be less conspicuous. This allows auditors to be overtly notified of severe exceptions. Again, the costs (such as increased burden on systems) must be weighed with the benefits (such as being able to immediately respond to fraud warnings) when deciding what strategy to adopt.

A way of looking at the SCARF audit method is in terms of software agents. A software agent is an autonomous actor (for example, a module) that resides in a system with sensors and is able to perform actions. Nehmer (2000) provides various examples of agents such as process, access, control and transaction agents. SCARF would be considered a transaction agent, using embedded audit modules to track transactions passing through the points where the module is hooked into the audited system. This transaction agent also has the ability to respond to these transactions, for example either logging them or blocking them completely.

Continuous and Intermittent Simulation (CIS)

CIS was proposed by Koch (1981, 1984) and is a variation on SCARF. CIS applies to systems employing a DBMS. CIS operation is similar to SCARF, except that it uses the DBMS to trap exceptions, instead of audit modules. That is, the act of transaction monitoring takes place at the database level, not the application level. Thus, while SCARF logs application processing procedures, CIS logs accesses to the DBMS. Otherwise, the premise is virtually the same as SCARF.

3.3.3.3 Methods for a Fraud Audit Strategy

Section 3.3.3.2 discussed the various ways of integrating continuous assurance into a production system, such as an e-commerce system. How though, can this system be used to detect fraud? E-commerce systems may process enormous amounts of transactional data, not all of which is relevant in fraud detection. An obstacle to a continuous assurance system that detects anomalies which may signal fraud is that it must also be able to filter out audit data which is extraneous. Once that has occurred, it must then determine which of the transactions that pass through it are relevant to be further screened. Thus, fraud auditing continuous assurance systems must be designed such that superfluous attributes of a transaction can be discarded from the review, and that legitimate transactions are left alone. Thus, several methods of a fraud audit strategy may be used separately or in combination with each other, depending on the type of transactions being processed. The technical implementations of these methods is beyond the scope of this thesis. Only what is available to condense and filter transactions captured by one of the continuous auditing methods that were covered in section 3.3.3.2 will be very briefly discussed.

Data mining is a very broad and active field of research that is aimed at data analysis. Data mining is defined as the non-trivial extraction of implicit, previously unknown and potentially useful information from data in a database (Chen, Han and Yu 1996). This 'useful information' may consist of knowledge rules, constraints or patterns. Data mining can be employed for a wide variety of different purposes, and fraud detection is one such area (Groth 2000).

Data mining relies very much upon the algorithms used to analyse data sets. In general, there are three categories of algorithms: data association rule algorithms, sequential analysis, and classification rule-learning approaches (Anandarajah and Lek 2000). Data association rules are rules which attempt to describe relationships of differing strengths that exist between the occurrence of certain events and other related ones. Sequential analysis is often used to detect things such as sales trends as sequential analysis' goal is to discover ordered data sequences, such that these sequences may be pre-emptively forecasted in the future. However, it is classification rule-learning approaches that are particularly interesting with regards to fraud detection. Algorithms utilising this approach attempt to classify new data, based on a set of grouping rules which have been derived from analysis (by human or machine) of existing data.

Classification rule-based algorithms rely heavily on statistical analysis and machine learning, and Anandarajah and Lek (2000, p. 46) further note that, *"The run-time efficiency of these systems is of critical importance if the knowledge learnt is to be put to any practical purpose,"* implying that a system should run in real time to be of most benefit, which is an attribute of continuous assurance systems. Furthermore, classification rules are highly suitable for tasks where events are required to be grouped into different classes. Electronic fraud detection requires the classification of e-commerce transactions into different levels of fraudulent risk. Implementations of this rule-based algorithm include artificial neural networks, statistical pattern recognition and ripple down rules.

It should be noted that different implementations of algorithms can be combined and applied to a data set to provide a more thorough analysis. Indeed, as noted in section 3.2.4's references to expert systems currently in use in industry, these systems may be composed of several analytical techniques.

Regardless of the method used, in the context of fraud detection, the aim is to detect transactions that may be fraudulent, based upon unusual patterns that may be derived from past data by a neural network, or even from rules manually entered by an auditor. Examples of such patterns may be:

- strange spending patterns such as transactions near a credit card's spending limit;
- strange combinations of products ordered; and
- realising certain products as being more susceptible to fraud (such as intangible products that may be digitally delivered).

Statistical Pattern Recognition

Pattern recognition can be defined as distinguishing patterns of interest from their background and then making judgements about classifying that pattern. Statistical pattern recognition uses statistical methods to achieve this goal, performing analyses in several stages: data acquisition and preprocessing (to get the data into an analysable format), data representation, and finally decision making where the data is classified or categorised. Statistical pattern recognition is useful for fraud detection because large amounts of transactions can be statistically analysed, where it would be impossible for a human to manually perform the same task.

Artificial Neural Networks

A neural network is an information system that recognises patterns based on a set of examples used to train it (Alter 1999). An initial 'learning phase' is used to train the neural network in future attempts to classify data. For certain purposes, neural networks are especially effective, most notably interpreting real world sensory data (biometric identification, for instance). A major requirement for neural networks is having a database from which the neural network can learn. This does not make it suitable for all circumstances in which fraud detection is desired. Nonetheless, groups such as the Internet Fraud Control Consortium (IFCC) have created databases of credit card transactions which can be used to build up and train neural networks. Neural networks can be considered as a special type of learning statistical pattern recognition system.

Ripple Down Rules

Ripple down rules are a formalised structure of rules that are used as a knowledge representation of a method (Delzoppo, Mulholland and Hibber 1993). This structure is composed of a binary tree of nodes, where each node represents an atomic test which can be evaluated as true or false. Nodes are linked by true and false branches. Data enters at the 'top' of this tree and traverses downwards through the nodes, with each test undertaken reducing ambiguity and bringing the data closer to being classified. When the bottom of the tree is reached (when a node does not have any branches to 'lower level' nodes), the conclusion at this final node is returned.

Ripple down rules structure classification rules in such a way that new rules can be added in manually with ease. Ng and Wong (1999) note how one ripple down rule system allowed classification rules to be added into a knowledge base at up to forty times faster than the older system. Existing rules are also easily readable by humans as ripple down rules are conducive to being graphically represented (such as in eSCARF).

An additional benefit of ripple down rules is that, whereas neural networks and statistical pattern recognisers are not easily modified, rules can be manually and straightforwardly configured by humans who need no expertise in knowledge engineering (Kang, Preston and Compton 1998). For example, eSCARF provides a rule designer interface which facilitates this process. An auditor may implement rules to manually target common patterns of fraud, without knowledge of the complex algorithms employed by other audit techniques.

3.4 Conclusions

The review of the literature has addressed RQ1 and RQ2 by examining the two areas of electronic fraud and fraud detection, and continuous assurance. The literature review established the context of this research – the e-commerce environment. Despite its prodigious growth and the numerous advantages it offers to businesses and consumers alike, it is not without its risks and weaknesses. It is important that these risks are identified and addressed, such that participants in e-commerce may be more confident when engaging in it. This confidence in turn translates to increased adoption and usage of e-commerce systems.

The concept of electronic fraud as a risk to businesses using e-commerce has been identified and detailed, including how payment fraud occurs, and the negative effects it has upon businesses. RQ1 was addressed by detailing the various methods for preventing and detecting fraud (eg: AVSes, SET, SSL, neural networks, continuous assurance, etc.). Despite these existence of these methods, it was recognised that the effectiveness of them is unestablished in academic research due to the primarily private and proprietary nature of such fraud prevention and detection systems.

Continuous assurance, a concept which relates to assuring data in a timely manner, was introduced as a means of controlling the risk of electronic fraud. An overview of continuous assurance detailed how the requisites for it (such as the necessity for highly automated procedures which provide audit assurance, and the necessity for the system being assured to be reliable) were met by the current e-commerce environment. Continuous assurance was then related back to e-commerce fraud auditing, discovering that it was particularly applicable as a fraud auditing method, thereby answering RQ2. Various specific methods of implementing continuous assurance were discussed, including ITF, snapshot and extended records, audit hooks and SCARF. SCARF was given closer examination because it is the method used in this thesis' implementation of a continuous assurance system. The SCARF method embeds audit modules (hooks) within a system to monitor in real-time the transactions flowing through it. Fraud is detected in these transactions by applying a fraud audit strategy. Such a strategy consists of comparing transactions against specially tailored rules and algorithms which attempt to discern any suspicious activity that may indicate the presence of fraud. The discussion of continuous assurance concluded with a brief overview of such strategies, including artificial neural networks and ripple down rules.

Chapter 4. Conceptual Model Development

The development of a conceptual model helps to piece together the concepts covered in chapter 3, and further addresses RQ1 and RQ2 by providing a holistic view of detecting electronic fraud, and how continuous assurance may be suitably adapted for such a purpose.

4.1 Motivation

Companies have staged numerous efforts, using many different techniques, in order to alleviate the problem of fraud. However, these efforts occur independently, so overall, the industry response to fraud detection has been fragmented. Partially, this is because of the need to maintain competitive advantage. A company that has an effective fraud detection system has a competitive advantage over others that do not, due to increases in business online, as well as cutting losses sustained from fraud. The commercial interests of these businesses protect their proprietary systems from competitors. Unfortunately, this means that industry is also incapable of developing a secure fraud detection framework as many of the existing solutions are unassessed and their effectiveness unverified. Without collaboration between organisations, researchers lack the opportunity to work on larger datasets from companies across the spectrum which hinders efforts to optimise detection methods.

A more generalised and non-vendor specific approach is required, with the development of a common framework or model that may be employed and adapted to suit by different companies, regardless of industry or size. Towards this goal, an understanding of the issues and concepts surrounding fraud auditing of e-commerce systems using continuous assurance will help in the development of effective responses to electronic fraud. This holistic, unified, integrated approach, should allow the future development of responsive, robust and measurably more effective systems that combat fraud.

An added advantage of a generalised model is that software that is developed based on the model may also be of a generic nature. Continuous assurance systems require a great deal of expertise and time in order to customise them for specific e-commerce systems. A generic 'template' system is much easier to customise than one that is already specialised in its creation. Although expert involvement in customisation and maintenance (to adapt to the inevitable changes in the business environment) will always be required for systems of this nature and complexity, a generic model will make these processes easier.

Generic systems are also flexible and should be able to be integrated into a diverse range of systems. The flexibility allows companies to tailor solutions to meet their needs and budgets, as these systems are costly to implement. It also allows the widescale deployment of a uniform, proven approach to fraud auditing.

4.2 Activity Theory

"Activity theory is a powerful and clarifying descriptive tool rather than a strongly predictive theory." (Nardi 2002) Activity Theory is a social theory designed as an approach to understanding human activity and interaction. Activity cannot exist as an isolated entity. The very concept of activity implies that there is an entity that acts (either an individual or collective, termed the 'subject'). A subject, who is engaged in activity, has its attention directed at an object (thing), interacting with other things along the way. Activity Theory postulates that activity mediates interaction between subject and object (Bannon 1997). Current Activity Theory's basic principles include object-orientedness (activity is composed of a set of objects), the dual concepts of internalisation/externalisation (mental processes are derived from external actions through the process of internalisation), tool mediation, the hierarchical structure of activity, and continuous development (activities evolve).

This school of thought was founded by Russian Lev Vygotsky in collaboration with his colleagues Luria and Leont'ev. Vygotsky introduced the concept of artifact-mediated and object-oriented action (Vygotsky, 1978, p. 40). A human individual never reacts directly to the environment, but instead the interaction between human and the objects of the environment is mediated by cultural means, tools and signs (the mediating artifacts). Leont'ev expanded upon Vygotsky's model, by introducing human beings and social relations as two other mediating forces (Engeström 1998). This produced Leont'ev's three-level hierarchical model of activity which makes the distinction between individual *action* and collective *activity*. At the bottom level of Leont'ev's model are the automatic operations driven by the conditions and tools of the action at hand. The middle level consists of individual or group action that is driven by a conscious goal. At the top level, collective activity is driven by an object-related motive (Engeström 1998). Engeström (1987) expressed Leont'ev's revised Activity Theory model into this figure:

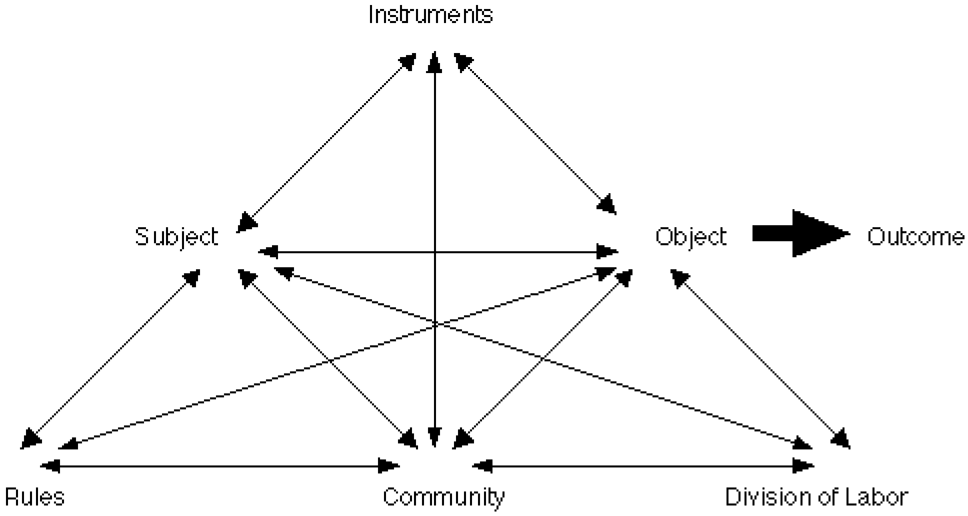


Figure 2: Activity Theory Model (Engeström 1987, p. 78)

Figure 2 shows how people (subjects), in their quest to achieve a purpose or objective (objects), are mediated by tools (instruments) and cultural factors (community). The latter two concepts in turn define the rules (rules) and roles (division of labour) within which the subjects act (Hasan and Handzic 2003).

Although a major area in Activity Theory research currently is understanding the interactions between multiple interacting activity systems and the process of continuous development, we are only concerned about using Leont'ev's model in a single activity perspective. By using his model to map out the processes and concepts involved in the activity of fraud auditing e-commerce systems using continuous assurance, a better understanding of the situation will be obtained.

4.3 Activity Theory Conceptual Model

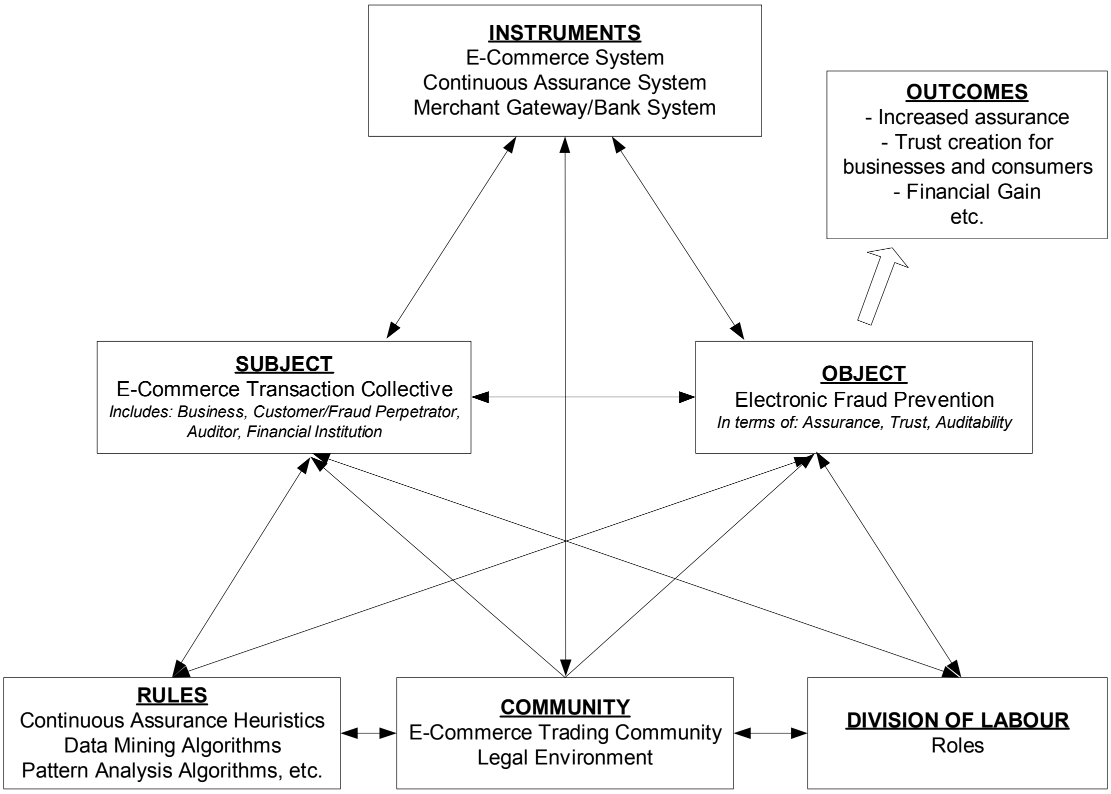

INSTRUMENTS
E-Commerce System
Continuous Assurance System
Merchant Gateway/Bank System
OUTCOMES
- Increased assurance
- Trust creation for businesses and consumers
- Financial Gain
etc.
SUBJECT
E-Commerce Transaction Collective
Includes: Business, Customer/Fraud Perpetrator, Auditor, Financial Institution
OBJECT
Electronic Fraud Prevention
In terms of: Assurance, Trust, Auditability
RULES
Continuous Assurance Heuristics
Data Mining Algorithms
Pattern Analysis Algorithms, etc.
COMMUNITY
E-Commerce Trading Community
Legal Environment
DIVISION OF LABOUR
Roles

Figure 3: Conceptual Model for Fraud Auditing E-Commerce Transactions using Continuous Assurance

Figure 3 integrates all the material discussed in the literature review. This section will review how the activity of fraud auditing e-commerce systems using continuous assurance works by first looking at the subjects and objects involved. After that, we shall see how this activity is mediated by tools (primarily technology) and the community (surrounding environment). Delving deeper, we shall explore the rules that underpin the process, along with the division of labour (roles) the entities within this model have.

4.3.1 Subject and Object

The subject of this model is **the collective involved in the e-commerce transaction process being audited**. These include the following parties:

- E-Commerce Transaction Collective
    - Business (the vendor)
    - Customers / Fraud Perpetrators

    - Financial Institutions
    - Auditors (may be internal or external to the business being audited)

The interaction of these parties produces the activity of e-commerce and thus they are also involved in the activity of e-commerce fraud auditing. The object of this activity is the **successful prevention of electronic fraud**. The object can be divided up into the following concepts:

- Electronic Fraud (prevention)
    - Auditability: Judged in terms of how easily auditable an e-commerce system is. This covers things like the effectiveness and ease of integration with e-commerce systems, along with the auditability of subject matter (section 3.3.1).
    - Assurance / Trust: The level of assurance that can be provided. The thoroughness and timeliness by which fraud is detected and prevented.

Evaluation of auditability and assurance is an issue that will be addressed in the implementation phase of this thesis.

4.3.2 Instruments

Naturally, the main instruments mediating this activity are the technological systems involved:

- E-Commerce System: The production system run by the business that customers purchase from. The transactions generated by these purchases are processed by this system and involve financial institutions (banks, for financing and credit processing) and, of course, the auditors who are auditing the system.
- Continuous Assurance System: This is the solution put in place to achieve the object of fraud auditing the transactions processed by the e-commerce system. It contains many components within itself – for example: an AVS, transaction agents, audit hooks and/or SCARF database. This system is controlled by a

rule set (see section 4.3.4), as are the other systems, although it is the rules affecting this system that are the most relevant to our interests.

- Merchant Gateway / Bank Systems: These systems are the property of the financial institutions who process transaction payments. For example, they handle the authentication of credit card details and the actual transfer of funds between business and customer accounts.

4.3.3 Community

The environment which this activity operates in has an indirect effect on the activity itself. The two cultural aspects that have most bearing upon it are:

- The E-Commerce Trading Community: This concerns the phenomenon of e-commerce and the Internet, and all those interacting with it. Section 3.1 explains the relevance of understanding e-commerce's effects upon this situation (such as trends of increasing reliance on e-commerce systems over traditional sales channels and methods). Customer and business perceptions of Internet security are also components of this community factor.
- The Legal Environment: The legal agreements between the groups involved in this activity explain how the situation has arisen. See sections 3.2.1 and 3.2.2. For example, because even though the fraud perpetrator is the one ultimately guilty of fraud, and is liable for the costs incurred, the e-commerce environment makes it difficult for this perpetrator to be caught. In this event, legally it is the business or merchant that ends up losing money.

4.3.4 Rules

The rules most relevant to this activity are the heuristics under which the continuous assurance system acts. These rules include the various algorithms and processes that continuous assurance system uses in order to achieve its tasks of detecting fraudulent transactions.

- Refer to section 3.3.3.3
- Data mining algorithms

- Rule based algorithms, ripple down rules
- Authentication rules

Naturally, financial systems and the e-commerce system operate according to their own set of rules, and the subjects operate within the legal environment (as a set of 'rules'), but we are looking at integrating our continuous assurance solution into existing systems as effortlessly as possible. This means the onus is developing a continuous assurance system that will cause minimal disruption to the current systems, yet still effectively perform its function.

4.3.5 Division of Labour

This section describes the role each major entity plays in this activity.

Customer / Fraud Perpetrator

Customers purchase goods or services from the business. Customers become fraud perpetrators when they misrepresent themselves in a way causing harm to business. Most commonly, this takes the form of providing fraudulent credit details in an effort to evade payment for goods or services.

Financial Institution

Financial institutions, most often banks, interface with business e-commerce systems to provide the facility of processing the financial side of transactions, handling the transfer of funds and limited verification of credit details. Sometimes they may partake in fraud detection schemes too (see section 3.2.4).

Auditor

The auditor's role is crucial in implementing an effective continuous assurance system. The Auditor is the primary party who designs, configures, maintains and uses the system. In consultation with the business, it is the auditor's task to determine what methods and techniques should be employed to integrate a continuous assurance system into a business' e-commerce system.

It is also the auditor's task to configure the system so that the rules by which fraud is detected and handled are optimised. As the auditor is not as familiar as the business with the nature of the transactions passing through the system (and what constitutes as abnormal), rule development will have to be done in close consultation with the business.

Once the continuous assurance system is active, it is the auditor's responsibility to handle the reports and alerts that the system generates during its operation. Auditors should then report back to the business about any fraud occurring and how possibly to handle it.

The auditor can be internal, if the solution implemented is managed and run within a business. The auditor can also be external, if external expertise is required – businesses often do not have the information technology resources required in order to implement such a system. The external auditor may customise their own product for the business, or design one from the ground up.

As already noted, the auditor must have proficiency in information technology and business processes in order to successfully implement an effective fraud auditing solution.

Business

Apart from the normal operations required to manage the e-commerce system the business operates, it is the business that must take the initiative and decide to address the problem of fraud in the first place. As discussed in the literature review, information systems security is a strategic issue that businesses must address at the management level. Management may delegate the responsibility of such a project to the information systems department, but they should continually be updated on the progress of it.

It is the business' duty to collaborate with the auditors in order to decide how best to plan a continuous assurance system. As no one is as familiar with the business operations as the business' management, it is up to them to provide the necessary information to auditors. This includes how their e-commerce system operates, the nature of transactions it processes, and details on patterns that may indicate fraud. The auditors translate these details for use in the actual continuous assurance system.

Management has to ensure proper resourcing. This includes financing the operation, headed by a feasibility study. Given that these systems can get quite complex and thus costly to implement, management must properly scope the size of the project and fund it. They must ensure that personnel with the relevant expertise are available, whether they be employees, externally sourced, or both (Wang, Bailey, Hidvégi and Whinston 2000).

Finally, it is the responsibility of business to respond to the outputs of the continuous assurance system. For example, an urgent alert generated by the system may necessitate immediate action on the business' behalf. For a high value transaction, the transaction may be held for manual inspection before being blocked or allowed to proceed. The operation of the system may also reveal changes in the business that will lead to improvements. For example, if a certain payment method is associated with substantially higher cases of fraud detected, then the business may look at further measures at securing that form of payment (perhaps in collaboration with financial institutions).

Continuous Assurance System

This system handles the fraud auditing of transactions that the e-commerce system it is integrated with processes. This includes the capture, filtering and analysis of data to detect fraud, according to the rules it has been programmed with. Subsequently, this real-time detection must be coupled with reporting such that auditors and businesses are aware of the occurrence of fraud. Action may then be taken to prevent it.

4.3.6 Outcomes

"Transforming the object into an outcome motivates the existence of an activity." (Mappin 1999)

The outcomes of this activity result in fraud reduction, which in turn benefits the business in several ways:

- Increased assurance and creation of trust – due to the presence of internal security controls, customers feel more comfortable with using the e-commerce system to make transactions with the business.
- Financial benefit – costs as a result to debts written off due to fraud are lowered. Revenue may be increased due to competitive advantage over competitors (customers are more likely to visit secure e-commerce sites than insecure ones (Elliot and Fowell 2000)). Because customers feel more comfortable, those withholding from e-commerce due to security concerns may be enticed to try it out, increasing the e-commerce-using customer base.
- Increased use of e-commerce systems by business, as business confidence is also increased. More business operations may be integrated with e-commerce systems.
- Deterrence of fraud.

4.3.7 Conclusions

This chapter satisfies RQ1 by modelling the environment in which electronic payment fraud occurs and noting how it can be addressed, as well as satisfying RQ2 by showing how continuous assurance can be applied to e-commerce systems in order to detect fraud.

Activity Theory has been expanded to include research into actions between multiple activity systems. Hasan and Handzic (2003) note that activities may produce the instruments, subjects or rules used in another activity system. The conceptual model outlined above has several of these interacting activity systems that feed into it. For example, Data Mining research can be utilised as part of the Continuous Assurance System tool in this model. Therefore, collaboration with other areas of information systems research is necessary to ensure the effectiveness of this activity.

Furthermore, continuous assurance is one method by which fraud may be prevented. Other activities may occur in conjunction to achieve the same objective. This acknowledges that other methods exist for detecting fraud, other than continuous assurance systems.

As eSCARF is an instrument within the conceptual model, the development of it corresponds to the creation of an instrument within the Activity Theory framework. The conceptual model places eSCARF within the larger context of the e-commerce fraud auditing environment and highlights the need for eSCARF's development.

Finally, this activity is itself part of a larger activity system – that of running a business, whose overall objective is profitability. Section 3.2.2 highlights why it is in the interests of business to engage in this activity, and how it melds in with this overall objective by potentially lowering costs and increasing revenue.

The conceptual model presented in this section covers at a high level the activity of fraud auditing e-commerce transactions using continuous assurance. It presents a generalised framework explaining how this activity occurs, and the requisites for the activity to be effectively achieved. From it, we can also see the context in which continuous assurance systems exist as an instrument for fraud auditing. This aids the development of such systems (such as eSCARF), in determining their scope and responsibilities.

Chapter 5. Research Methodology

The output of this research involves three parts: a theoretical component, a system implementation and a user evaluation survey. This section describes the methodology associated with each of these parts.

The literature review presented in chapter 3 forms the theoretical background for this thesis, placing it in the context of current research and explaining the significance and relevance of it. In an attempt to tie together the concepts covered in the literature review, a generalised conceptual model has been generated from it which allows us to better understand the problem domain.

The system implementation involves building and testing a 'proof of concept' continuous assurance system called eSCARF. Apart from the production of a working piece of software, the implementation will provide insight into how a continuous assurance system operates, as well as the procedures necessary for integrating such a system with an e-commerce system. This will answer RQ3, detailing how a continuous assurance system functions.

The user evaluation survey involves exploratory research into the general quality and perceived usefulness of the implemented eSCARF system from the perspective of auditors (thus answering RQ4). Feedback received from this survey should shed some valuable insight into improving eSCARF, whether auditors see the system as useful, and also indicating what auditors desire in a continuous assurance system (thus answering RQ5).

As the system implementation requires an engineering approach, and the case study requires a partially qualitative and quantitative approach, the two research methodologies required for both components are separate and distinct. This section is thus divided into two parts where the research methodology of the system implementation and evaluation survey are separately discussed with regards to their approaches.

5.1 System Implementation

5.1.1 Research Aims and Expected Outcomes

The system implementation section of this research aims to build a 'proof of concept' continuous assurance system. An implementation of the SCARF audit method, eSCARF, will be produced, adapted from Ng and Wong's (1999) early prototype version. eSCARF will be converted to be compatible with a commercial e-commerce environment in current use, IBM WebSphere Commerce 5.4.

Apart from the aim of producing a working piece of software, the implementation should also provide a better understanding of what comprises a continuous assurance system, and how it integrates with an e-commerce system. That is, how a continuous assurance system functions (RQ3). The emphasis here is not on building a system which employs the latest, most sophisticated technical fraud detection methods (audit strategies). Other research, especially in science, in the field of data mining, pattern detection and statistical analysis methods, deal with this area much more thoroughly (eg: Anandarajah and Lek 2000; Wang, Hidvégi and Whinston 2001). However, the system should be designed so that it can easily be modified to employ other fraud detection strategies. Special attention will also be paid to the processes involved in customising eSCARF for integration with an e-commerce system.

This section will comprehensively document the design process of eSCARF. This documentation, in addition to giving insight into the architecture of continuous assurance systems, can also be used as a reference for further development work on eSCARF, or for development work on continuous assurance systems of a similar nature.

5.1.2 Research Approach

System implementation requires an engineering approach to research (as opposed to qualitative or quantitative information systems approaches) as it involves the design, development and testing of an information system. Clarke (2000) notes that this type of research is *"essentially concerned with technology, including artefacts, techniques and combinations of both of them."*

Two traditional engineering research orientations are applied research and problem-solving, the latter of which is applicable to this research. Problem-oriented research, as the name implies, begins with a problem, in our case, fraud. In an effort to devise a solution to the problem, this approach experiments with existing technologies (or prototypes new ones), which is true in the case of this research.

The engineering approach that was used is construction. Construction is defined as:

> *"the conception, design and creation (or 'prototyping') of an information technology artefact and/or technique (most commonly a computer program, but sometimes a physical device or a method). The new technology is designed to intervene in some setting, or to enable some function to be performed, or some aim to be realised. The design is usually based upon a body of theory, and the technology is usually subjected to some form of testing, in order to establish the extent to which it (and, by implication, the class of technologies to which it belongs) achieves its aims."* (Clarke 2000)

The specific method of construction that was used was the waterfall system design life cycle (Royce 1970), where the development of the system progresses through a series of stages: requirements analysis, system design, system coding, and system testing.

5.1.3 Research Methodology

The system implementation was divided into various stages roughly corresponding with the traditional waterfall system design life cycle. eSCARF was originally implemented for the IBM Net.Commerce suite environment. It uses audit hooks, combined with a quasi-ripple-down rule system and reporting functionality to provide continuous assurance for detecting fraud. eSCARF will be refined from this original prototype and adapted for IBM WebSphere Commerce.

Requirements Definition and Analysis

The first stage of gathering requirements is understanding the context in which the system is being developed. This requires an understanding of the e-commerce system that eSCARF is being designed for, namely, WebSphere Commerce. Therefore, an analysis of WebSphere's architecture and how it manages e-commerce was necessary.

Secondly, a requirements specification which details all the desired functionality of the system was be developed. The requirements specification was adapted from the feature set provided by Ng and Wong (1999). The specification also added functionality that takes into account the post-implementation suggestions Ng and Wong made, as well as other enhancements.

Implementation

The task following the requirements definition phase is to implement eSCARF in Java. A conceptual design document will be written up which will detail, at a high level, the components of eSCARF, how they work, and how they interact with each other. A code review (Pfleeger 1998) will be conducted to determine what coding needed to be done.

After this, the actual program will be implemented in Java and integrated with a sample WebSphere e-commerce store.

Testing

During the previous phase of system implementation, unit testing will occur as the system is incrementally modified. This phase involves testing the entire completed system in order to ensure its proper operation. Testing will also demonstrate how the system handles complex auditing rule sets. Preliminary testing will be derived from that performed by Ng and Wong (1999). For testing the new functionality present in eSCARF, two test scenarios will be generated, each with separate test data sets. During the first test scenario, screen captures will be taken to visually document eSCARF in operation.

5.1.3.1 The Bigger Picture

Ultimately, developing software is an ongoing, cyclical process. Systems undergo a continuous cycle of development, testing, release, evaluation and then further development again as developers strive to implement improvements and bug fixes, as well as catering for the changing wants of users (Pfleeger 1998). Indeed, this thesis represents only one iteration of this cycle. In addition to improving upon Ng and Wong's (1999) early eSCARF prototype, the changing e-commerce environment has outdated Net.Commerce, thereby necessitating eSCARF to be adapted to a new e-commerce system, WebSphere. The current version of this software too, will eventually undergo further refinements. It is therefore necessary to look towards the future to see how WebSphere may be further developed, and an evaluation survey, discussed next, will help accomplish efforts to continually improve the eSCARF software.

5.1.4 Limitations

Section 6.6 documents a listing of limitations encountered while implementing eSCARF.

5.2 Evaluation Survey

The evaluation survey will answer RQ4 and RQ5. It will determine if auditors found eSCARF was useful for assuring e-commerce systems (RQ4) and also determine what factors are important to the design of a continuous assurance system such as eSCARF (RQ5). The answers to RQ5 include what is already in eSCARF (what auditors liked about the current version of eSCARF) and what is now (suggestions for future improvements).

5.2.1 Research Aims and Expected Outcomes

The aim of the survey will be to perform a user evaluation of the eSCARF fraud detection continuous assurance system. This research will be of an exploratory nature. Exploratory research is an attempt to develop a preliminary, rough description or understanding of a phenomenon. The phenomenon under investigation for this research is eSCARF. Exploratory research is *"necessary when very little is known about the topic being investigated, or about the context in which the research is to be conducted."* (Blaikie 2000, p. 73) eSCARF, being a new, prototype system, fits this criteria.

Blumer (1969) explains how the exploratory process helps researchers achieve better understanding through the creation of a detailed 'picture' of the phenomenon under study:

> *"The picture provides the scholar with a secure bearing so that he knows that the questions he asks of the empirical area are meaningful and relevant to it, that the problem he poses is not artificial, that the kinds of data he seeks are significant in terms of the empirical worlds, and that the leads he follows are faithful to its nature."* (Blumer 1969, p. 40)

Up until this point, users have not been actively involved in the development of eSCARF. Users have had no input into Ng and Wong's (1999) design of the system. User evaluation of the completed prototype has not occurred yet, either. Pfleeger (1998) notes that user involvement is essential in the software development cycle, as they are the best qualified party for evaluating software issues dealing with appropriateness of audience, ease of use and other human factors. Furthermore, performing testing with users of the system is, *"essential, especially if they were not present when the system requirements were first defined. A user is likely to be familiar with the problem because of daily exposure to it, and can be invaluable in evaluating the system to verify that it solves the problem."* (Pfleeger 1998, p. 343) Therefore, performing a user evaluation is an important aspect of developing any system.

An evaluation of eSCARF enables us to obtain a much fuller picture, or understanding of the system – something that cannot be obtained by purely internal development processes.

This improved understanding will be crucial to directing eSCARF's future development and ensuring that it is an effective assurance system. This evaluation will be acquired the system's target users: auditors. Auditors will be evaluating eSCARF in a test environment that simulates a real world e-commerce scenario.

Specifically, the aims of this evaluation survey will be to:

- Evaluate the quality and perceived usefulness of eSCARF.
- Discover how eSCARF may be improved upon, either by modifying the current system, or with the addition of new functionality.
- Explore what auditors perceive as important in a continuous assurance system.

By achieving these aims, we also gather knowledge that may be used in the future development of eSCARF (since software development is a cyclical, ongoing process – see section 5.1.3.1) and continuous assurance systems in general.

The survey will use a questionnaire as the research instrument. This questionnaire will be partially quantitative (where it attempts to measure the quality of eSCARF), and partially qualitative (where suggestions and comments from the participant are collected) in nature.

5.2.2 Research Approach

In light of the research aims above and the exploratory nature of the research, we will briefly discuss this research's philosophical assumptions and stances. These stances shape the nature of the research, as well as determine what perspective frames the research methods the best.

5.2.2.1 Ontology, Epistemology and Theoretical Perspective

Ontology refers to how we understand the world to exist, an understanding of 'what is' (Crotty 1998). This research adopts a 'realist' ontology, where the universe is seen to be made up of discrete and observable events. Realism assumes that social phenomena exist independently of both the observer and social actors. It is the regularity and patterns of this reality that realist research strives to discover and describe (Blaikie 2000). The phenomenon under study, eSCARF, along with its properties and attributes, is believed to be an observable and discrete entity existing in the real world, thus matching the realist ontology. eSCARF is not considered to be a representation of an ideal in our minds (an idealistic ontology), nor purely a label for a phenomenon without deeper meaning (a nominalistic ontology).

Epistemology refers to how we understand knowledge is found or constructed, and what kind of knowledge is possible. There are three predominant forms of epistemology: objectivism, constructionism and subjectivism (Crotty 1998). Objectivist epistemology maintains that meaning and reality exists independent of consciousness. Crotty (1998, p. 8) explains that, *"understandings and values are considered to be objectified in the people we are studying and, if we go about it in the right way, we can discover the objective truth."* In concurrence with this perspective, eSCARF exists, whether there is a human mind to observe it or not. Meaning is not constructed by observing the phenomenon (as in constructionism), nor is meaning completely imposed upon the phenomenon by the observer (as in subjectivism). Instead, meaning is embedded within the phenomenon's existence, and may be determined by objective human observation. Our aim is to objectively discover information about the eSCARF continuous assurance system.

Positivism believes that knowledge is posited in what is being observed. Hence, reality can be *"described by measurable properties which are independent of the observer (researcher) and his or her instruments"* (Myers 1997, p. 241). Therefore, positivism, which forms the theoretical perspective of this research, is a natural extension of the ontological and epistemological stances detailed above – that phenomena, such as continuous assurance systems, are observable in the real world and that their inherent meanings and traits can be discovered by the use of human senses. Objective observations form data through which the state of reality can then be documented. This objective data in turn becomes theoretical statements about the order of reality. It must be noted here that although this survey is collecting data that is subjectively expressed by the survey respondents, the manner in which the data collection occurs is objective. Furthermore, Blaikie (2000) classifies the inductive research strategy described in the next section as ascribing to the positivist theoretical perspective.

5.2.2.2 Research Strategy

The inductive view of strategy believes that *"meticulous and objective observation and measurement, and the careful and accurate analysis of data, are required to produce scientific discoveries."* (Blaikie 2000, p. 102). Generally, inductive research involves three steps:

1. Accumulation of data, where facts are objectively observed and recorded without attributing any relative importance to any of them.
2. The data gathered is reduced, analysed, compared and classified *without using hypotheses*.
3. Inductive logic is used to derive generalisations as to the relations between facts. (Hempel 1966)

The main criticism with an inductive strategy is that no observations can be made from a completely objective standpoint (due to factors like researcher bias). Also, observations must themselves be guided, lest how would the researcher know what data is relevant for collection? This issue is rectified by introduction of concepts into the research. These concepts, and the way they are defined, guide the data collection process. For this research, the concept of fraud is pertinent and has been examined in the literature review.

As this survey's aim of building a better understanding of eSCARF and continuous assurance systems is descriptive in nature, the inductive strategy is therefore a useful and valid one (Blaikie 2000).

5.2.3 Research Design Overview

Surveys are a generalised means of data collection, accomplished through the use of research methods such as questionnaires or interviewing. Often the terms 'survey' and 'questionnaire' are used interchangeably, but a survey is considered to be a broader term which refers to an overall research design of which a questionnaire may be a component. A survey must account for other areas such as data collection and analysis (Frazer and Lawley 2000).

Surveys are used extensively in information systems research as a way of turning observations into theory. The survey process has been outlined by Newsted, Chin, Ngwenyama and Lee (1996). Firstly, questions are formulated which attempt to measure concepts of interest. Observed responses from surveyees to these questions are turned into data by using a research instrument such as a questionnaire. Secondly, the data gathered is aggregated, and then analysed, normally by quantitative formulas. The results of the analysis then lead to establishing relationships between concepts and producing conceptual representations of what has been measured. Researchers use these results to make sense of the overall phenomenon being studied. The survey process thus is also suitable for, and consistent with, an inductive research strategy.

Traditionally, the survey is a research design used for quantitative research, as it is particularly conducive to quantitative analysis. In quantitative studies, data typically begins as words which are transformed into numbers and subjected to statistical manipulation during analysis (Blaikie 2000). Similarly, data collected in a survey is normally either translated into a number format immediately, through the pre-coding of responses (such as providing respondents with a fixed set of options to select from), or just prior to the analysis stage, where post-coding is applied to responses given in words. Nonetheless, Newsted, Huff and Munro (1998) acknowledge that a survey may be a qualitative tool as well, complementing other forms of data or observations. In the context of this research, because of the limited number of respondents (see section 5.2.4.2), full quantitative analysis cannot be performed. Instead, quantitative data will be complemented by qualitative data. Therefore, this survey is both quantitative and qualitative.

It is in light of the aim of system evaluation, the exploratory nature of this research, and a variety of practical constraints that a survey has been selected for the design this research will adopt. One strength of surveys is that they are easy to administer, score and code (Newsted, Huff and Munro 1998). The data collection process is not resource intensive, which makes it suitable for this thesis which has resource and time constraints. A survey's ability to be used in a quantitative and qualitative manner also permits a richer evaluation of eSCARF.

There are certain key aspects to be considered when designing a survey. Tasks required to be performed in a survey can be broken up into two categories: data collection and data analysis. Data collection involves the construction and validation of the research instrument (the selected instrument for this research is a paper-based questionnaire), determining the sources of data and then actually carrying out the collection process. Data analysis involves collating the data collected, applying relevant analytical techniques to it, and writing up the results. The following sections will document the design process used for this thesis. The design of the questionnaire instrument has been separated out into section 5.3.

5.2.4 Data Collection

5.2.4.1 The Questionnaire Process

The research instrument used for this survey is a paper-based questionnaire, designed to provide evaluative feedback on the eSCARF system. Participants will be asked to fill out the questionnaire only after the system has been demonstrated to them. The demonstration process involves taking the participant through a walkthrough of the system (they may ask questions or sidetrack into other parts of eSCARF during the walkthrough, if desired). This is followed by a period where they may wish to interact with the system themselves and ask further questions about it. Although the demonstration will be performed by the questionnaire administrator (who will be the researcher), the administrator will not be physically present when the participant fills out the questionnaire (so as not to influence the questionnaire results). A copy of the walkthrough procedure is detailed in appendix 8. The questionnaire also comes with a briefing section, handed to the participant before the system is demonstrated to them. This briefing section serves to familiarise the participant with the purpose of the questionnaire, and the general nature of the system they will be evaluating.

The rest of this section will discuss the particulars of the data collection process – how participants are chosen, the form data collected will take, and how the questionnaire will be constructed and validated.

5.2.4.2 Sources of Data

Sources of data is concerned with where the data will be collected from, what type of data is being collected, and what properties the data being collected has.

Type and Form of Data

All data collected will be primary data, as it is being entirely generated by this research. Because the questionnaire includes questions of both a quantitative nature (consisting of scaled response questions) and a qualitative nature (consisting of open ended questions), the data collected will both be in numerical and word form.

Setting

This questionnaire is being delivered in an artificial setting. Blaikie (2000, p. 192) notes that a *"limited range of social research places people in experimental or simulated conditions in order to study some form of social behaviour in a controlled environment."* Experiments and simulations are two types of methods that collect data in an artificial setting. Because this questionnaire is being run directly after a demonstration of eSCARF, a process controlled by the researcher, a simulation is being performed. Participants will be experiencing use of eSCARF under simulated conditions – eSCARF has not been integrated into a real life e-commerce system, but is instead integrated with a test e-commerce system that simulates a real-life online store.

Timing

The timing of this survey will be cross-sectional, providing a snapshot evaluation of the current version of eSCARF. A longitudinal study (over multiple versions of eSCARF) would yield more informative results, but due to time constraints, was not undertaken. The survey will take place over a fortnight where the data is collected from the participants.

Data Selection – Sampling

Sampling, which defines where or who the survey data is collected from, has significant ramifications for the research being performed. Because it is not normally practical to perform a census, where a survey is conducted over an entire population, a sample of the population is taken instead (Frazer and Lawley 2000). A sample that is surveyed, in a best case scenario, should be perfectly representative of the whole population. However, as this is not normally true of a sample, there are two main factors to consider that influence the degree of certainty about the inferences that can be drawn from one. Firstly, the larger the size of the sample, the more likely it will be more representative of the population. Secondly, the greater the extent of variation (of characteristics under study) within a population, the greater the level of uncertainty that a sample is faithful in representing the population (Malhotra 1996).

The population for this survey are auditors, or professionals who have experience in auditing, as they will be the main users of eSCARF. The auditors are also likely to have a background in using information systems, given the prevalence of such systems in industry (Vasarhelyi, Kogan and Sudit 2000). Although a background in continuous assurance is not necessary, knowledge and expertise in the field of auditing, is necessary. An issue with this target sample is ensuring that participants match this profile. This issue is addressed with screening questions, discussed in the questionnaire design section (section 5.3.2.1).

Two major limitations exist to the way sampling has been carried out with this survey. Firstly, the sampling method for this survey is a single-stage, non-probability convenience sample. A convenience sample is, *"Any sample in which the probability of a sample member's inclusion in the sample cannot be computed."* (Schonlau, Fricker and Elliott 2001, p. 33) In other words, convenience sampling samples from a population based on accessibility, expediency, cost, efficiency or any other reason not directly related with sampling parameters. The advantage of this method is the simplicity of obtaining data. However, the disadvantage is the high potential for large and unmeasured bias to exist within the sample. Statistical inference is therefore much more problematic because the sample may be highly unrepresentative.

Secondly, despite the fact that the response rate for this survey is expected to be 100%, the sample size for this survey is extremely low (15 auditors, 7 with a background in information systems and 8 with a traditional auditing background). Therefore, as noted above, the less likely the sample will be representative of the population. Quantitative analysis on this sample may not yield meaningful results, but provide an indication of the potential for eSCARF.

The two problems above are, however, not as dire as they appear. This is due to the nature of the sample as well as the nature of the research aims of this survey. A small convenience sample of staff in the University of New South Wales, eligible for this survey (that is, with past auditing experience), was used for this research due to time and resource constraints. Nonetheless the problem of an extremely small sample size is offset by the expectation that the population being sampled will not be highly variant in nature. This is because the characteristic endemic in the population that is relevant to this survey is their view of what is important in a continuous assurance system. This view is strongly related to their understanding of auditing principles (and to a lesser extent, information systems use), which is likely to be similar amongst them given their experience in the field.

Statistical inference is not the primary aim of this survey, due to the small sample size. Some statistical analysis will be applied to the data collected as an exploratory measure, although it is not expected to generate any significant findings. Thus, findings may be indicative of what to expect in future evaluative studies on eSCARF.

Finally, Joppe (2002) notes that despite the disadvantages of a convenience sample, the information obtained from it *"could still provide some fairly significant insights, and be a good source of data in exploratory research."* Therefore, convenience sampling is quite valid for this exploratory research.

5.2.4.3 Instrument Creation and Validation

This section details the procedure followed to create and validate the questionnaire used as the research instrument for this survey. The development of the actual questionnaire and the validation procedure is documented in section 5.3.

Instrument creation is important, and an effective and valid instrument is required to perform effective and valid research. *"Attention to instrument issues ... brings greater clarity to the formulation and interpretation of research questions. In the process of validating an instrument, the researcher is engaged, in a very real sense, in a reality check. He or she finds out in relatively short order how well conceptualization of problems and solutions matches with actual experience of practitioners."* (Straub 1989, p. 148)

There are a series of tasks that need to be completed before a questionnaire is properly prepared for data collection. The following list of tasks has been adapted from Frazer and Lawley (2000), Malhotra (1996) and Newsted, Huff and Munro (1998):

1. Determine sampling and appropriate response rate.
2. Determine the measures of constructs that will be used.
3. Prepare a draft questionnaire, determining:
    - Question content
    - Question wording
    - Response format (eg: multi-item scales)
    - Structure and layout
4. Internal validity testing.
5. Piloting the questionnaire and assessing reliability, construct validity and content validity.

A description of each of the above tasks follows. These tasks are documented in the context of the evaluation survey in section 5.3.

Task 1: Sampling and Response Rate

Section 5.2.4.2 details sampling methods. A response rate to the questionnaire of close to 100% is expected, due to the small sample size and the targetting of participants.

Task 2: Construct Measures

Constructs, or the concepts that are being investigated, need to be measured somehow. Newsted, Huff and Munro (1998) advises that literature should be searched for existing measures of constructs. A search of literature may also turn up an already designed and validated instruments that may address the same concepts a researcher is trying to measure. The determination of constructs and measures provide a guideline for the next section, where the questionnaire is drafted.

Task 3: Drafting the Questionnaire

Question Content

Drafting question content refers to writing up the actual questions that will go into the questionnaire. This process involves translating into words the construct measures determined in the previous task.

Question Wording

Choice of wording is important in questionnaire design. Properly worded questions help avoid misinterpretations by respondents, and should be conducive to being answered by the respondents. Respondents are more likely to give an accurate answer if the questions are phrased such that they appear appropriate, relevant and neutral (not loaded) (Frazer and Lawley 2000).

Response Format

Response format refers to the possible types of responses possible to questions posed to respondents. Three types of format are possible: open-ended, close-ended and scaled responses. Open-ended responses allow the respondent to give free-formed answers, useful for when there is no set of responses that can be predicted. Close-ended questions provide the respondent with a fixed set of choices to select from. Scaled responses use a scale format, attributing numbers to options, in order to measure the attributes of a construct. There are four commonly used scale formats: nominal, ordinal, ratio and interval.

Nominal scales merely attribute numbers to responses for categorising purposes and does not express any values or relationships between variables. Ordinal scales categorise answers based on their ordered relation to each other, although the ordering is relative and the exact degree of difference between consecutive answers is unknown (eg: an attitudinal Likert scale). Interval scales are similar to ordinal scales, except that the degree of difference between consecutive answers is the same (eg: a temperature scale). Finally, ratio scales include all the properties of interval scales, but additionally include a meaningful zero point (eg: an age scale). The type of scale that is used must suit the question being asked, and also affects the type of quantitative analysis that can be employed (Joppe 2002). For example, nominal scales, as the numbers do not really signify anything quantitatively are not suited for this type of analysis.

Structure and Layout

Structure mainly refers to the ordering of questions as it may affect the motivation of, and manner in which participants answer questions. Screening questions, which verify the participant's eligibility to complete the questionnaire are normally placed first (Frazer and Lawley 2000). Demographical questions are sometimes recommended to be placed last in a questionnaire, when the participant feels more comfortable about answering such questions. Order bias occurs when the ordering of questions have unintended flow-on effects to subsequent questions. Care must be taken to think about what types of effects early questions may have on the answering of subsequent questions, as the flow of questioning may inadvertently 'lead' the participant to answer in a particular fashion.

Layout refers to the visual design of the questionnaire. The questionnaire should be clearly and cleanly presented, and most importantly, not confusing. The layout of the questionnaire should not impede the effective answering of questions.

Task 4: Internal Validity Testing

Internal validity is the degree of confidence the researcher has in the casual effects between variables (Frazer and Lawley 2000). This task is performed by the researcher carefully reviewing the questionnaire draft to ensure all questions are clearly presented and are unambiguous from the researcher's perspective.

Task 5: Piloting the Questionnaire and Ensuring Validity

Piloting, or pre-testing the questionnaire gives the questionnaire a trial run. Piloting straightens out any potential problems that exist in the instrument, and imparts some useful information about what to expect when the finalised questionnaire is deployed (such as the questionnaire's completion time). Colleagues are typically used in this process because they understand the study's purpose and have similar training as the researcher, from an academic standpoint. During the pilot process, the questionnaire may be revised several times in response to feedback obtained from the pilot.

Questionnaire validity also is assessed at this stage. 'Validity' can be divided into three components, reliability, construct and content validity (Straub 1989). Validity is normally determined by technical methods, piloting and/or expert validation.

Reliability relates to the consistency of results that can be obtained over time. That is, if the results of a study are reproducible under a similar methodology, then the research instrument is considered to be reliable (Joppe 2002). Questions that can be easily misunderstood by respondents leads to low reliability, so the goal is to maximise the clarity of questions.

Content validity is concerned with how representative the questions and response formats are in measuring the constructs. Having a content-validated instrument will increase the validity of the data collected since the bias associated with the selection of questions used to measure constructs will be removed.

Construct validity is based on how results from an instrument relate to other measures in the theoretical environment under study (Newsted, Huff and Munro 1998). A questionnaire would have construct validity if the results obtained from questions positively correlate with the constructs they are designed to measure.

5.2.5 Data Analysis

Surveys generally undergo a quantitative form of analysis, but due to the sampling constraints noted section 5.2.4.2, the analysis of this survey in this manner will be limited. The quantitative analysis methods that will be used are mainly descriptive (means, modes, minima, maxima and standard deviations). Because the sample contains two subgroups – auditors with backgrounds in information systems and auditors without – one-way ANOVA (analysis of variance) tests will be used to determine if any significant differences exist between the responses of the two subgroups. Correlation matrices and linear regression modelling will be used in order to explore if any relationships between variables exist.

One-way analysis of variance compares the means of one or more groups as based on an independent variable (in our case, whether the participant was an IS auditor or non-IS auditor) by using variances. It allows us to see if the means between groups (in our case, responses between different types of auditors) differ significantly.

Correlation attempts to find if two variables are correlated (related) to each other. A correlation matrix is a table which shows the intercorrelations among all the variables (Hair, Anderson, Tatham and Black 1998). The relationships tested are users' evaluations of eSCARF's attributes to eSCARF's overall usefulness, and users' evaluations of eSCARF's individual components to eSCARF's overall usefulness.

Regression modelling is conceptually about fitting a line through two or more variables in order to explain the dependency of one variable on the others. In other words, regression modelling attempts to predict a relationship between a set of independent variables and a dependent variable that are related in a nondeterministic fashion (Devore 2000). Two models, corresponding to the two correlation matrices, are created, with the dependent variable in each case being overall usefulness. Due to the small sample size and the expectation that a significant regression model will not be generated, stepwise regression will also be performed. Stepwise regression adds independent variables into the model one at a time, based on the discriminatory power they add to the prediction. The resulting model excludes all independent variables whose inclusion would cause the model to lose significance.

Quantitative analysis will be supplemented by qualitative analysis. Therefore, it is necessary to briefly review methods of analysing the qualitative data derived from the open-ended questions in the questionnaire.

Dey (1993) formulated a method by which qualitative data analysis occurs in three stages: describing, classifying and connecting. Describing is a process that will have already been accomplished in the data collection process.

Classifying involves distilling concepts from the raw data collected in order to enable effective analysis of the data which may be voluminous. Classification involves looking through the data, creating categories and classifying sections of data into those categories. This approach is called coding, or the *"process of breaking down, examining, comparing, conceptualizing, and categorizing data."* (Strauss and Corbin 1990, p. 61). Physically, coding occurs by reading the data line by line or paragraph by paragraph and assigning those sections codes which refer to the concepts they have been classified under. The questionnaire developed for this survey partially pre-categorises responses due to the specific nature of the questions. Nonetheless, the responses provided by different respondents will still have to be reconciled and categorised to comprehensively analyse all the data collected.

Finally, connecting the data involves finding the relationships between different categories, after the data has been classified.

5.2.4 Survey Limitations

See section 8.7 for a discussion of limitations associated with surveys, as well as specific limitations of this survey research.

5.3 Evaluation Survey Instrument Creation Methodology

5.3.1 Determining Measures and Constructs

The survey's primary purpose is to evaluate the quality and usefulness of eSCARF as a continuous assurance system. Within the field of information systems, the explanation of what causes information systems to be effective has been a major theme in research (Yap and Thong 1996). While, there is no universally accepted model for measuring information systems effectiveness (quality and usefulness), measuring the level of user satisfaction connected with a system has been a popular method (Harrison and Rainer 1996, p. 81). Therefore, constructs from research in user satisfaction have been derived from various sources in the area.

Doll and Torkzadeh (1991) developed an instrument for measuring user satisfaction, the End User Computing Satisfaction Instrument (EUCSI). EUCSI divides user satisfaction into five dimensions: content, accuracy, format, timeliness and usability (see also sections 6.3.5.4 and 6.3.5.5). The former four dimensions relate to the information provided by the system, whereas usability refers to the way users interact with the system (Nielsen 1998). EUCSI has been extensively validated in subsequent research (eg: Farhoomand and Etezadi-Amoli 1991; Hendrickson, Glorfield and Cronan 1994; Gelderman 1998) and has therefore been adapted for the creation of this survey's questionnaire. That is, this questionnaire will evaluate eSCARF based on the five EUSCI constructs, with the end users being auditors.

Furthermore, Galetta and Lederer (1989) concluded that user satisfaction is dependent upon users' perceptions and attitudes. It is for this reason that the questionnaire first measures auditors' perceptions of the constructs, followed by their attitudes towards them as they appear in eSCARF.

The open-ended qualitative questions are not based on EUCSI. Their purpose is to qualitatively determine auditors' thoughts of, and suggestions for, eSCARF. They supplement the data gathered from the construct measures by providing the rationale behind participant's response choices for quantitative questions.

5.3.2 Drafting the Questionnaire

A copy of the draft questionnaire that was pilot tested is available in appendix 6. Question references in this section refer to the draft version.

Attached to the front of the questionnaire is a briefing section, which will be given to a participant before the system is demonstrated to them. This briefing section serves to familiarise the participant with the purpose of the questionnaire, about the research area of continuous assurance, and the general nature of eSCARF. The questionnaire should be answered in one sitting and only after eSCARF has been demonstrated. (A copy of the walkthrough procedure is provided in appendix 8.)

The questionnaire itself is divided into four main sections, which will now be discussed separately.

5.3.2.1 Demographics

The start of the questionnaire collects various demographical details about the survey participants. The participant's name is collected only for internal questionnaire identification purposes, no names will be mentioned in the analysis. The rest of the questions are screening questions which test a participant's eligibility, as participants are required to have attained a level of expertise in auditing. Therefore, an ordinal scale is used whereby participants give a relative indication of their knowledge and expertise in three areas: information systems, information systems auditing and continuous assurance. A ratio scale is used to ask participants to quantify how many years of experience they have had with auditing. The largest response option of "greater than 5 years" is deemed to equate to significant experience, therefore another interval (such as 5-10 years) was not required.

5.3.2.2 Section A – Perspectives

This section measures the perspective of participants on what makes a continuous assurance system effective. As can be seen, the questions have been derived from EUCSI constructs (comprehensiveness and conciseness are measures of 'content', presentation, user-friendly interface and ease of customisation are measure of 'usability').

Because the sampling for this survey comprises university academic staff, it is felt that the 'factors' listed will be understandable (content validity). This should be validated during pilot testing.

The Likert Scale

For section A and all subsequent sections in the questionnaire, a 7-point Likert scale (an ordinal scale) is used for the response format. In section A, a selection at the lower end of the scale signifies that the participant considers a factor as 'very unimportant'. A selection at the opposite end signifies a fact as 'very important'. The other five choices give a gradient of answers in between. (Sections B and C are similar. A selection at the lower end of those scales signifies the participant 'strongly disagrees' with the question statement. A selection at the opposite end signifies 'strong agreement.') The Likert scale, although strictly an ordinal scale, commonly has numbers attributed to it to make it an interval scale (Malhotra 1996) so that it can be used for quantitative analysis. This questionnaire attributes a value of 1 to the lower end of the scale, progressing up in increments of 1, to 7 at the upper end.

A 7-point scale was selected because it is felt that participants in a 5-point scale are biased against giving responses at the extremities (a 1 or a 5). Answers are therefore weighted towards the centre (2, 3 and 4), effectively resulting in a tendency to act like a 3-point scale. Therefore, a 7-point scale is used. This provides more variation so we may also differentiate between varying magnitudes of disagreement or agreement.

5.3.2.3 Section B – eSCARF Component Evaluation

Because eSCARF is made up of various modules, some modules may be of a better quality than others. By looking at each module that the auditor will use in turn (rule management, the server console log, rule checking and alerts, and web reporting), we may gain specific knowledge about the individual components in eSCARF. This knowledge is especially useful for future development of eSCARF, as improvements may be targeted at specific modules.

This section is divided into subsections, one for each module being examined. At the end of each subsection, an open-ended question is asked, prompting the participant to provide further comments or suggestions regarding the module under evaluation.

EUCSI constructs are also used here, as in section A. However, because only some constructs may be applicable to a particular module, only the relevant constructs for each module are included. For example, the server console log is not actively ‘used’ by the user – it is merely viewed, therefore usability is not examined, only the properties of the information delivered (questions 14-17).

5.3.2.4 Section C – eSCARF Overall Evaluation

This section acquires a participant’s view of eSCARF as a single system. Questions 27-33 use the same factors as in section A. This allows participants’ perceptions (of important factors in continuous assurance systems) in section A to be compared with participants’ attitudes (to how eSCARF exhibits these factors).

Question 34 gives an overall rating of how useful the participant felt eSCARF was (an entirely holistic view of eSCARF).

The closing two questions are open-ended questions. The first question aims at collecting additional ‘general impressions’ of eSCARF, a qualitative evaluation of the system. The second question aims at suggestions for added functionality in eSCARF, perhaps to be integrated into the requirements specification for future eSCARF development cycles.

5.3.2.5 Internal Validation

The draft questionnaire was proofread and checked for readability and understandability. Some tweaking occurred of certain phrasing of questions, and of question ordering.

5.3.3 Piloting the Questionnaire

The questionnaire was pilot tested with academic staff who are considered experts in the field of information systems auditing. The pilot test participants were asked to comment on the questionnaire in terms of its readability and understandability.

Expert validation for reliability, construct validity and content validity was also undertaken during the pilot. Although research typically uses methods to evaluate validity (eg: Cronbach alphas), the pilot sample size was insufficient to permit this. This has been noted in the limitations section. Nonetheless, construct and content validity, as well as reliability, were evaluated and verified by the participants, who were familiar with the constructs, and background literature related to them.

The comments and suggestions raised by the pilot participants were subsequently integrated into the questionnaire to improve its validity. Ambiguities were clarified, and various questions were reworded. These changes are documented below. Refer to appendix 7 for the post-pilot, finalised questionnaire.

Revisions to the Briefing

- A few minor changes to wording were applied to this section.
- The comparison of continuous assurance to traditional auditing was rewritten after input was received that it “doesn’t make sense”.

Revisions to Demographics

- The year ranges chosen for the question, “How many years experience have you had in auditing?” were validated as being suitable for their purpose. That is, although there are auditors with more than 10 or 15 years of experience, any more than 5 years is considered somewhat experienced in auditing.
- “What is the extent of your knowledge in…” was rephrased to, “How would you rate the extent of your knowledge in…” Similarly, “What is the extent of your expertise in…” was rephrased to, “How would you rate your expertise in…” This revised phrasing implies the participant is giving a self-perceived view.
- “Auditing” was added as an additional screening question.

- The issue of self-efficacy problems was raised. Self-efficacy is a social cognitive theory which relates to "the belief in one's capability to organize and execute the courses of action required to manage prospective situations." (Bandura 1997, p. 2) In other words, the self-perception of one's own abilities to perform certain actions or tasks. This questionnaire relies on the participants having expertise in auditing. To ensure this, screening questions are used to evaluate participants' knowledge and expertise. However, those questions ask participants to self-evaluate their knowledge and expertise, and self-efficacy problems enter because a self-evaluation of ability may not reflect ability in reality. Various research (Pajares 1997; Weinberg 2002) has shown that when actual ability is low (but not non-existent), perceived self-efficacy tends to be higher than reality. The converse is true when actual ability is high. Therefore, this fact must be taken into account when the screening questions are analysed. Input from the pilot suggested that self-evaluation of knowledge and expertise be moderated based on the years of experience in auditing, which is a more objective measure.

Revisions to Section A

- 'completeness' was deleted from question 2, as the concept of 'completeness' is different from 'comprehensive'. The presence of the former word in the question was therefore confusing.

Revisions to Section B

- Question 8 was deleted because the concept of 'comprehensiveness' is not relevant to the rule management module. It is more a tool to build rules, and not for providing any more information than is necessary to get the task done (unlike reports, which should provide *all* relevant information, including information which is not consciously required by the auditor at the time the report is generated).
- In question 14 (and also in questions 21 and 27), the phrasing "is accurate" was changed to "appears to be accurate". Because this is a simulated demonstration of eSCARF, the auditor cannot make a precise assessment of whether eSCARF provides accurate information all the time – only whether it *appears* to provide it, given what has been witnessed.

- Question 17 was corrected for a grammatical error.
- Question 20 was deleted. Alert generation is a functional requirement of eSCARF, and being a necessary part of the system, is by nature useful. Therefore, it is a biased question.
- Question 23 was revised to include "well formatted" to additionally define "presented well" as referring to the format of the information provided.

Revisions to Section C

- In the final question, "Do you have any suggestions for…" was rephrased to, "Do you have any further suggestions for…"

General Notes

- Ensure that the questionnaire administrator is not physically present when the participant fills out the questionnaire. Otherwise, this presence is likely to bias the replies of the participant.
- Order bias in questions may be experienced due to perceptions (section A) being measured before attitudes (section B). For example, a participant who rates a factor in section A as very important, may be likely to pay more attention to the presence of that factor in eSCARF during the demonstration, and be less attentive to other factors. Unfortunately, the same is true if attitudes are measured before perception. The experience of using eSCARF may affect what the participant perceives as important in a continuous assurance system. For example, if something is performed poorly by eSCARF, the participant may find it necessary to attach greater importance to that aspect when considering continuous assurance systems in general. Therefore, it was decided to retain the current ordering of questions.
- It was felt that given the nature of the participants, the measures used in the questionnaire would be clearly understandable to them (a reliability assessment).

5.3.3.1 Suggestions and Changes Made to eSCARF

The output from the open-ended questions in the pilot were mainly suggestions and comments about eSCARF's functionality. A number of these suggestions were integrated into eSCARF to improve it before the full-sample survey was performed.

These changes have been made to the system described in chapter 6 and are described below, in addition to other comments made during the pilot.

Web Reports: Transaction Logs Viewable by Date Range

As the operation of eSCARF in conjunction with highly trafficked e-commerce system is likely to generate a sizable quantity of data, it is often necessary to restrict the viewing of such data to a window of time that is of interest to the auditor. Therefore, an additional form has been added to the *Transaction Summary Report* to allow the auditor to select a time range, down to a precision in hours, of transactions to retrieve.

Web Reports: Export Transaction Logs

Often, auditors may want to perform offline analysis on transactions, either by manual processes, or by external programs such as ACL (Audit Control Language) or Microsoft Excel. An option to export transaction logs to a simple CSV (comma separated values) file was added. This export option was added to the *Transaction Summary Report* and returns a list of transactions and their details, along with the relevant product, item and price lists for those transactions.

Web Reports: Viewing Alerts by Rule Triggered

It is a useful capability to view all the alerts that have been triggered by a single rule. This is for auditors who want to concentrate on the operation of a single rule (for instance, to see the effectiveness of a newly imposed rule). This functionality was added to the *Rule View Report*.

Additionally, a statistical display was added to the *Rule Management Page*. The display is a graph, showing rules by the frequency of alerts they have generated. In this way, auditors can visually inspect which rules are receiving the most 'traffic'.

Rule Checker: Batch Processing (unimplemented)

A recommendation was made for the inclusion of batch processing in the system. Because transaction analysis is a potentially computationally intensive task, having to capture and analyse transactions simultaneously may be too taxing on system performance. Instead, the rule checker could be configured to analyse rules in nonpeak times, when the rate of incoming transactions is lowest. This would reduce the processing burden of the system by smoothing it out over the course of a day. Although alerts would not be generated in real-time, they would still be timely. This would be an option for the auditors, depending on the number of transactions processed and the level of system performance required of the e-commerce system.

5.4 Ethical Considerations

Ethical issues to consider when carrying out this research are noted in this section.

5.4.1 About eSCARF

Although there are few ethical concerns surrounding the development of software (apart from the obvious issues of software theft and plagiarism of code), due consideration must be given to the nature and proper use of the software developed.

eSCARF is a system used by auditors to assure an organisation's e-commerce system, which means that during its operation, eSCARF handles all the transactions generated by the e-commerce system. These transactions contain many sensitive customer details such as credit card numbers, addresses and other personal information. The transactions are stored in the eSCARF database and become a cache of sensitive data. It is therefore the auditor's responsibility to ensure proper and secure handling of this data by only authorised personnel. The personal information contained in this data also raises privacy issues, and any existing, applicable privacy legislation must be adhered to, such as Australia's *Privacy Act* 1988.

Auditors must take reasonable measures to secure the computer eSCARF is operating on, and also to secure the communications link that connects eSCARF to the e-commerce system it is assuring. Security considerations must include physical measures, to prevent unauthorised physical access to the eSCARF computer, and logical measures, such as securing the computer from unauthorised remote access. Security measures should be as stringent, or more stringent, than the security measures operating on the client's e-commerce system so as not to compromise their client's data. Otherwise, the auditor's system will become a tool that increases risk for an organisation, instead of decreasing it.

Auditors should collect only the portions of transaction data that are necessary to effectively perform assurance. Extraneous, unused data is unnecessary and its presence gives potential for its misuse. Furthermore, auditors should always keep their client organisation notified of how eSCARF operates, along with how and what details are being collected from their e-commerce system. Client organisations should also be made aware of the modifications required to their system that are necessary to get it to interface with eSCARF.

eSCARF currently uses 'TEA', a free two-way encryption algorithm written by a third party, found in the file `TEA.java`. If eSCARF's code is modified, and the use of TEA is retained, all header and copyright information pertaining to TEA within that file must be left unmodified.

5.4.2 About the Survey

The survey procedure, and how the data collected will be handled, were disclosed to the survey participants. This included describing the purpose of the research, the expected benefits arising from it and an assurance that results would be kept confidential from public use.

Survey results appear in an aggregated format. Individuals' names are not cited in the results where quotations are taken from participants. The questionnaire was filled out by participants without the physical presence of the questionnaire administrator so as not to unfairly influence the results.

5.5 Conclusions

This chapter detailed the research methodology for two parts of this thesis – the system implementation (addressing RQ3) and the evaluation survey (addressing RQ4 and RQ5). Each part requires a different research approach.

The system implementation requires an engineering research methodology as it involves the development of an information system. The engineering approach used was construction, and the method of construction used was a single iteration of the waterfall system design life cycle. The life cycle divides the development of the system into several progressive phases: requirements definition and analysis, implementation and testing.

The evaluation survey involves the user evaluation of eSCARF. Because a user evaluation of the system has never been undertaken before, the research is considered exploratory and thus an inductive research strategy was employed. The aim of the evaluation is to assess eSCARF's usefulness and efficacy in fraud auditing e-commerce transactions, as viewed from an end-user's perspective (i.e.: an auditor). The research method selected to achieve this aim was the survey. A questionnaire instrument was created, pilot tested (section 6.3) and used to perform data collection. The sample for the data collection was a group of 15 auditors. The questionnaire collected data of both a quantitative (using scaled response questions) and qualitative (using open-ended questions) nature. Data analysis for the survey involved some statistical analysis of the quantitative data (descriptive statistics, correlation matrices and regression modelling) and some quantitative analysis of the qualitative data via the coding analysis technique. Analysis was undertaken with the survey design's inherent limitations in mind – especially the limitation of a small sample size, which reduces the ability to reliably generalise results.

Chapter 6. System Implementation

6.1 Overview

This chapter comprises a discussion of the processes undertaken to design and implement eSCARF for IBM WebSphere Commerce. The development of the system loosely followed a waterfall systems design life cycle (Royce 1970), commencing with a requirements analysis stage, before progressing onto system design, coding and testing.

A design overview will provide information about the context in which eSCARF is being developed, namely, the IBM WebSphere environment. An understanding of WebSphere's architecture is crucial as it is the system eSCARF must be properly integrated with. The requirements specification section details what functionality eSCARF should have. The specifications are divided up into several modules representing logical groupings of functionality. The section on conceptual design details system design, exploring each of the modules in greater depth, explaining how they work and interface with WebSphere to provide fraud auditing functionality. The changes and enhancements that have been made to the original Ng and Wong eSCARF for Net.Commerce application will also be noted.

System testing will be briefly covered, including documentation of system installation and usage. Implementation notes and issues will discuss observations and issues that arose during system development. Of particular interest in this part are the issues to do with systems integration, for example, how eSCARF integration procedures may differ if adapted for other e-commerce systems. Finally, further avenues for development of features not implemented in this version of eSCARF will be noted.

This section is primarily a non-technical discussion of the eSCARF system, focusing on what the system does, and how it does it from a 'high level' point of view.

6.2 Design Overview

Section 3.3.3.2 explains how SCARF uses embedded audit modules placed within various points of an e-commerce system. To see how SCARF may be integrated with WebSphere, an understanding of WebSphere's architecture and e-commerce transaction module is necessary.

IBM WebSphere Commerce 5.4 (for Windows NT/2000)[11] is a packaged solution providing the necessary software infrastructure to operate an e-commerce store, be it B2C or B2B. WebSphere Commerce itself is composed of several components: IBM HTTP Server, IBM WebSphere Application Server, IBM WebSphere Commerce and IBM DB2 Universal Database. For this thesis, WebSphere Commerce was installed on a Windows 2000 Advanced Server machine. The system requirements necessary to operate WebSphere Commerce and eSCARF are shown in appendix 9.

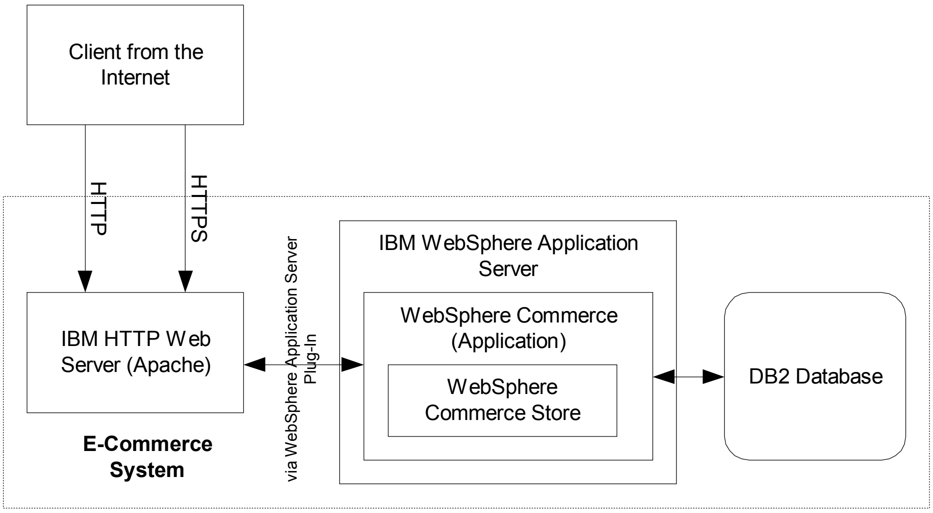

Client from the Internet
HTTP
HTTPS
IBM HTTP Web Server (Apache)
E-Commerce System
via WebSphere Application Server Plug-In
IBM WebSphere Application Server
WebSphere Commerce (Application)
WebSphere Commerce Store
DB2 Database

Figure 4: WebSphere Commerce Architecture

Figure 4 shows how the WebSphere components interact with each other. The IBM web server, which is based on the Apache web server[12], is the gateway for communications between customers and the rest of the e-commerce system. HTTP and secure HTTP (HTTPS) requests arrive here and may be forwarded on to the WebSphere application server. The WebSphere application server plug-in is used to handle this communications process.

[11] www.ibm.com/websphere

[12] www.apache.org

WebSphere application server hosts a variety of web applications, of which WebSphere Commerce is one of them. Under a single instance of a Commerce application, one or several stores may be set up. All the data for these stores, and WebSphere applications is stored in the DB2 database. WebSphere is flexible in that it may be set up in a one, two or three tier configuration. In a one tier configuration, all components are installed on one machine. In a two tier configuration, the DB2 database is separated onto a different machine. In a three tier configuration, the database and the web server are both separated onto two different machines. In the case of two and three tier setups, communication between the components occurs over a network interface. eSCARF was developed on a single tier configuration.

It should be noted that often an additional component, IBM Payment Manager, is installed. Payment Manager takes over the responsibility of handling payment for transactions, interfacing with an external financial institution's gateway. Since we are not concerned with how transactions are ultimately carried out by the business, but merely how we can capture these transactions as they occur, Payment Manager was not installed. Thus, all payment information remained within the WebSphere Commerce application.

6.2.1 WebSphere Commerce Application Architecture

Figure 5 will be discussed using a top-down approach. It shows what components, or layers, make up the WebSphere Commerce Application, and which layers are of interest to us.

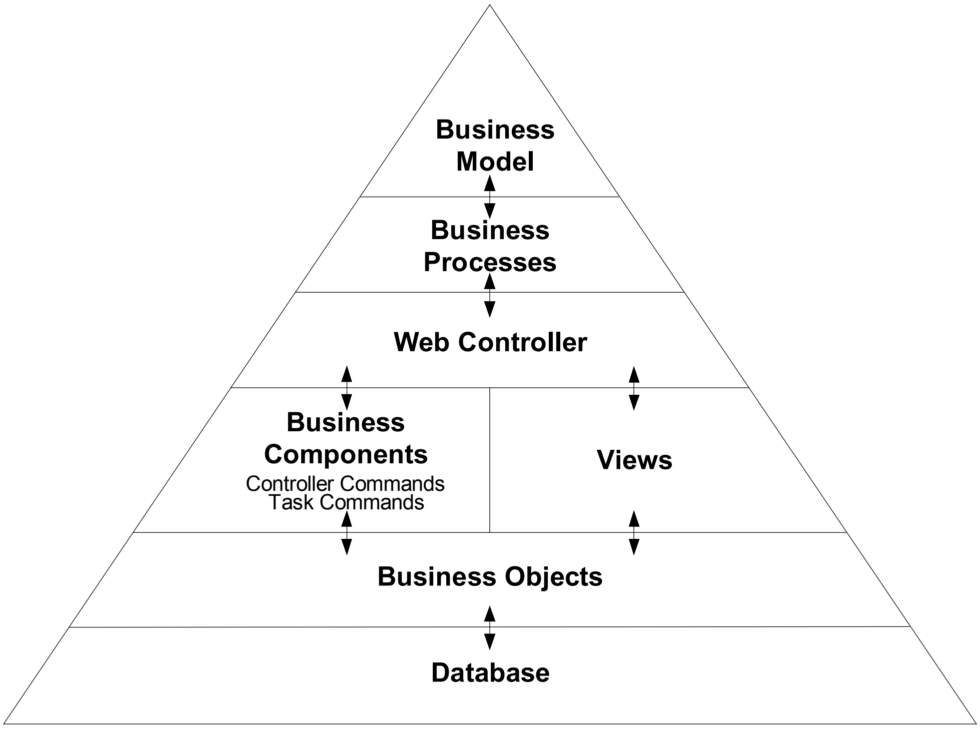


Figure 5: WebSphere Commerce Application Architecture (adapted from IBM 2002, p.4)

Business Model and Business Processes

The top two layers define the business requirements of a store in the application. The top layer refers to what e-commerce business model, such as B2C or B2B, a store will be based upon. This focus filters down to how the other layers will work. Under the business model comes various business processes. These processes comprise the workflow and siteflow by which a business model is implemented. Typical business processes may include user registration and catalogue navigation.

Web Controller

This layer handles requests received via HTTP. Each request, depending on the nature of the request, activates a set of business logic which carry out the request. Once complete, the controller invokes a view, which is the response returned to the user.

Views

Views are the method by which responses to user requests are displayed. They are implemented using JSPs (Java Server Pages).

Business Components

Business components are units of business logic (code). These discrete units are known as business logic 'commands', of which there are two types: controller and task commands. Controller commands encapsulate all the logic required to carry out a single request. Controller commands use task commands to perform a single unit of work. An example of a task command would be processing payment details. This structure allows code to be highly modularised and is thus easily maintainable. It is also in the library of task commands that we can insert our audit hook. This will be discussed in more detail later. Commands are implemented in Java and may use business objects for database access, or may access the database directly.

Business Objects

Business objects are data entities which provide an interface for business components to fetch and store data with the database layer. This interface comes in the form of EJBs (Enterprise Java Beans) which encapsulate the logic needed to extract information from the database. They facilitate data access in that the need to comprehend the complex relationships between databases tables and columns is mostly removed. Using EJBs also allows database queries to be carried out using Java code instead of reverting to SQL.

Database

WebSphere Commerce stores all its data in a database using a specifically designed database schema. The schema has been tailored for e-commerce applications and the data required to support them. Examples of database tables in the schema include:

- Order: a table which stores information about individual orders
- OrderItem: a table which stores information about the individual items (such as price and quantity ordered) which make up an individual order in the Order table.

6.2.2 WebSphere E-Commerce Transaction Model

For internal testing of eSCARF, the working sample store, 'WebFashion', provided with WebSphere, was used. It uses a B2C business model, in accordance with this thesis' focus. WebFashion is a retail store with a catalogue of clothing and accessories.

Customers entering the site can browse the store catalogue, adding items they wish to purchase to a shopping cart. A shopping cart view allows customers to remove items or modify item quantities from their cart. Once the customer is satisfied with his or her purchase list, they begin the checkout procedure. Users may also register at the site and set up a user account. A user account expedites future purchases from the site as it stores personal details (eg: shipping addresses) so that the user will not need to re-enter them during checkout.

Checkout involves the collection of various customer details, including billing and shipping details (phone numbers, addresses, etc.), the shipping method, and payment details. Once the order is submitted, it is stored in the database awaiting fulfilment from the business. The business must manually approve orders before fulfilling them.

Since we are interested in the point of the system where payment transactions are processed, let us look closely at the checkout procedure when payment details are submitted.

The order payment view (OrderDisplayPending.jsp) calculates the final cost of the order, including cost of goods sold, tax and shipping. It also requests payment details including credit card type, number and expiry date. Once credit card details are entered, they are submitted to the OrderProcess controller command, which calls upon various task commands which in turn process the order and enter the relevant data into the database. It is at this point where we want to insert our audit hook so that we can capture the details of the customer's completed order for processing by eSCARF.

6.3 Requirements Specification

The requirements specification will list the functionality of eSCARF, broken up into various modules. The specification will then provide a basis for the design of the system. The eSCARF system monitors transactions passing through an e-commerce system. These transactions are then processed by the eSCARF server, which applies a set of audit rules to this data and flags potentially fraudulent transactions, according to these rules. Auditors may be alerted in different ways to such transactions. Depending on the severity of transactions triggering alerts, the actual alerts may be accorded a corresponding level of visibility (that is, the more severe the potential for fraud, the more visible and urgent an alert will be to the auditor). Therefore, eSCARF must also provide a way to manage these rules and alerts. Reporting functionality must also be built in, summarising the data collected by eSCARF in a readily readable format that may be accessed by the auditor, and also presented to the business itself, if needed.

The primary user of eSCARF will be auditors, although interaction with the employees of the business whose e-commerce transactions are being assured is necessary in order to properly set up and integrate eSCARF with the e-commerce system.

6.3.1 Audit Hook: Capturing Transaction Data

The SCARF method of auditing requires that all relevant audit data be monitored (unlike SARF, a method which is the same as SCARF, but randomly samples data). Therefore, a mechanism must exist to pull such data from the e-commerce system, called an embedded audit module, or audit hook. This hook must integrate with the e-commerce system and forward on the audit data (in our case, payment transaction data) to the eSCARF system. In this way, the two systems are kept relatively separate, with the hook acting as the only eSCARF component that must be integrated with the e-commerce system. It is through the hook that the two systems must necessarily communicate. To minimise the work that must be performed to integrate the two systems is advantageous as it is less disruptive to the activity of the current e-commerce system.

Two things must be determined to create a hook: where to place it within WebSphere (how to integrate it), and what data does it need to capture to pass onto the rest of eSCARF. Determining what data is required is an important step in planning, due to the fact that different e-commerce systems may offer different data, and sufficient data must be provided to eSCARF if it is to properly detect fraud.

In our case, the audit hook is placed at the point where the final order is submitted by the customer, and captures the following details:

- Customer ID
- Customer credit card details (type of credit card, card number and expiry date)
- Merchant ID (WebSphere Commerce's store ID, as multiple stores may exist under a single instance of the WebSphere Commerce application)
- Timestamp when the order was placed
- Products ordered (include product IDs, prices and quantity ordered)

There are many more details that may be captured (such as a customer's designated billing and shipping address, tax costs, IP addresses) but the data above is sufficient for our proof of concept system, and also reflects the data most strongly related to payment information. (These additional details, especially customer addresses, should be captured in future versions of eSCARF.)

Another requirement is encryption. Because transaction details, which are sensitive and confidential, are potentially being sent over a network by the audit hook to the rest of eSCARF, some sort of encryption should be used on these details to protect them from being intercepted and viewed by unauthorised third parties.

6.3.2 Rule Management

As the introduction in 6.3 states, the data captured by the audit hook is compared with a set of pre-defined rules. It is with these rules that an auditor will implement a strategy for detecting fraud. Therefore, the auditor must be provided with an interface to manage this ruleset. This interface must allow the auditor to:

- define new rules
- view and modify existing rules
- delete existing rules
- define the action to be taken when a rule is 'satisfied' (triggered)
- save rules

Rules are further split into active rules and inactive rules. Active rules make up the ruleset which is applied to incoming transactions. Inactive rules are still stored within eSCARF, but are not applied. This allows an auditor to keep a historic record of rules, and also provides a way of testing different sets of rules. Therefore, also required is the ability to:

- activate rules
- deactivate rules

Note that once a rule is triggered by a transaction, it should not be able to be modified. Instead, changes to that rule must be saved in a new version of that rule. This preserves the linkages between transactions and the rules that they have triggered for future reference.

About Rulesets

Defining a proper set of rules such that fraud detection is effective is the domain of the auditor. It is not envisaged that the auditor will 'get it right' the first time. Instead, over time, as new fraud patterns emerge, rules will be fine-tuned to increase the accuracy of the audit strategy.

As Ng and Wong (1999) observed, if an unusually large number of transactions are being flagged as potentially fraudulent, then this could be due to two things. Either there is an actual increase in fraudulent activity occurring in the e-commerce system, or the ruleset used is too broad, producing a series of false positives (legitimate transactions being detected as potentially fraudulent). The converse is true as well, if too few transactions are detected as potentially fraudulent, either there is an actual decrease in fraudulent activity, or the ruleset is too narrow, producing a series of false negatives in which fraudulent transactions go undetected by eSCARF. It is the auditor's continual responsibility to ensure that the ruleset is optimised.

6.3.3 The eSCARF Server

When the audit hook captures transaction data, it must pass it on to the rest of eSCARF for further processing. The module of eSCARF that receives this data is the eSCARF server, which may be placed on another machine, such as an auditor's computer. This allows the eSCARF system to be physically separated from the e-commerce system.

The duty of the server is to process incoming transactions, and check it against the active ruleset, logging to the database and generating alerts as necessary. The data flow can be summarised as follows:

1. eSCARF server waits and listens for transaction data.
2. Upon receiving data, transaction is logged to the eSCARF database.
3. The transaction is compared to the active ruleset.
4. If any rules are triggered, an appropriate alert response is generated.
5. Loop back to 1.

The auditor must be able to control the starting and stopping of the server, as well as the TCP port it listens on. Apart from that, the operation of eSCARF will be automated. While the server is operating, its activity will be displayed on a server console log. This console log is for diagnostic purposes only, for a technical view of what the server is currently doing, and is not intended for auditor analysis (the web reporting is used for this purpose).

Generating Alerts

When a rule is triggered, eSCARF must respond appropriately. Auditors should have the ability to define the nature of this response (see 6.3.2), which may include logging the potential infraction to the eSCARF database, generating an onscreen alert, and/or sending out an alert email. The visibility of the action performed should correspond to the severity of the infraction. For example, a fraudulent transaction worth $10,000 is an order of magnitude more severe than one worth $100, and correspondingly the former should produce an alert more visible to the auditor.

6.3.4 eSCARF Reporting

Data processed and collected by eSCARF (otherwise known as the SCARF ‘file’) is stored in the database. However, an auditor must have the ability to retrieve this data for review and analysis in a more user-friendly way than directly accessing the database. Also, as viewing raw data is not particularly helpful, generating reports from the database will allow auditors to selectively drill down to the information they require. Therefore, a separate interface must be provided to perform this task of reporting. The reports should be concise, well formatted and understandable, providing the auditor with the necessary information for their audit.

A web interface is an appropriate match for this requirement as it provides a flexible graphical user interface that may be customised for different types of situations. Additionally, web pages are easily printed. The inputs to the web interface will be requests from the auditor to display certain types of data including:

- details of all transactions captured;
- details of transactions triggering rules;
- the ruleset in operation; and
- various summary statistics.

The behind-the-scenes processing required to generate these reports from the raw data should be hidden from the auditor.

6.3.5 Miscellaneous Design Issues

Although the principles below are not functional requirements, they are design issues that must be considered in developing eSCARF. These issues impact upon the functionality and usability of eSCARF, and do not apply to any specific component discussed above, but rather, apply to the system as a whole.

6.3.5.1 Modularity

As no two e-commerce systems are *exactly* alike, eSCARF must be customised to a certain extent in order to integrate it with an e-commerce system. Modular design separates a system's components so each component can be maintained separately from others (Pfleeger 1998, p.197). As long as the interfaces between components remain static, the logic within a component can be changed without affecting other components. This means that during customisation, only the components that require changing need to be changed. For example, if the system eSCARF is running on does not support JSP technology, but only ASP technology for web reporting, then only the web reporting component needs to be rewritten to accommodate this. All other components need not be modified.

6.3.5.2 Ease of Customisation

Although modular design helps separate a system into separate components, these individual components should be written such that they can be easily customised – whether this customisation takes the form of adapting code for a different e-commerce system, or adding extra functionality. Examples of this include:

- modifying the audit hook to capture additional transaction data;
- creating new types of alerts (for example, SMS alerts, if SMS infrastructure is available); and
- tailoring web reports to specific needs.

6.3.5.3 Portability

E-commerce systems will often run on a whole array of different platforms and environments. WebSphere is but one e-commerce system package, but others exist such as, iPlanet and Oracle Application Server, which offer different features and have dissimilar programming models. Additionally, they may run on other operating systems such as Linux, AIX, or derivatives of Unix. Therefore, if possible, eSCARF should be designed so that it has a high degree of portability between different combinations of platforms and environments. Optimally, with regards to technical compatibility, only the audit hook would require modification, as it is the sole point of integration of eSCARF and the e-commerce system. (Naturally, more modification than this is necessary to adapt eSCARF to the e-commerce system's business properties, such as selecting what transaction data to capture.)

6.3.5.4 Quality of Information

It is important that eSCARF communicates information it generates effectively to the auditors who use the system. Quality of information delivery can be divided up into various factors which include: content, accuracy, timeliness and format (Doll and Torkzadeh 1998). Content refers to whether the information provided meets the user's requirements. In other words, whether the information delivered is comprehensive or concise. Accuracy refers to the information provided being correct and truthful. Timeliness refers to how current and up-to-date the information delivered is. Format refers to the layout of the information and the way in which the information is visually presented to the user. The survey section of this thesis will use these factors in the evaluation of eSCARF.

Continuous assurance systems demand a high quality of information (that is, a high degree of comprehensiveness/conciseness, accuracy, timeliness and presentation), especially as they operate in real-time. eSCARF's information should be timely, to ensure auditors are getting the correct information within a relevant timescale; be accurate, to ensure proper analysis; be comprehensive, to ensure auditors receive all the information necessary; and be presented well to allow auditors to find the data they need easily.

6.3.5.5 System Usability

A system is only effective if it is used properly by its users. Nielsen (1998) defined usability as, *"the measure of the quality of the user experience when interacting with something – whether a web site, a traditional software application, or any other device the user can operate in some way or another."* Goodwin (1987) has found that the effective functioning of a system depends much on its usability, or ease of use. Usability refers to how 'user-friendly' a system is – how 'intuitive' and 'navigable' its user interface is. The design of the graphical user interface in eSCARF should be conducive to auditors being able to use and customise eSCARF with minimal training.

6.4 Conceptual Design

This section details the work done during the coding phase of system development. The conceptual design of eSCARF for WebSphere was adapted from Ng and Wong (1999), with modifications. These modifications came in two forms. Firstly, there was the conversion work necessary for eSCARF to work with WebSphere. Secondly, there were several enhancements made to eSCARF's functionality. The changes made to the 1999 version of eSCARF for Net.Commerce will be noted in this section.

Development commenced with a code review (Pfleeger 1998, p.290), as well as a review of the specifications and design documents. This gave a thorough understanding of the current state of the early eSCARF prototype, and aided the requirements specification phase for the new version of eSCARF (section 6.3).

Development occurred using an iterative process, whereby changes and additions made to the system were individually integrated and tested, gradually building towards fulfilling the requirements specification.

6.4.1 eSCARF Architecture

eSCARF has been divided up into a suite of several modules, each with their own set of functions. As shown in figure 6, these modules include the eSCARF server and rule checker, rule management, rule activator, reporting, the eSCARF database and the audit hook (although installed on the e-commerce system, the audit hook is still considered a component of eSCARF).

Installation Scenarios

eSCARF can be installed in two different configurations, one- and two-tiered. In a one-tiered configuration, both the WebSphere system and eSCARF system are installed on the same physical machine. Additionally, eSCARF's database and WebSphere's database may utilise the same DB2 server as each other (since DB2 can host multiple databases, called 'nodes' in IBM terms). Reasons for using a single tiered configuration include:

- Lack of a separate machine to implement a two-tiered configuration.
- Considering networking security issues is not required as eSCARF and WebSphere communicate on the same physical machine.

In a two-tiered configuration, eSCARF is installed on a separate machine from the WebSphere system. The two machines do not have to be in the physical vicinity of each other, but must be able to communicate to each other via TCP/IP. This configuration allows the auditor's copy of eSCARF to run from a location remote to that of the e-commerce system, allowing for remote monitoring. However, care must be taken that the communications occurring between the two systems be secured (for example, by encryption) as they now will be transmitted across a network or internetwork. Reasons for using a two-tiered configuration include:

- allows remote, off-site auditing by auditors;
- separates two systems so they can be independently maintained;
- improves performance as the load of processing e-commerce transactions for fraud is transferred onto the eSCARF machine; and
- allows for a dedicated eSCARF server – useful if eSCARF will concurrently audit multiple WebSphere stores running on different WebSphere servers (see section 6.4.5).

Note that in both cases, the eSCARF hook still uses a TCP connection to transfer transaction information captured to the eSCARF server, but for a one-tiered configuration, that connection is back to the local machine.

The system designed for this thesis was tested, out of convenience, in a one-tier configuration.

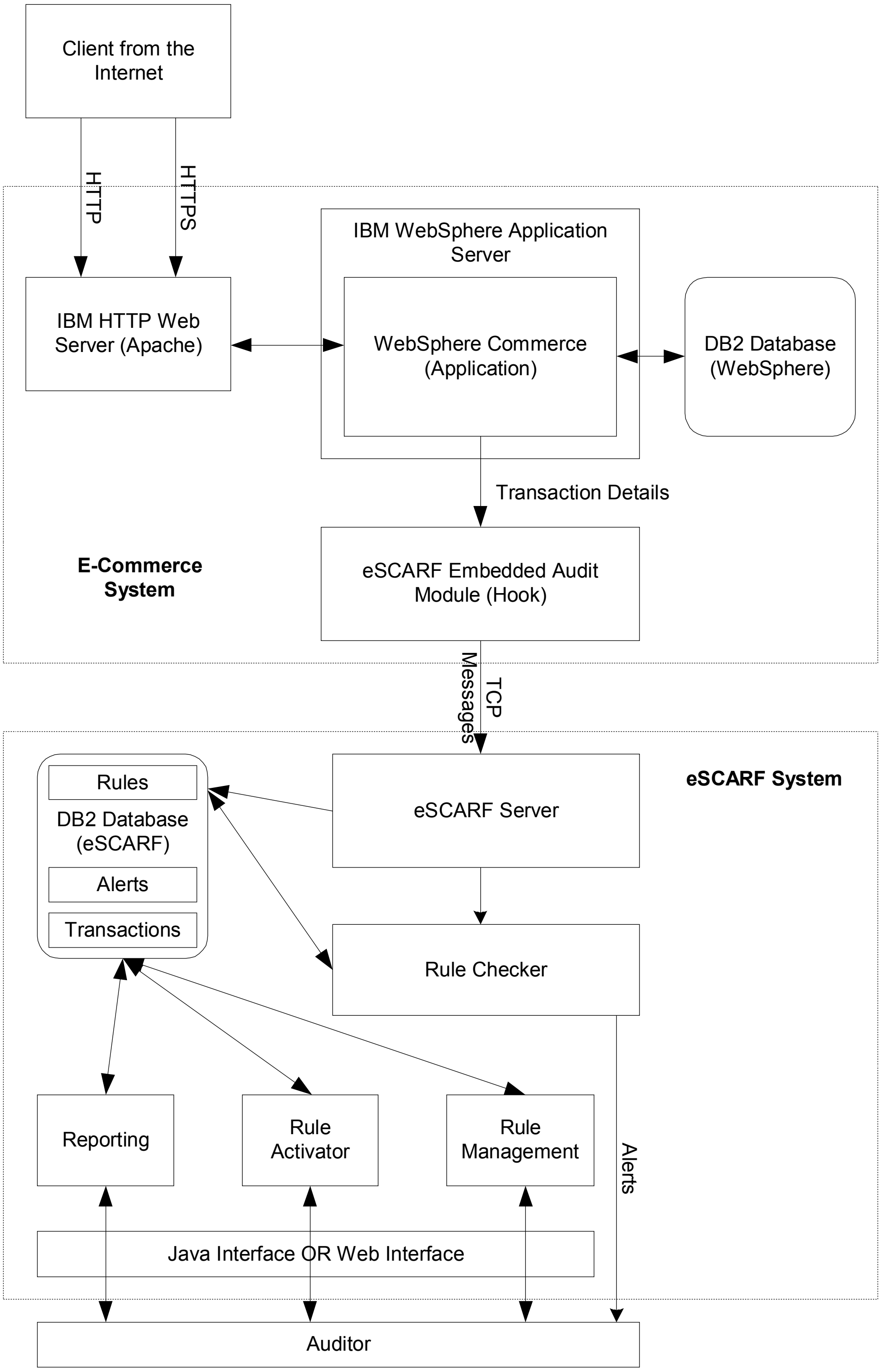


Figure 6: eSCARF Architecture

Overview of eSCARF's Flow of Operations

We can detail eSCARF's general flow of operations by dividing its activities into those related with a store customer's actions, and those related with an auditor's actions. A more detailed look at each of the individual modules that comprise eSCARF follows.

In a typical scenario, the eSCARF server will be started and listening for connections from eSCARF audit hooks. eSCARF will be set up with a ruleset.

When a customer finalises an order at the online store (submits the payment details), WebSphere calls a controller command which handles the ordering process. This controller command calls several task commands, one of which is the eSCARF audit hook. This hook, when called, gathers all the pertinent information of the transaction just submitted by the customer and then sends it to the eSCARF server. The eSCARF server receives the transaction information and logs it to the eSCARF database. The transaction is then processed by the rule checker module which checks it against the active ruleset. Any rules triggered by this transaction will generate an appropriate alert response (such as an onscreen alert).

It should be noted here that when a customer submits an order, WebSphere performs its own validity checking before calling the controller command which handles the ordering process. This includes checks such as: numeric fields do not contain alphabetical characters, and that the credit card number is valid by running it through a simple Mod 10 algorithm (however, this only validates the credit card number is syntactically correct, not if it is linked to a real, operating and open bank account. This is something eSCARF can be programmed to check.)

The auditor interacts with the eSCARF system in a variety of ways, accessing functions through the eSCARF main menu. From the main menu, the auditor can start and stop the eSCARF server, manage eSCARF rules (these rules are also stored in the eSCARF database) and view eSCARF web reports. Selecting the appropriate menu option will activate the corresponding eSCARF module.

Software Used

eSCARF was programmed in Java v1.3.1. It requires a Java runtime environment to run in. eSCARF web reporting was programmed using Java Server Pages.

For the eSCARF Java package hierarchy, refer to Appendix 3.

6.4.2 General Enhancements and Changes

Before considering each module individually, there are a few general enhancements and changes made to the earlier eSCARF prototype:

Upgrade to Java 1.3.1

As with the earlier version, Java was retained as the language eSCARF is written in. Java has two main properties that make it suitable – it is an object oriented language, and is not platform dependent (Java compiles to 'bytecode', which is platform neutral and theoretically runnable across different operating systems without the need to edit the underlying programming code). These attributes allow eSCARF to be programmed in a modular fashion (section 6.3.5.1) as well as allowing for a measure of portability (section 6.3.5.3).

This version of eSCARF has been upgraded to be compatible with Java v1.3.1 (otherwise known as Java 3). Deprecated Java methods have been replaced, for instance, stopping threads in a safer manner than calling thread.stop(). The earlier version of eSCARF was written for Java v1.2. Java 3 is also the version of Java that WebSphere has been written with and runs on. For a one-tier configuration, this gives the added benefit that a Java Runtime Environment is already installed, so that eSCARF can be run without installing further software.

Code Revision

Code was tidied up, and commenting was improved. This allows eSCARF to be more easily maintained, especially when it needs to be customised for different environments.

Installation Packaging

One of the implementation issues discussed in Ng and Wong (1999, section 4.4.2) was packaging the eSCARF software. While an installation wizard has not been created, the installation procedure has been greatly automated (see Appendix 4 for the installation procedure).

The Java code has been packaged into a JAR file, and the database setup has been automated with batch files. As a result, eSCARF has been made more portable. The need for users to use the command line to configure eSCARF has been reduced, thereby simplifying the install process. eSCARF can also be loaded with a batch file without the need to type in any commands or set any classpaths.

6.4.3. Database

eSCARF uses the IBM DB2 database server to store its data, as DB2 is a part of WebSphere Commerce. In section 4.4.4 of Ng and Wong (1999), the issue of database dependency is raised. Because of eSCARF's modular design, adapting eSCARF's database for another platform (such as Oracle or Microsoft SQL Server) does not require much modification. Database access is handled by the `scarf.db.DBConn` class, and some minor modifications to this class are all that is necessary for eSCARF to work with a different database server. In fact, this has already been achieved as Anandarajah and Lek (2000) have successfully enabled eSCARF to work with MySQL.

All the SQL commands in the code as they currently stand do not need to be updated, as they are all compliant with the ANSI 92 SQL standard, which is adhered to by all major database systems today. That is, no SQL statements proprietary to any specific database system are used.

The table structures and table relationships of the database are essentially unchanged. Some table and field names have been changed for improved clarity. The changes to the design of the database mainly reflect the requirements specification, and the need to capture a different set of data than that of the older eSCARF prototype which ran on Net.Commerce. This design change will be necessary if, for some e-commerce systems, the data capture requirements vary from the standard. See integration issues in section 6.5 for more details about this.

The data dictionary set out in appendix 2 gives a thorough description and explanation of the table structures and contents for this version of eSCARF.

Database Maintenance

Database backups are currently performed using the backup mechanism found in IBM DB2, instead of one found in eSCARF, as all of eSCARF's data is stored in a single DB2 database. A batch file is also supplied (`refreshdb.bat`, see appendix 5) to clean out and reset the eSCARF database if eSCARF needs to start afresh.

6.4.4 The Audit Hook

The audit hook (or embedded audit module) has the sole purpose of intercepting transactions from the e-commerce system, as they happen in real-time, and passing them on to the eSCARF system.

The WebSphere Commerce Programmer's Guide (IBM 2002, p.15) notes that, *"you must use Java to customize functionality. This is very different from the model that had been used in WebSphere Commerce Suite, Version 4.1 (and earlier versions of Net.Commerce) in which C++ and Net.Data macros were used for customization."* WebSphere uses task commands to carry out units of business logic. WebSphere's architecture conveniently provides a dummy task command called `ExtOrderProcess`. After the payment processing controller command has finished processing the order, it passes control to the `ExtOrderProcess` command, which, being a dummy command, does nothing. This task command is represented by the Java class `ExtOrderProcessCmdImpl`. We may extend on this class and provide our own implementation of it, that is, adding our audit hook there.

In order to get WebSphere to use our new, extended class, the table `CMDREG` is modified. This table contains a mapping of WebSphere commands to Java classes. Therefore, we update the entry for `ExtOrderProcess` to call our new class (`HookOrderProcessCmdImpl`).

Within this class, our hook's logic is stored. The hook retrieves the order's reference number, then proceeds to fetch details related to that order from WebSphere's database. Once it has gathered that information, it encodes it all into a single text string. This text string is then encrypted via a simple encryption algorithm[13]. The hook opens a TCP/IP connection to the eSCARF server and then transmits the encoded, encrypted string to it. The connection is closed, and the hook's task is complete.

When the hook captures a transaction, it also writes what it is doing into the standard output log file of the WebSphere Application Server. A sample excerpt from the log file output follows, showing the transmission of one transaction to the eSCARF server:

```
[10/3/02 12:38:50:593 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Running with ID: 10606
[10/3/02 12:38:50:625 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Connection Established
[10/3/02 12:38:50:640 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Merchant number acquired
[10/3/02 12:38:50:640 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Member login id number
      acquired
[10/3/02 12:38:50:640 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Item list acquired
[10/3/02 12:38:50:640 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Connecting to eSCARF server at
      127.0.0.1:10002
[10/3/02 12:38:50:656 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Connection established.
[10/3/02 12:38:50:656 GMT+11:00] 532548dc SystemOut U Scarf Hook -> Sending (Encrypted):
      order_no=10606&order_success=1&merchant_no=10051&login=252&product1=11109&quantity1=3&price1
      =2500~
```

Ideally, audit hooks should be integrated into a system at design time (Gul, Teoh and Andrew 1991). Given the extensible architecture of WebSphere, as opposed to Net.Commerce, it is now possible to develop the audit hook in parallel with the development of an e-commerce store. The issue was raised in section 3.4.1 in Ng and Wong (1999) and has now been addressed.

Changes

The audit hook mechanism has been entirely rewritten, due to the differences in architecture between WebSphere and Net.Commerce. Additionally, the login audit hook was removed, as this thesis is focusing only on detecting fraud related to e-commerce payment transactions.

The old eSCARF hook involved using a two-part hook, consisting of a Net.Commerce listener, and a SCARF client. The listener captured the order reference number as a transaction was placed with Net.Commerce. This reference number was communicated to the client, which then fetched additional information about the transaction. The client opened a single persistent connection with the eSCARF server and then relayed the transaction details to it. The new hook combines the functionality

[13] The encryption algorithm used is TEA. More information about TEA is available at this address: http://www.theorem.com/java/tea/index.html

of both of these parts into one. Additionally, instead of opening a persistent connection with the eSCARF server, each separate transaction opens a separate connection to the server, and closes it once the details have been transmitted.

The audit hook now also encrypts transaction details before sending them to the eSCARF server.

Please refer to integration notes (section 6.5) for a more technical summary of the process involved in creating an eSCARF audit hook.

6.4.5 eSCARF Server

The eSCARF server's role is to listen (on an auditor-specified TCP port) for transactions transmitted to it by audit hooks. Once it receives a transaction, it decrypts and decodes it, and then stores it in the eSCARF database (into the `OrderHistory` table). The transaction is then passed on to the Rule Checker module for further processing. The operation of the server is detailed in the server console log, which shows what the server is doing. The data displayed is fairly 'raw' and unformatted, as it appears for diagnostic purposes only. For example, an auditor can view the console log and monitor any technical difficulties eSCARF may encounter, such as if a drop out in the communications (network) link between it and the e-commerce system has occurred.

Changes

The option to specify the port the server listens on has been added. This is in case the default port eSCARF is configured to listen on (10002) conflicts with another application on the same system.

The new server has been changed to reflect the behaviour of the new audit hook's use of non-persistent connections. Once a connection is made with the hook, and the transaction's details have been received, the connection is terminated and the server goes back to waiting for another connection (instead of holding the connection open, waiting for further transactions along it). This change allows the eSCARF server to accept transactions from a variety of separate WebSphere systems (these systems may be identical systems mirrored for load balancing, or completely different systems representing different e-commerce stores). This allows one eSCARF server to handle multiple WebSphere systems, and centralises an auditor's records. A store_id variable is passed along with the transaction details to allow eSCARF to differentiate between transactions originating from different stores.

In correspondence to the addition of encryption in the audit hook, the eSCARF server now also includes a decryption algorithm for decrypting encrypted transaction details.

6.4.6 Rule Checking

The rule checker module's role is to process an incoming transaction, received from the eSCARF server, against the active ruleset, to attempt to detect fraud.

The audit strategy used is a variant of ripple-down rules, as employed by Ng and Wong (1999). A rule is composed of one node or several nodes. Nodes are connected into a nodal tree, with each node potentially being linked up to three others (a parent and two child nodes) via true and false branches. A node consists of several properties: an expression which can be evaluated as true or false, an alert level, and optionally, paths to a 'true node' and a 'false node'. Rules have a single node designated as a start node where processing begins from.

When the start node is read by the rule checker module, the expression it stores is evaluated. This expression is similar to what you would find in an `if (`*`expression`*`)` statement, composed of various transaction variables (such as price, quantity, credit card expiry date, etc.) and operators (=, >, <, etc.). These transaction variables may represent variables of the transaction being processed, or may be cumulative, such as representing the total value of goods purchased by a user, including previous transactions.

If the expression is evaluated as true, the rule is 'triggered'. The rule checker then responds to this, its action determined according to the alert level of that node (see below). Depending on the expression being evaluated as true or false, eSCARF will check if that node has a 'true path' or 'false path', respectively, linking it to another node. It will then read in that node in the same manner as the start node. Processing will recursively continue down the tree until reaching a node that does not have any child nodes.

The alert level of a node determines what eSCARF should do when an expression is evaluated as true (i.e.: when the rule is triggered). The alert level is an integer, with higher numbers representing a higher level of alert. Higher levels of alert are attributed to events of more significance and generally produce more visible alerts to the auditor.

The current alerts programmed into eSCARF include:

Alert Level	Action Taken
0	No action taken
1	Log transaction to database
2	Generate onscreen alert
3	E-mail an alert

These alerts are cumulative, meaning that an alert level of two will trigger the action of alert level two and one. Similarly, an alert level of three will trigger the action of alert level three, two and one. The pseudocode for this is:

```
if (alertlevel > 2) { generateEmailAlert }
if (alertlevel > 1) { generateOnscreenAlert }
if (alertlevel > 0) { logTransactionToDB }
```

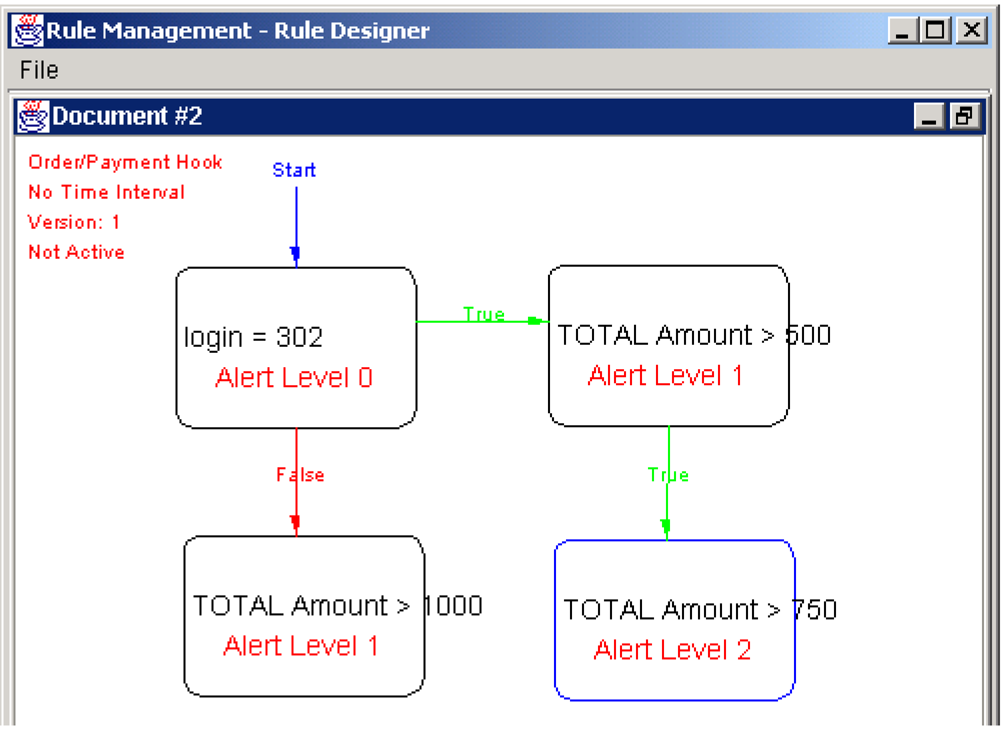
With a minor modification, some alert levels can be created so that they do not act in this cumulative fashion. For example, if an auditor wanted to add a special alert level of 100, which only generates an onscreen alert and does nothing else, the pseudocode would be as follows:

```
// First section for non-cumulative alerts
if (alertlevel = 100) { generateOnscreenAlert
} else {
   // Section for cumulative alerts
   if (alertlevel > 2) { generateEmailAlert }
   if (alertlevel > 1) { generateOnscreenAlert }
   if (alertlevel > 0) { logTransactionToDB }
}
```

The alert level scheme is customisable, such that additional levels may be programmed in for different actions. Examples of other actions that can be taken include: sending a short text message to the auditor, and initiating a procedure to block the transaction on the e-commerce system. This will require the auditor modifying the Java code of eSCARF and should take place during the time when eSCARF is being integrated with the e-commerce system. Pseudocode showing how an SMS alert could be implemented as a level 4 alert follows:

```
if (alertlevel > 3) { generateSMSAlert }
if (alertlevel > 2) { generateEmailAlert }
if (alertlevel > 1) { generateOnscreenAlert }
if (alertlevel > 0) { logTransactionToDB }
.
.
.
method generateSMSAlert {
   // code here that would open connection to SMS gateway,
   // and then transmit alert to a mobile phone
}
```

Note that a single rule may actually trigger several actions, as each node is individually assigned an alert level. The screen capture below shows an example of a rule as it appears in the Rule Management graphical user interface.



Please refer to integration notes (section 6.5) for a technical description of how an auditor may customise alert levels and their corresponding responses.

Sliding Window Time Mechanism

As in Ng and Wong's prototype, rules are also associated with a window of time (Ng and Wong 1999, section 3.4.2). This enables rules to have a temporal dimension to them. This window of time determines what transactions in the transaction log are available for the rule to access, therefore restricting the rule to using the most recent transactions made. For example, a rule with a window of seven days means that transactions up to seven days older than the transaction currently being processed will be considered in rule checking. In this way, the transaction variable `TOTAL PRICE` can represent the total price of all transactions made in the last seven days.

Changes

The most notable change is the introduction of alert levels, which are a more flexible and easily extensible way of dealing with rules that are triggered. (The old method provided two flags, one which logged the transaction to the database, and one which displayed an onscreen alert. To modify this, changes to the database schema would be required, in addition to numerous changes in the Java code.)

A minor change with the onscreen alerts was made where additional information was added to the alert, namely, the order reference number generating the alert, and the web report page where more information on it is viewable.

6.4.7 Rule Management

The rule management module, accessible via eSCARF's main menu, allows an auditor to create new rules, modify and delete existing rules. Rule management is entirely driven by a graphical user interface, opening up in its own window. From the file menu, the auditor can:

- create a new rule;
- open an existing rule;
- save currently open rules;
- change the name of the rule currently opened; and
- exit.

Rules open in their own sub-window. Information shown in this sub-window includes the rule's status (active or inactive), the rule's version, whether the rule is read-only (see Versioning, below) and the time interval the rule is applied over (see Sliding Window Time Mechanism, section 6.4.6). Actions are performed by right clicking to bring up a popup menu. A new rule is created without any existing nodes in it. Right-clicking in an empty area in the sub-window will provide the following options:

- add node – if this is the first node being added, it will also be designated as the rule's starting node. New nodes are blank, set with an alert level of 0;
- set time interval – this sets the time interval for the sliding window time mechanism (see section 6.4.6); and
- set active state – Open available for saved rules, this allows the auditor to activate or deactivate a rule.

Nodes are represented by boxes, in which are written the alert levels of the nodes, as well as the node expressions. Right clicking on a box provides the following options:

- set as start: sets the selected node as the start node;
- set rule: sets the expression for this node (see later);
- set alert level;
- add true path: adds a link to another node that eSCARF will go to if the expression evaluates as true, during processing;
- add false path: adds a link to another node that eSCARF will go to if the expression evaluates as false, during processing;
- delete node;
- delete true path; and
- delete false path.

Nodes are joined to each other via true paths and false paths. These are represented by lines with arrowheads. The arrowhead points to a child node. Right clicking on an arrow will give the auditor the option to delete the path.

Setting Node Expressions

Right clicking on a node and selecting “set rule” will bring up an expression builder dialog box. That dialog box is split into three sections:

- the left hand pane contains all the transaction variables available (such as “Quantity OF” and “TOTAL Quantity”);
- the right hand pane contains all the operators (such as =, >, <, AND, OR); and
- the text box at the bottom contains the actual expression.

Auditors build their expression by selecting variables and operators, and typing into the text field values as required. For example, if an auditor wishes to set up a node expression to trigger true when the user’s login name was “JohnDoe”, and if he orders any item costing more than $100, then the resulting expression will be:

```
login = JohnDoe AND ANY Price > 100
```

The values underlined represent values typed in by the auditor. Values representing prices must be given in dollars, even though they are stored as cents in the database.

An important feature of node expressions is that they are customisable. This is because auditors will often want to perform different, custom tasks with the transaction data that may be complex and proprietary to the e-commerce system. Therefore, when the current set of transaction variables and operators are insufficient, an auditor must be able to add his or her own in.

The way expressions are designed and parsed by eSCARF allows the expression language set to be extensible. The language that expressions are composed of can easily be extended to include a variety of tasks. This also allows eSCARF to interface with external sources to further explore a transaction’s validity. Examples of ways an auditor may extend the expression language set include:

- Integrating eSCARF with an Address Verification System (AVS). A call to an AVS could be made which will verify details of a shipping or billing address (eg: does the suburb and postcode match up?). The AVS will return a response which eSCARF can then use in its own processing.

- Calling external payment gateways to verify credit card details. For example, IBM provides a system called a CICS Transaction Gateway[14] that performs this task.
- Adding a simple variable `DIFF_CNTY` which compares the countries in the shipping and billing addresses, and returns *true* if they are different and *false* if they are the same.
- Calling a datamining module to analyse the transaction.

Please refer to integration notes (section 6.5) for a technical description of how an auditor may extend the expression language set.

Versioning

All rules in eSCARF have a version number, starting at one. Versioning allows the preservation of old, triggered rules. A rule, once triggered, becomes read only, and if that rule needs to be modified, a new version of it must be created. The old version of the rule becomes deprecated and is never used again, except when back referencing old transactions which triggered the older version of the rule. All versions of a rule, except the latest, are deprecated and will not be used by the eSCARF rule checker.

Changes

The Java code for this module was improved so that it was more conducive to being customised by the auditor for the purpose of extending the expression language set. A new transaction variable, NUM_CCS was also implemented, which counts the number of different credit cards a customer has used to purchase goods from the store in the past.

The term “rule” was changed to “expression”, when referencing a node’s expression. This terminology is clearer than that used by Ng and Wong (1999), as it will not be confused with the concept of rules (which contain nodes).

[14] www.ibm.com/software/ts/cics/ctg/

6.4.8 Rule Activator

The rule activator allows the auditor to set which non-deprecated rules are currently in the active ruleset, by flagging rules as active or inactive. The rule activator remains unchanged from the old version of eSCARF, except that the rule activator module's functions are now also accessible via the web reporting interface.

6.4.9 eSCARF Reporting

eSCARF's reporting capabilities use web pages in order to convey information. Because HTML only provides static web pages, a web scripting language must be used to return dynamic pages. As a result, eSCARF web reporting was written in Java Server Pages (JSPs), as WebSphere natively supports them as well. However, these pages can be easily modified to run in another languages, such as Active Server Pages (ASPs), PHP, or Cold Fusion Markup Language (CFML), if the machine eSCARF runs on supports these other languages.

Reports compose the end of the system where auditors view the output of eSCARF. The system provides reports that present this information to auditors in an easy-to-read form, with auditors being able to narrow down searches to items that particularly interest them.

There are several reports eSCARF can display, all of which will be detailed below.

Main Page and Dashboard View

This is the page an auditor will see first when web reporting is opened up in a web browser. This page provides a menu through which the rest of the reports can be accessed. This page also provides two summary reports which allow auditors to pick up important information at a quick glance, all on one screen – in effect, a broad overview of the current status of eSCARF.

Information on this page includes:

- transactions eSCARF has processed;
- the number of alerts triggered by these transactions;

- the number of rules in the active ruleset; and
- a listing of the names of active rules.

Also on the page is a listing of the last five alerts. Alerts of an alert level over 1 are highlighted in red. These alerts are listed by showing their alert level, date and time the alert occurred, the rule expression triggered and the user who made the transactions.

Transaction Summary Report

This report summarises in a table the details of alerts that have been triggered, sorted in reverse chronological order (most recent alerts first). Each row in the table represents one alert, and the following information is provided for each alert:

- alert ID number;
- transaction ID which triggered the alert;
- timestamp for when the alert occurred;
- order Reference number (WebSphere's order number for this transaction);
- alert level;
- name of the rule violated, together with rule ID, node ID and the node's expression;
- login ID or username of the person who made the transaction;
- total quantity of items ordered; and
- total price of the orders.

The columns alert ID, timestamp, and alert level can be clicked on to sort by them. Furthermore, transaction IDs and rule ID/node ID pairs are hyperlinked for cross-referencing purpose. By clicking on a transaction ID, the auditor can examine the specific details for that particular transaction (Transaction Detail Report). By clicking on a rule ID/node ID pair, the auditor can view details for a single rule, with the node ID specified, highlighted on that page (View Rule Report).

Transaction Detail Report

This report lists all the details for a single transaction:

- order Reference number (WebSphere's order number for this transaction);
- merchant number (the store ID where the order was made);
- login ID or username of the person who made the transaction;
- payment details (credit card type, number and expiry date); and
- listing of the items ordered (by product ID, quantity, price and subtotals).

At the bottom of the report are hyperlinks to cross reference this transaction by login, or by the last 20 transactions. Cross referencing by login will display all transactions made by that customer, so an auditor can view a customer's transaction history. Cross referencing by the last 20 transactions will show the auditor the last 20 transactions that occurred before the one currently being viewed.

Rule Management Page

The rule management page shows a listing of rules divided into four categories, rules that are:

- Active and triggered
- Active but untriggered
- Inactive
- Deprecated (old, disused versions of rules)

This page also provides the facility to activate and deactivate non-deprecated rules. Each rule is linked so that it can be viewed in more detail on the View Rule Report.

View Rule Report

This permits the auditor to view all the details of a rule, including its node tree. It essentially replicates the view shown in the eSCARF Rule Management module, where rules are designed. If a rule is being viewed by cross referencing it with an alert that triggered it, the node which caused the triggering is bolded for visibility.

Options Page

Here the auditor can set various miscellaneous options that modify how eSCARF operates. Currently, the following fields exist:

- eSCARF server port: determines which port the eSCARF server listens on for transactions passed to it by the audit hook. The eSCARF server must be restarted for changes to this option to take effect. (Audit hooks must also be reconfigured to send data to the new port.);
- e-mail alert address: For alerts which respond by sending the auditor an e-mail, this field defines where to send the address; and
- e-mail alert SMTP server: Defines which mail server to use if eSCARF needs to send out an e-mail.

Changes

The reporting functionality of eSCARF was considerably overhauled from the elementary functionality provided by Ng and Wong's (1999) prototype. This was necessary as web reporting is a crucial tool used by auditors to provide assurance (this was verified by the survey results in chapter 8).

Rule Management and Rule View Section

Now included in reports are the ability to view the ruleset, and view the details for individual rules. This was added because auditors often will need to reference rules after analysing the alerts generated and transactions monitored by eSCARF. Instead of having to switch out of the web browser to the Java interface, auditors can now view rules by staying within the web browser.

Cross Referencing Capabilities

With the introduction of rule management and rule viewing, alerts can be cross referenced with rules, and transactions can be cross referenced with other past transactions. These functions further aid the auditor's analysis of the data collected by eSCARF.

Ability to Activate Rules

This function of the Rule Activator module was duplicated in the web interface for the convenience of auditors.

Main Page and Dashboard View

In addition to giving the auditor access to all the other web reports, the main page now includes an assortment of useful summaries and details an auditor can view at a glance. Reloading this page periodically will provide the auditor with a good idea of the system's current status.

Graphical User Interface Revamp

The graphical user interface for the web reports was completely revamped in an aesthetic sense. A header at the top of the page provides a menu through which all major sections of the site can be reached. The design is clean and sharp, aiding auditors in finding the information they want without any unnecessary clutter to confuse them.

Usage of JSPs

The old version of eSCARF used Java servlets to serve reports. However, servlets are proprietary to the Java web architecture, and are not conducive to being ported to another web scripting language such as ASPs. JSPs on the other hand, are easily translated to these other languages.

6.5 Integration Considerations

Although this version of eSCARF has been customised to work with WebSphere Commerce, one of the important features of the eSCARF architecture is that it can be adapted to any type of e-commerce system. This adaptation process is the process of integrating eSCARF with an e-commerce system. This section reviews the considerations an auditor is required to make concerning system integration. As noted in the literature review, one of the requirements for continuous assurance systems to be effective is the auditor being knowledgeable about the subject matter being assured, as well as about the information system providing the continuous assurance. The latter refers to proficiency in the technical aspects of information systems. In the case of the eSCARF continuous assurance system, the auditor must have a good understanding of the e-commerce system, as well as Java, knowledge of which is required to customise eSCARF.

However, before eSCARF can be deployed in an organisation, planning for integration must occur, and the following issues must be thought about:

- What transaction data needs to be captured? An e-commerce system will provide a lot of data related to a transaction, and an auditor must decide how much of it is relevant to help detect fraud and thus required by eSCARF. Sometimes all the data will be desired, or sometimes only a subset of it;
- Where to insert the audit hook? The auditor requires a technical understanding of how and where a hook can be added to the e-commerce system;
- What types of alerts are required? Should the system be able to alert auditors by other methods besides logging to the database and onscreen alerts?;
- What audit strategy to implement?; and
- Whether eSCARF should be run in a one or two tier configuration.

The following sections give a description of how different aspects of eSCARF can be customised by the auditor to match his or her requirements.

Designing an Audit Hook

There are two main considerations to make when designing an audit hook: how to integrate it with the e-commerce system, and what data it should capture. As each e-commerce system is different, the methods by which the hook will gather the information will be different. In all cases, however, the hook must be placed after the point in time an order is finalised and submitted. For instance, some simple e-commerce systems use only web scripting languages, like ASPs, to do their order processing. The order data is submitted via an HTTP `Post` request, which is then received, interpreted and handled by another ASP page (as opposed to a controller command in the WebSphere architecture). A hook for this type of system should be placed within this ASP page, and could be written in ASP.

When the code for the hook is written, it will also need to be modified so that it collects all the data it requires. This version of eSCARF receives the order reference number, then queries the WebSphere store database for all the other data it requires. In the ASP example above, the hook may simply intercept and copy the variables passed to the page by the `Post` request, if that request contains all the data required concerning that transaction.

The eSCARF database must also be modified to contain fields that will store any extra variables that are captured. For example, if the shipping method is captured by the hook, then the `OrderHistory` table in the database should have a shipping method field to store this. The eSCARF server should also be modified accordingly to insert this extra data into the table field.

Ultimately, the hook needs to be able to generate a query string in the format below, which must be sent to the eSCARF server via a TCP connection:

```
variable_name_1=value_1& ... &variable_name_x=value_x~
```

The query string must be terminated by a tilde.

The audit hook also contains a 16-digit hexadecimal key that is used to encrypt query strings. This key must be identical to the key used for decryption in the eSCARF server, so the auditor must ensure that these two keys match.

Changing the Server Port

If the eSCARF server is changed to listen on a different port, the audit hook must be updated to send transaction data to the new port.

Rules: Customising the Expression Language Set

If the language set for eSCARF is to be extended, any new variables and operators should first be inserted into the expression builder dialog box (`scarf.rulemaker.SetRule`). The `scarf.rulemaker.RuleValidator` class must also be updated to include any new variables or operators in its syntax checking. The actual logic for the parsing of expressions is found in scarf.rulechecker.OrderRuleProcessor. For an example, we will assume the auditor wishes to call an external module that will validate whether a suburb belongs to a particular postcode. The variable will be called invalidSuburb. It takes no other arguments or operators. Expressions evaluate as true or false, so invalidSuburb will return true if the suburb does not belong to the postcode.

The rule parsing method, parseRule(StringTokenizer st), parses the expression tokens. To the bottom of this method, something like the following could be added

```
else if (token1.equals("invalidSuburb")) {
   results = checkSuburb();
}
```

When the rule checker encounters "invalidSuburb" in an expression, the checkSuburb() method will be called. This method must be added to the class, and will handle all the necessary logic, including retrieving the suburb and postcode information from the eSCARF database, and then validating the suburb/postcode combination through some external source (like a mailing directory):

6.6 Implementation Notes, Limitations and Further Avenues for Development

The following list contains considerations made *before* the evaluation survey was undertaken. The results of the survey in chapter 8 show a much more thorough listing of issues to be considered, as provided by the auditor participants.

6.6.1 Security Issues

Transactions often contain highly sensitive and confidential information (for instance, credit card numbers). Because of this, any system which handles such transactions must have the relevant security measures built in. There are three areas where security would be especially important, namely, the TCP connection between the audit hook and eSCARF server in a two-tier configuration, the eSCARF database and the web reports.

In a two-tier configuration, because data passes through a network, there is a chance that this data could be intercepted by a third party. The risk of this increases if the eSCARF system is further away from the e-commerce system (transaction data must pass through more intermediate hosts to travel between the origin and destination). Currently, eSCARF addresses this security risk by performing simple private key encryption on this data. This elementary level of security can easily be fortified. For example, a secure protocol running over TCP/IP could be employed to transfer the data, such as SSL. Another example is how a public key encryption scheme could be employed in place of the current private key one.

The eSCARF database may contain credit card details, and as such, these should be protected. Again, encryption can be used to do this. Other alternatives include moving sensitive portions of transactions to another location that may be referenced by the eSCARF computer, but not any other computer networked to it.

The web reports run on a web server which may be openly accessible. Therefore, access to the reports must be strictly controlled and limited to authorised users (auditors). Some sort of password protection should be employed to ensure this. Also, because information viewed over the web is subject to being intercepted by third parties, the communications channel between web browser and the eSCARF web reports should be encrypted, perhaps via SSL.

Finally access to eSCARF should be secured as well, lest the ruleset be modified, or the server be started or stopped by an unauthorised party. Requiring a password to access eSCARF, as well as an access log of failed and successful log on attempts are two measures that will address this.

Naturally, physical security of both systems must also be ensured, if the above measures are to work.

6.6.2 Limited Set of Transaction Details Captured

The current version of eSCARF, being a proof of concept model, did not capture all the transaction details possible that could be used to detect fraud. Information such as shipping details, billing and shipping addresses, IP addresses and taxation data were all available, but not used by eSCARF.

6.6.3 Audit Strategy

The current audit strategy employed is fairly rudimentary, however, its main benefit is that it is extensible. More modules, in the form of Java classes, or calls to external applications, can be bolted on to it. It is envisioned that in future, effective fraud auditing strategies can be developed in other research and then integrated with eSCARF.

For example, datamining algorithms could be implemented with eSCARF, as Anandarajah and Lek (2000) have done. Many neural networking technologies also exist, which evaluate transactions and allocate to them a fraud score indicating fraud risk. Both the neural networking algorithms and fraud scoring concept could be integrated into the rule checker/designer parts of eSCARF.

In addition, the expression language set may be extended further to include some other fairly common variables and checks, such as if two different users use the same credit card, and the number of orders a user has placed.

6.6.4 Performance Impacts

Because eSCARF runs concurrently with an e-commerce system, its operation will impact upon the performance of the e-commerce system. The extent of the impact on performance requires evaluation in order to establish how scalable eSCARF is in comparison to the level of transaction traffic the e-commerce system handles. Therefore a performance testing of eSCARF is an avenue for future research.

6.6.5 Generalisation of eSCARF

Although eSCARF is currently designed to look at payment fraud, there is no reason why it cannot be generalised to analyse any type of transaction. All that is required is that the audit strategy and expression language set be tailored to match the type of transactions being audited. In our case study, we will examine identity fraud in an organisation, and how the transactions undertaken are processed and examined for possibilities of identity fraud.

6.7 Conclusions

This chapter has provided comprehensive documentation of the design of eSCARF, which was developed using a waterfall systems design life cycle. Firstly, the requirements for eSCARF as a continuous assurance system were mapped out. eSCARF was divided into several functional modules, each with their own required sets of responsibilities and functionalities. A design overview explored the IBM WebSphere Commerce system in order to understand how eSCARF could interface with it. Figure 5 provides an architectural overview of WebSphere. The conceptual design examined the implementation of each of the eSCARF modules (the audit hook, rule checker and alerts, rule management, rule activator, reporting and server modules) and how they all fit together architecturally with each other, and with WebSphere (figure 6). In the integration considerations section, methods by which eSCARF may be integrated into other e-commerce systems were detailed, such as designing an audit hook and extending the expression language set. Finally, limitations of this version of eSCARF were noted. These included a lack of performance data on eSCARF, security issues and the limitations of a basic audit strategy. These limitations should be used in the future development of eSCARF.

This chapter has therefore satisfied RQ3 by providing a thorough description of how a continuous assurance system functions, both internally, and when interfaced with an e-commerce system.

Chapter 7. System Testing

This chapter documents the system testing of eSCARF to ensure its correct post-implementation operation.

7.1 Aims

The primary aim of the testing stage is to verify the correct operation, as defined by the requirements specification, of the eSCARF system. This involves demonstrating that the system can:

- Correctly interface with WebSphere.
- Detect instances of fraud, given a ruleset and set of transactions.

The testing will also display eSCARF's robustness and ability to employ complex rulesets to detect patterns of fraud.

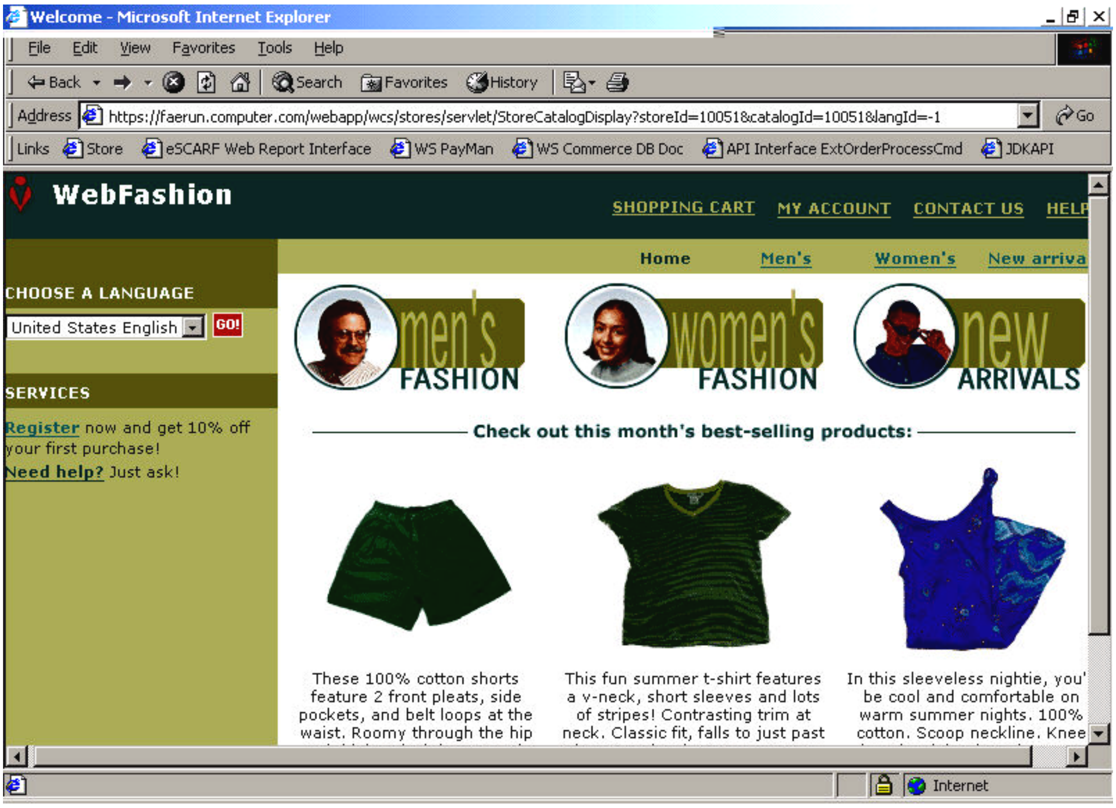
These tests do not constitute an exhaustive test of eSCARF's capabilities, but look at the system's expected outputs in terms of the given inputs. For internal testing, since the system was a modified version of Ng and Wong's prototype, the tests Ng and Wong (1999) ran were duplicated to ensure that the system still responded as expected (where the functional requirements of the old and new system were still the same). Testing of user input fields was also performed, to ensure that users could not enter erroneous data (syntactically incorrect or illogical data) that would cause the system to malfunction, such as entering a letter in a numeric-only field.

7.2 Test Methodology

The test environment used was selected to simulate a typical B2C e-commerce store running on WebSphere. The store used was a sample store provided by WebSphere Commerce called 'WebFashion', focusing on clothing sales. WebFashion came with a clothing catalogue divided into men's clothing, women's clothing and clothing accessories. The goods range in price from $25 to $100, and the expected demographic for visitors to this site are adults looking for everyday clothing wear for personal use.

Visitors to this site may order goods with or without creating themselves a user account within the system (registration). The registration process assigns the user an identifier which is linked to that person whenever they return to the store in future and log on as themselves. A registered user is identified by their e-mail address, whereas an unregistered user is assigned a unique identification number in lieu of an e-mail address.

Two testing scenarios were used. The first was a simple test designed to ensure eSCARF was operating correctly. As a result, this test also includes screen captures to show the visual flow of events. The second included a more complex ruleset and larger series of transactions and was designed to see how eSCARF would handle detecting fraud under these conditions.



https://faerun.computer.com/webapp/wcs/stores/servlet/StoreCatalogDisplay?storeId=10051&catalogId=10051&langId=-1

Register now and get 10% off your first purchase!

Need help? Just ask!

Check out this month's best-selling products:

These 100% cotton shorts feature 2 front pleats, side pockets, and belt loops at the waist. Roomy through the hip

This fun summer t-shirt features a v-neck, short sleeves and lots of stripes! Contrasting trim at neck. Classic fit, falls to just past

In this sleeveless nightie, you' be cool and comfortable on warm summer nights. 100% cotton. Scoop neckline. Knee

A screen capture of the WebFashion e-commerce store.

7.3 Tests

The preliminary setup for eSCARF was a fresh installation with an empty eSCARF database. The eSCARF server listening port was left on the default setting of 10002.

7.3.1 Test Scenario 1

Test scenario one consists of a simple one rule, one transaction test to demonstrate correct basic operation of eSCARF, as well as to visually show a typical auditing workflow through screen captures.

7.3.1.1 Inputs

RULES						
Rule ID	*Node*	*Expression*	*Alert Lev*	*Time Period*	*TrueN*	*FalseN*
A	A1	`Any QUANTITY > 3`	2	Unlimited	-	-

TRANSACTIONS						
ID	*Login*	*Credit Card Details*	*Item Name*	*Price*	*Quantity*	*Item Subtotal*
1	joe@blog.com	VISA; Exp: 10/04 #0000000000000000	Men’s Pleated Shorts	$25.00	4	$100.00

7.3.1.2 Expected Results

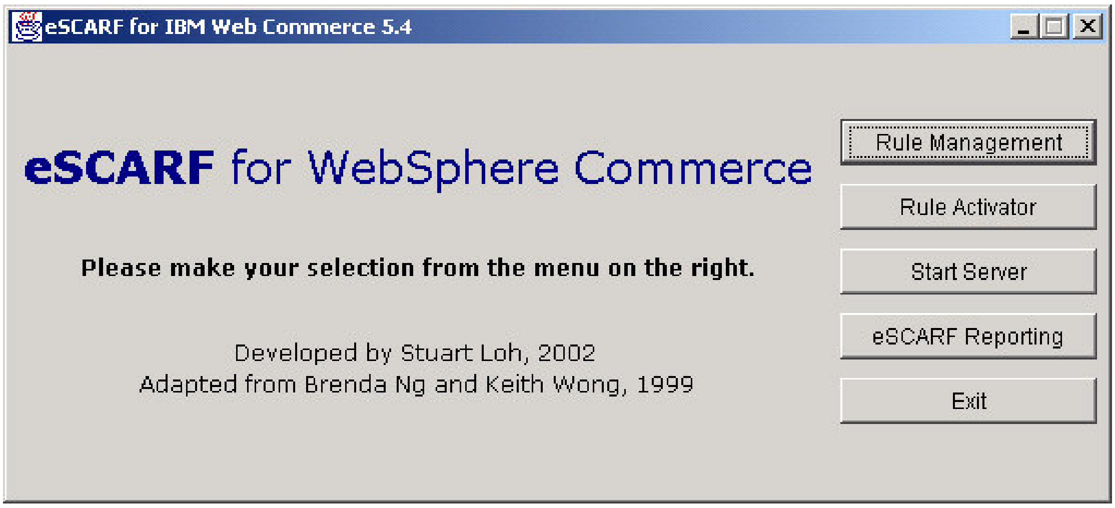
After transaction 1 is submitted by the customer:

- The transaction should be logged to the OrderHistory table.
- This transaction should trigger an alert by rule A, since the transaction matches the expression in Node A1.
  - An onscreen alert should be displayed, alerting of this fact.
  - The transaction should be logged to OrderLog table.
- The results should be viewable in the web reports.

7.3.1.3 Results

The following series of steps illustrate this test being carried out.

1. eSCARF was started up by double-clicking on `runscarf.bat`. This caused the eSCARF main menu to load up:



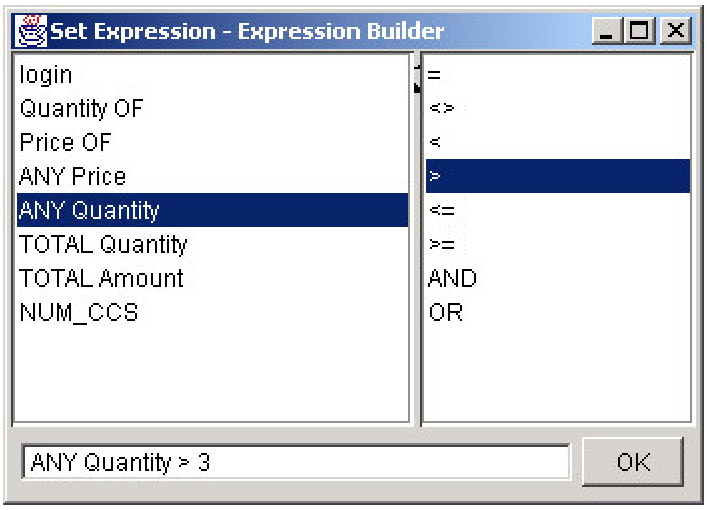
eSCARF for IBM Web Commerce 5.4

eSCARF for WebSphere Commerce

Please make your selection from the menu on the right.

Developed by Stuart Loh, 2002
Adapted from Brenda Ng and Keith Wong, 1999

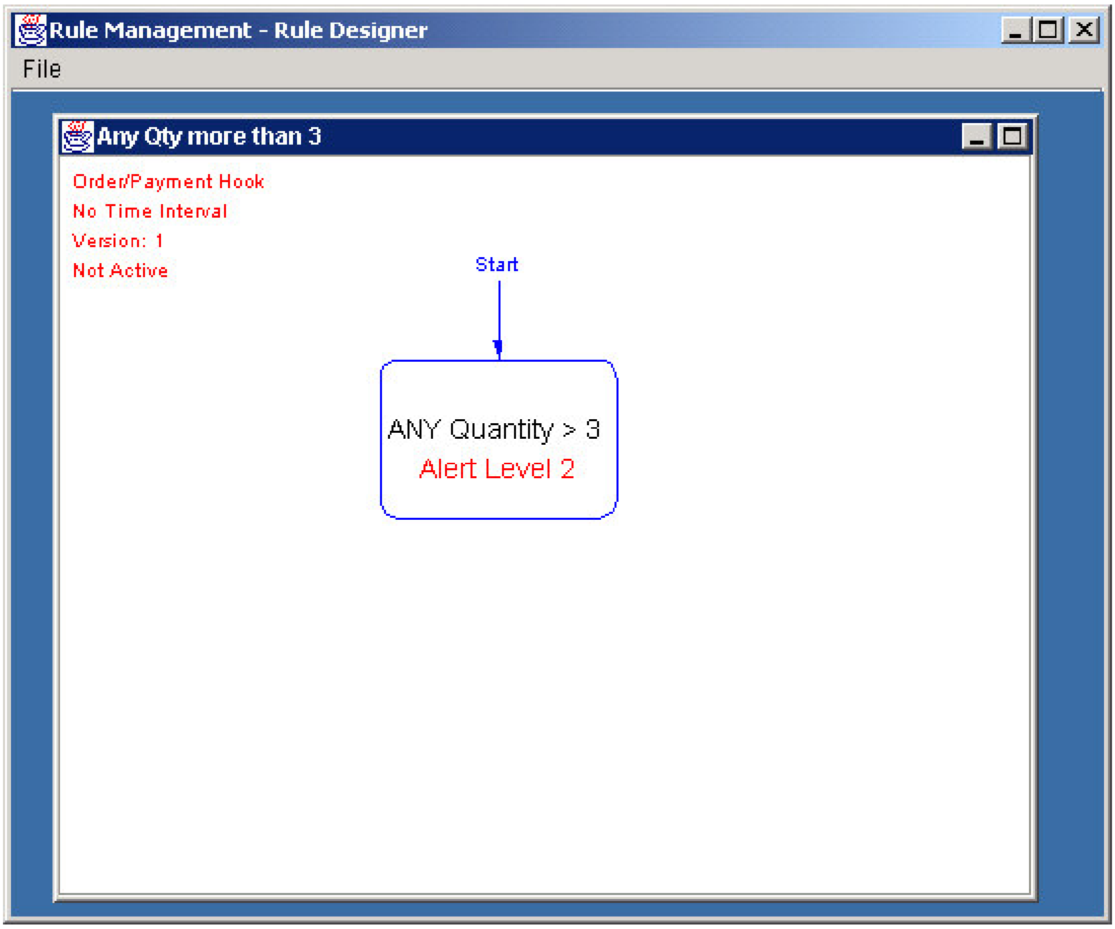
2. The Rule Management module was loaded by selecting that option from the main menu. A new rule with a single node was created. That node's alert level was set to 2, and its expression was set to `ANY Quantity > 3`, as below. The alert level of 2 means that if the expression is evaluated as true, then an onscreen alert will be produced to notify us of that fact.



Set Expression - Expression Builder

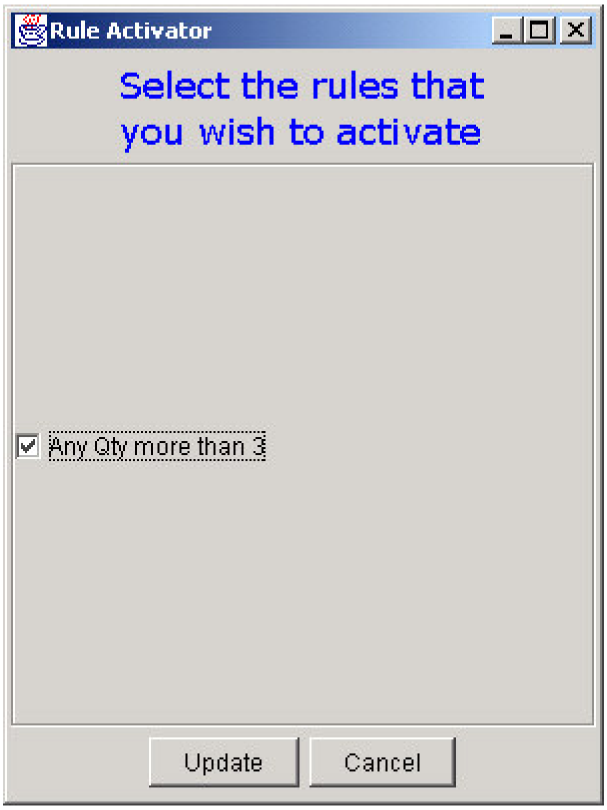
ANY Quantity > 3

3. The rule was saved under the name "Any Qty more than 3":



Order/Payment Hook
No Time Interval
Version: 1
Not Active

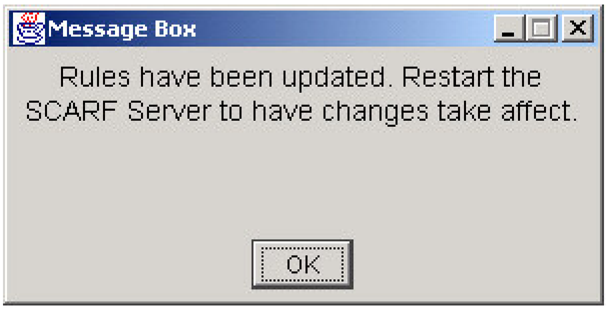
4. Rule management was closed, and the Rule Activator module opened. The new rule was ticked, to signify that it should be made active, and then the update button was clicked to confirm this:



Select the rules that you wish to activate

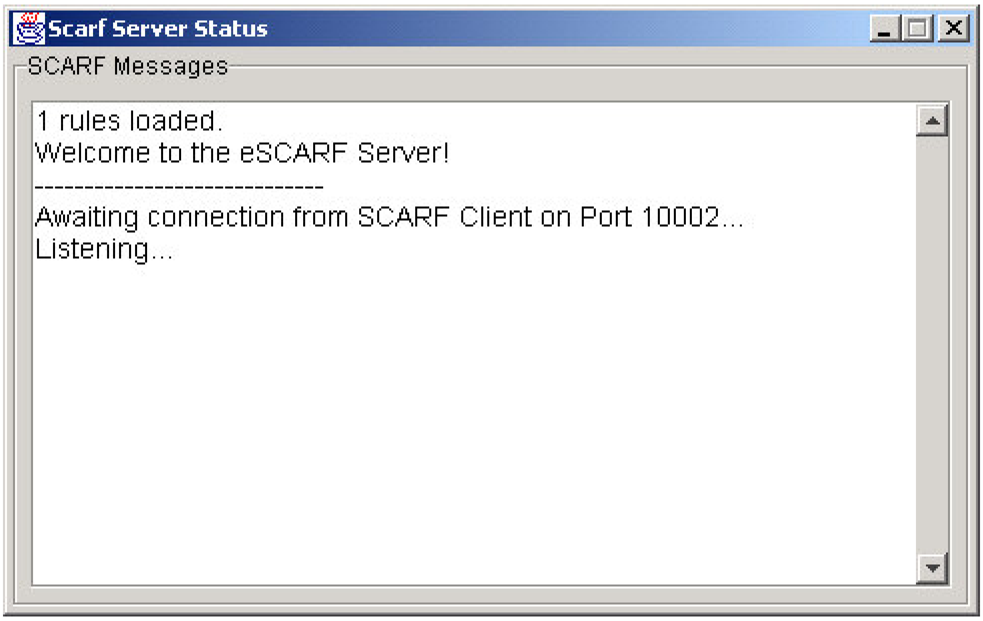
☑ Any Qty more than 3

5. eSCARF returned a dialog box notifying that the eSCARF server should be restarted (if it was already in operation), for the changes to take place:



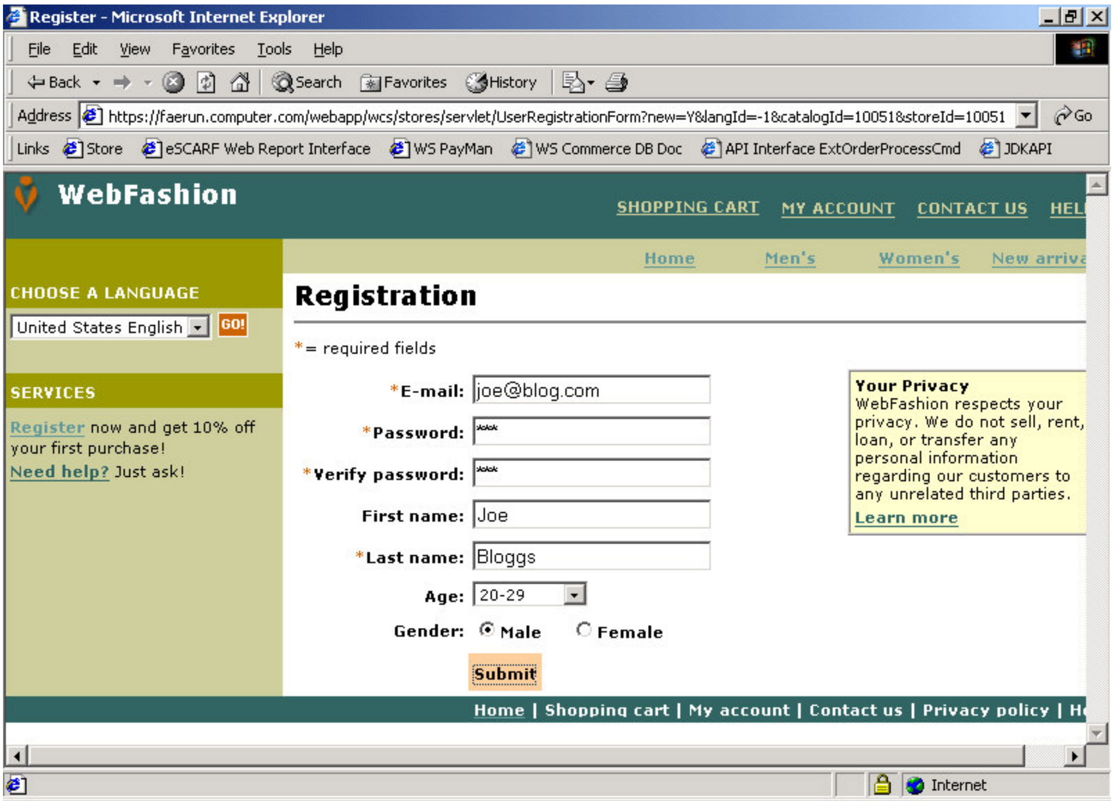
Rules have been updated. Restart the SCARF Server to have changes take affect.

6. Back in the main menu, “Start Server” was clicked, bringing up the server console window. It confirms that the one rule we entered in step 3 is currently active and being used by eSCARF. The server then starts listening for transactions on port 10002.



```
1 rules loaded.
Welcome to the eSCARF Server!
------------------------------
Awaiting connection from SCARF Client on Port 10002...
Listening...
```

7. Switching now to the customer's view of things, we enter the WebFashion web store via the web browser and register a new customer with details as follows:


Register - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address https://faerun.computer.com/webapp/wcs/stores/servlet/UserRegistrationForm?new=Y&langId=-1&catalogId=10051&storeId=10051
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
WebFashion
SHOPPING CART MY ACCOUNT CONTACT US HEL
Home Men's Women's New arriva
CHOOSE A LANGUAGE
United States English
GO!
SERVICES
Register now and get 10% off your first purchase!
Need help? Just ask!
Your Privacy
WebFashion respects your privacy. We do not sell, rent, loan, or transfer any personal information regarding our customers to any unrelated third parties.
Learn more
Submit
Home | Shopping cart | My account | Contact us | Privacy policy | H
Internet

Registration

*= required fields

*E-mail: joe@blog.com

*Password: ***

*Verify password: ***

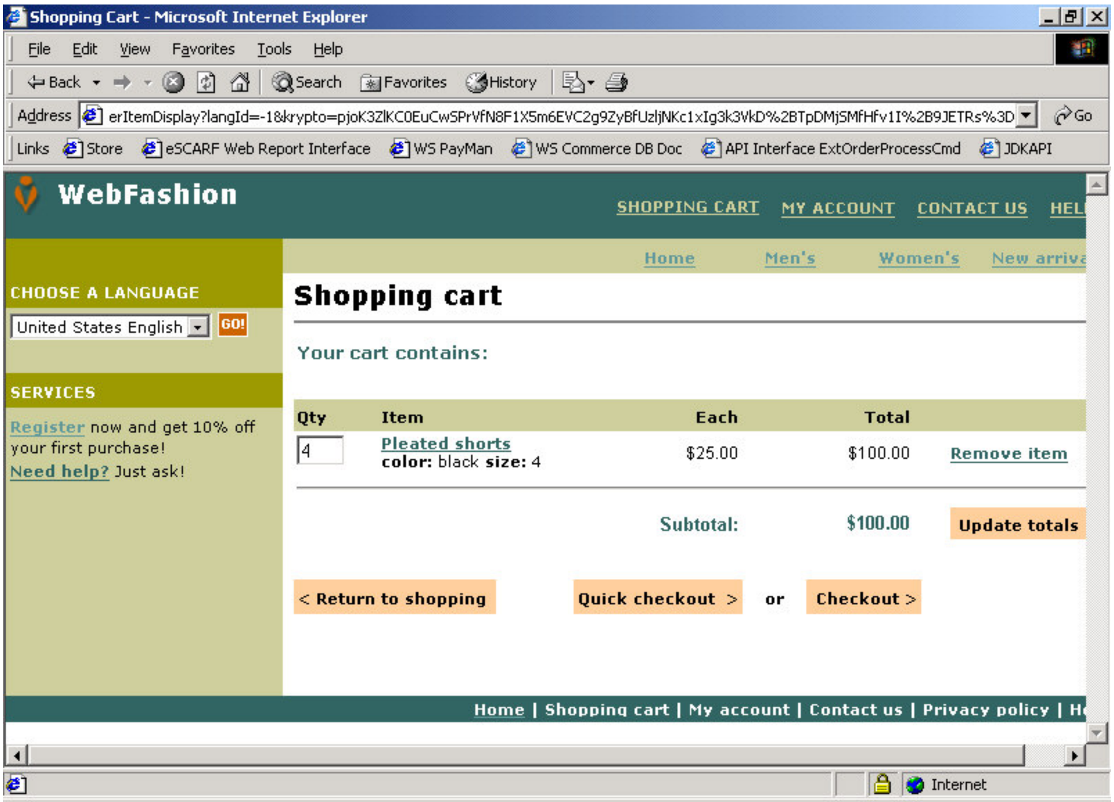
First name: Joe

*Last name: Bloggs

Age: 20-29

Gender: ● Male ○ Female

8. After account creation, the customer is now logged onto the store under his own account. From the store catalogue, 4 pairs of "men's pleated shorts" are added into the shopping cart:


Shopping Cart - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
WebFashion
SHOPPING CART MY ACCOUNT CONTACT US HEL
Home Men's Women's New arriva
CHOOSE A LANGUAGE
GO!
SERVICES
Remove item
Update totals
< Return to shopping
Quick checkout >
or
Checkout >
Home | Shopping cart | My account | Contact us | Privacy policy | H
Internet

Address: erItemDisplay?langId=-1&krypto=pjoK3ZlKC0EuCwSPrVfN8F1X5m6EVC2g9ZyBfUzljNKc1xIg3k3VkD%2BTpDMjSMfHfv1I%2B9JETRs%3D

United States English

Register now and get 10% off your first purchase!
Need help? Just ask!

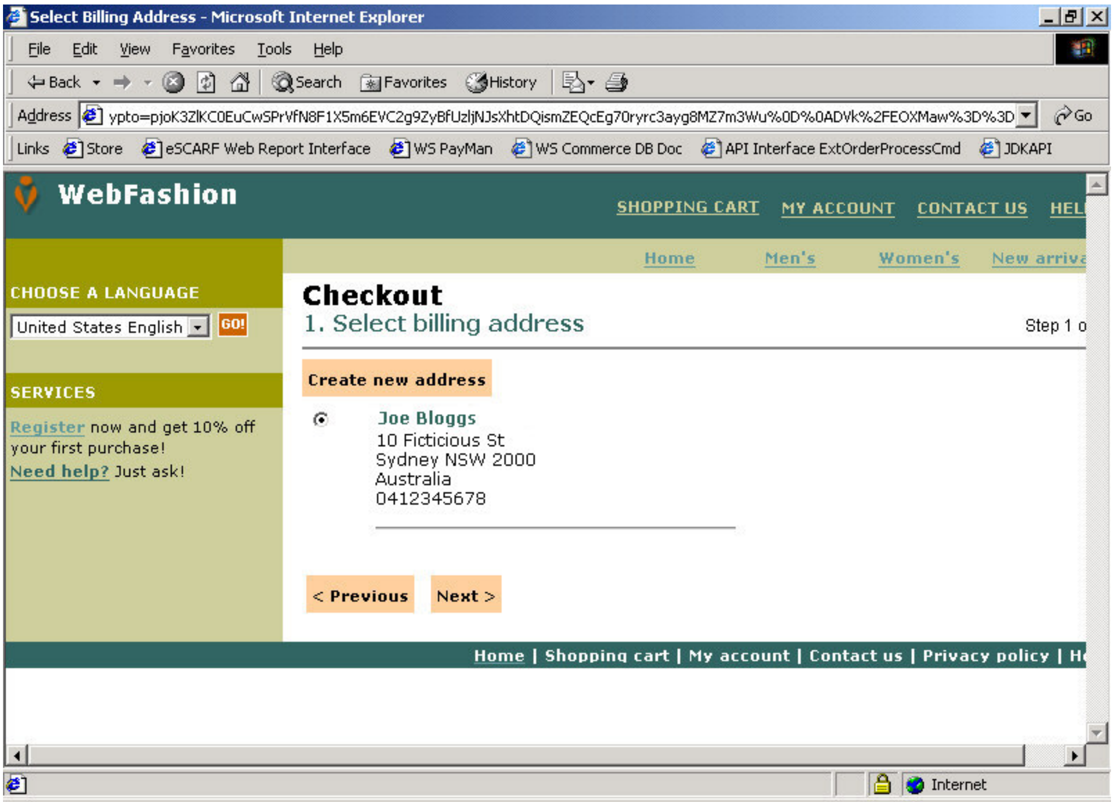
Shopping cart

Your cart contains:

Qty	Item	Each	Total
4	Pleated shorts color: black size: 4	$25.00	$100.00

Subtotal: $100.00

9. “Checkout” is clicked to begin the checkout process, the first three steps of which are to select a billing and shipping address, and a shipping method:


Select Billing Address - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
WebFashion
SHOPPING CART MY ACCOUNT CONTACT US HEL
Home Men's Women's New arriva
CHOOSE A LANGUAGE
GO!
SERVICES
Create new address
< Previous
Next >
Home | Shopping cart | My account | Contact us | Privacy policy | H
Internet

Address: ypto=pjoK3ZlKC0EuCwSPrVfN8F1X5m6EVC2g9ZyBfUzljNJsXhtDQismZEQcEg70ryrc3ayg8MZ7m3Wu%0D%0ADVk%2FEOXMaw%3D%3D

United States English

Register now and get 10% off your first purchase!
Need help? Just ask!

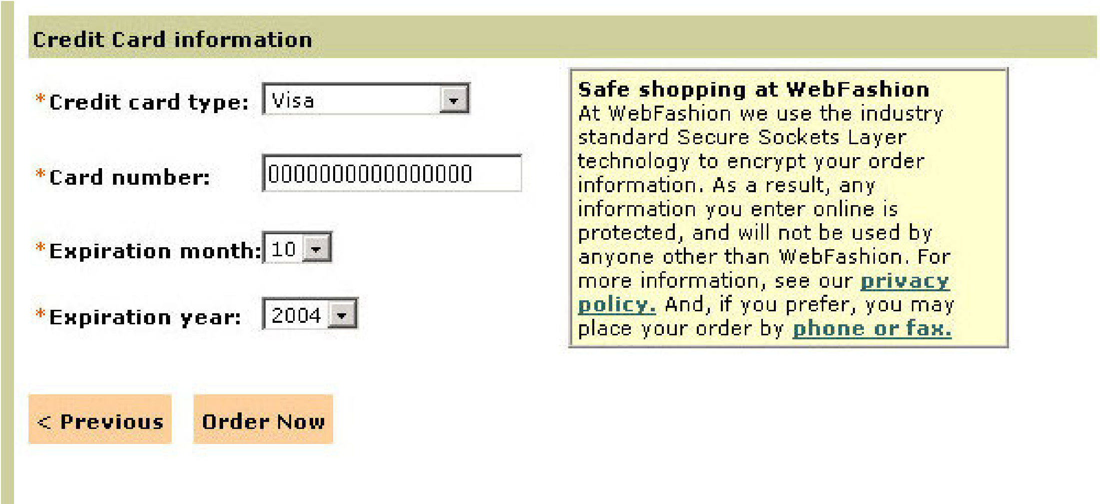
Checkout

1. Select billing address

Step 1 o

● Joe Bloggs
10 Ficticious St
Sydney NSW 2000
Australia
0412345678

10. The final screen in the checkout procedure is entering payment details. The following details were added, and then “Order Now” was clicked:



Credit Card information

*Credit card type: Visa

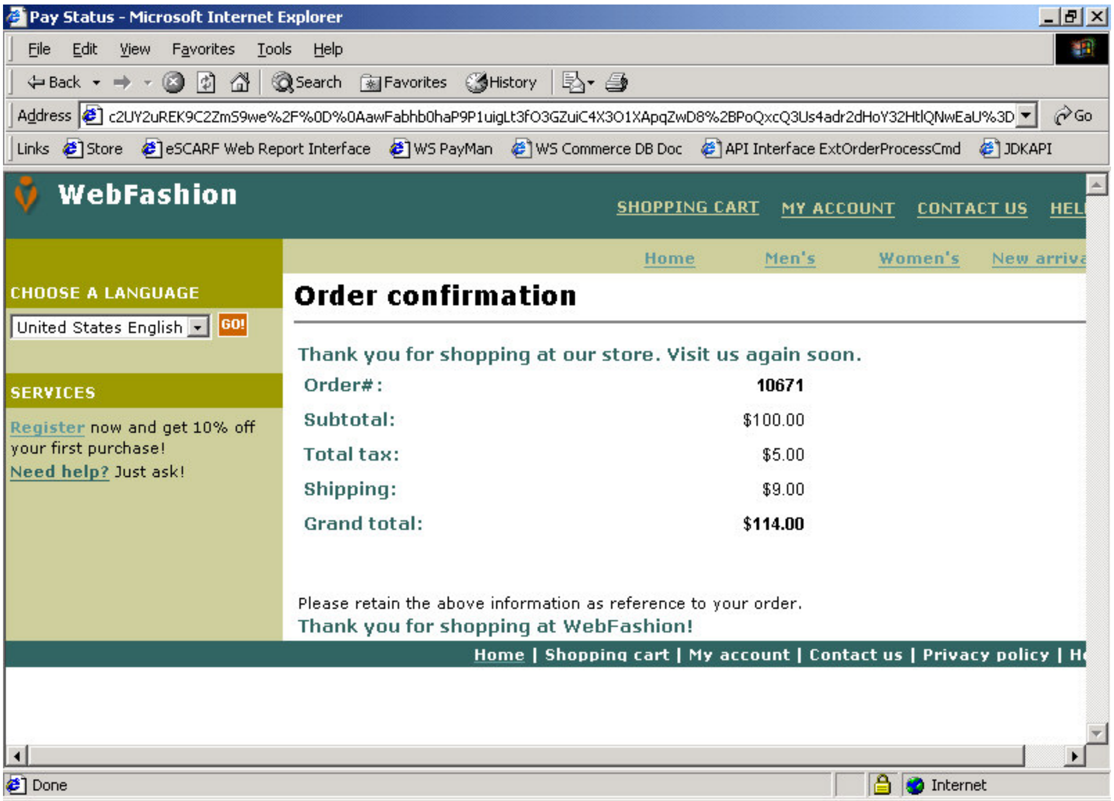
*Card number: 0000000000000000

*Expiration month: 10

*Expiration year: 2004

Safe shopping at WebFashion
At WebFashion we use the industry standard Secure Sockets Layer technology to encrypt your order information. As a result, any information you enter online is protected, and will not be used by anyone other than WebFashion. For more information, see our privacy policy. And, if you prefer, you may place your order by phone or fax.

11. “Order Now” finalises and submits the order, the necessary order placement processing occurs, and the following order confirmation screen is displayed:



Pay Status - Microsoft Internet Explorer

Address c2UY2uREK9C2ZmS9we%2F%0D%0AawFabhb0haP9P1uigLt3fO3GZuiC4X3O1XApqZwD8%2BPoQxcQ3Us4adr2dHoY32HtlQNwEaU%3D

Register now and get 10% off your first purchase!
Need help? Just ask!

Order confirmation

Thank you for shopping at our store. Visit us again soon.

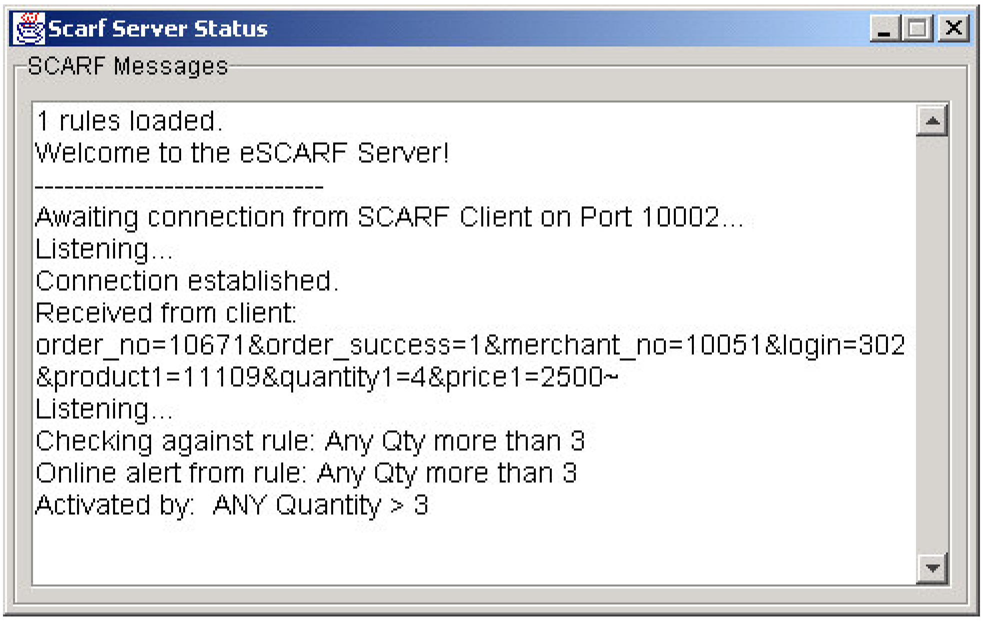
Order#:	10671
Subtotal:	$100.00
Total tax:	$5.00
Shipping:	$9.00
Grand total:	$114.00

Please retain the above information as reference to your order.
Thank you for shopping at WebFashion!

Home | Shopping cart | My account | Contact us | Privacy policy | H
Done
Internet

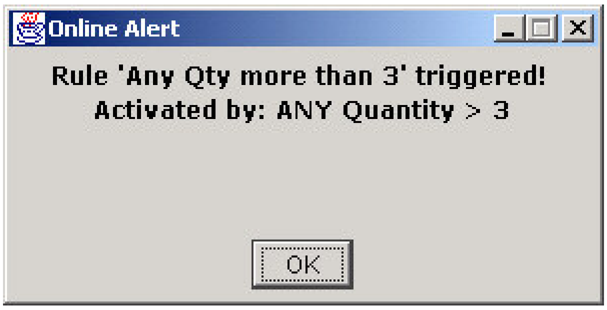
12. Between steps 10 and 11, WebSphere will run its order processing controller command, which should include a call to our audit hook task command. The audit hook should open a connection to the eSCARF server and send the transaction’s

details to it. Switching back to the eSCARF server console, we can indeed verify that this has happened:



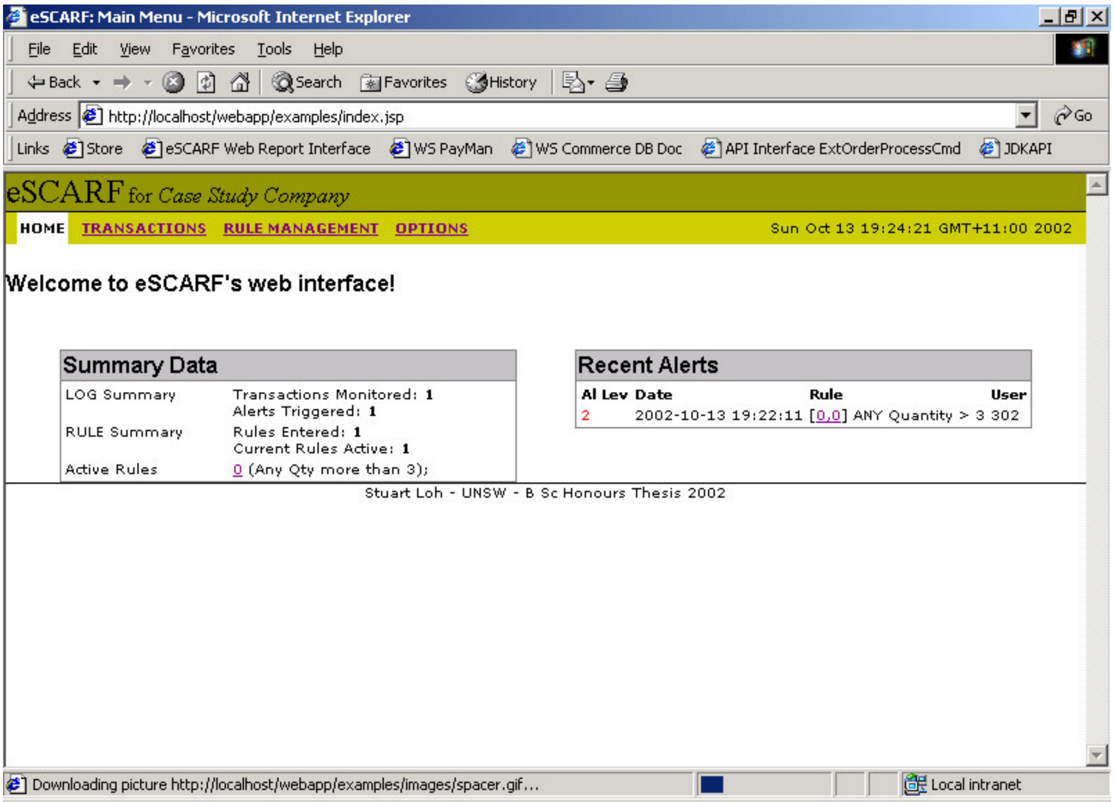
```
1 rules loaded.
Welcome to the eSCARF Server!
-----------------------------
Awaiting connection from SCARF Client on Port 10002...
Listening...
Connection established.
Received from client:
order_no=10671&order_success=1&merchant_no=10051&login=302
&product1=11109&quantity1=4&price1=2500~
Listening...
Checking against rule: Any Qty more than 3
Online alert from rule: Any Qty more than 3
Activated by:  ANY Quantity > 3
```

13. Additionally, we also encounter an onscreen alert that our rule added in step 3 has been triggered, along with the expression that triggered it. This confirms proper operation of our rule checker and alert level of two (the onscreen alert).



Rule 'Any Qty more than 3' triggered!
Activated by: ANY Quantity > 3

14. Switching now to the reporting interface of eSCARF, we can view and analyse the data eSCARF has captured. This should also verify that the alert generated above was logged to the database. The starting page of the web reports has a dashboard type view. In the summary data section, we see evidence of our recent transaction:


eSCARF: Main Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
Go
Local intranet

Address http://localhost/webapp/examples/index.jsp

eSCARF for *Case Study Company*

HOME TRANSACTIONS RULE MANAGEMENT OPTIONS Sun Oct 13 19:24:21 GMT+11:00 2002

Welcome to eSCARF's web interface!

Summary Data

LOG Summary	Transactions Monitored: **1** Alerts Triggered: **1**
RULE Summary	Rules Entered: **1** Current Rules Active: **1**
Active Rules	0 (Any Qty more than 3);

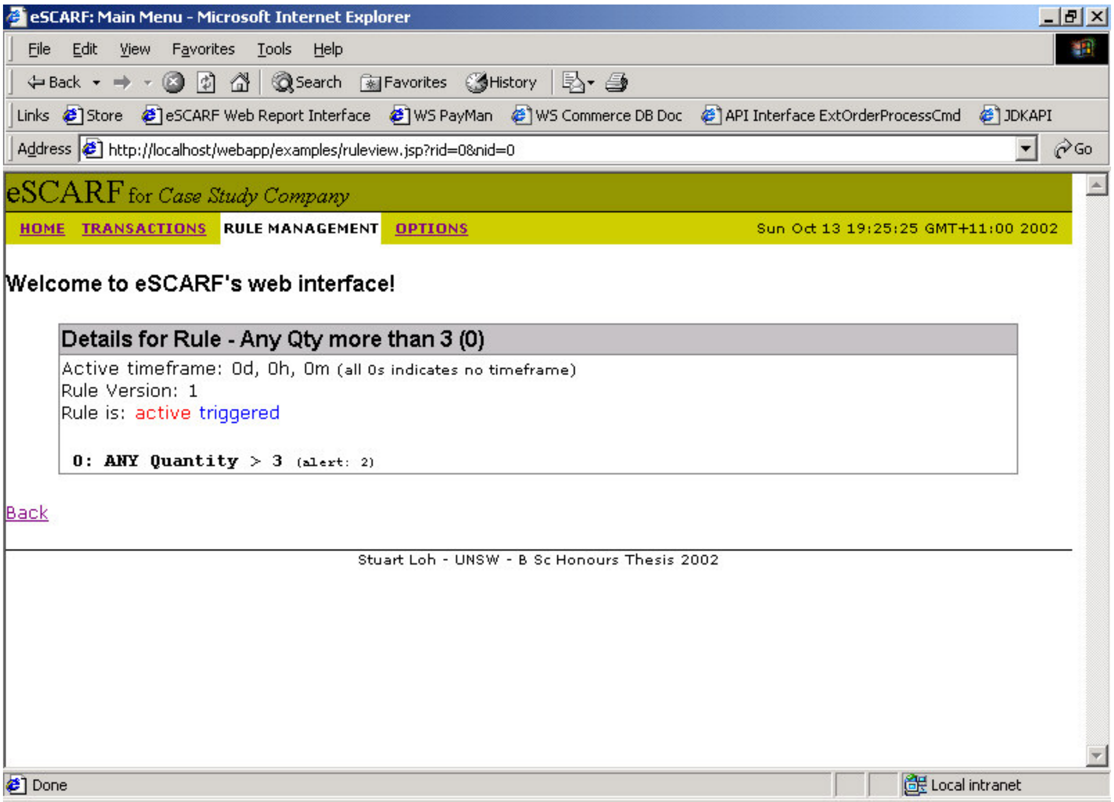
Recent Alerts

Al Lev	Date	Rule	User
2	2002-10-13 19:22:11	[0,0] ANY Quantity > 3	302

Stuart Loh - UNSW - B Sc Honours Thesis 2002

Downloading picture http://localhost/webapp/examples/images/spacer.gif...

15. In the “recent alerts” box, the hyperlink on `[0,0]` can be clicked to view the rule and node the alert corresponds to. The first number in the pair refers to the rule ID, and the second refers to the node ID. The node ID, 0, is highlighted in bold:


eSCARF: Main Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
Go
Local intranet

Address http://localhost/webapp/examples/ruleview.jsp?rid=0&nid=0

eSCARF for *Case Study Company*

HOME TRANSACTIONS RULE MANAGEMENT OPTIONS Sun Oct 13 19:25:25 GMT+11:00 2002

Welcome to eSCARF's web interface!

Details for Rule - Any Qty more than 3 (0)

Active timeframe: 0d, 0h, 0m (all 0s indicates no timeframe)
Rule Version: 1
Rule is: active triggered

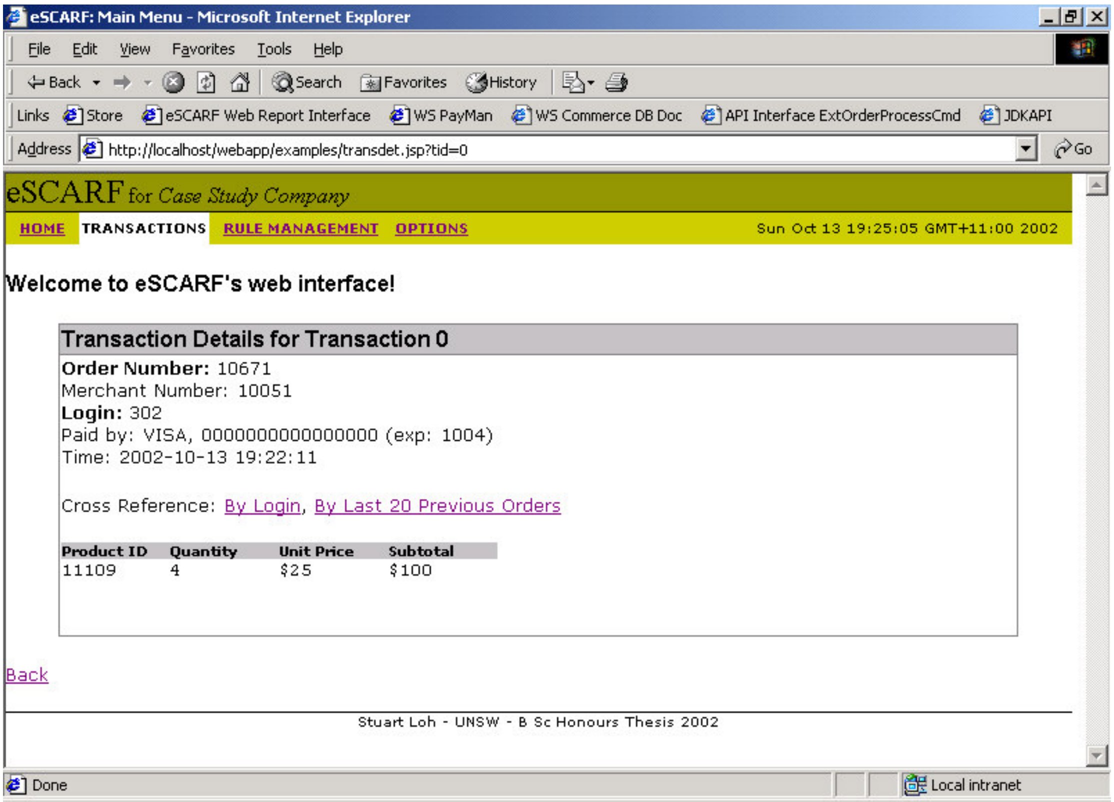
0: ANY Quantity > **3** (alert: 2)

Back

Stuart Loh - UNSW - B Sc Honours Thesis 2002

Done

16. eSCARF also can show details for the transactions processed. Shown below is the detail view for the transaction made in step 10. Product ID is WebSphere's ID number for men's pleated shorts, and the login ID of 302 is WebSphere's ID number for the user registered in step 6.


eSCARF: Main Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
Address http://localhost/webapp/examples/transdet.jsp?tid=0
Go
Done
Local intranet

eSCARF for *Case Study Company*

HOME TRANSACTIONS RULE MANAGEMENT OPTIONS Sun Oct 13 19:25:05 GMT+11:00 2002

Welcome to eSCARF's web interface!

Transaction Details for Transaction 0

Order Number: 10671
Merchant Number: 10051
Login: 302
Paid by: VISA, 0000000000000000 (exp: 1004)
Time: 2002-10-13 19:22:11

Cross Reference: By Login, By Last 20 Previous Orders

Product ID	Quantity	Unit Price	Subtotal
11109	4	$25	$100

Back

Stuart Loh - UNSW - B Sc Honours Thesis 2002

7.3.1.4 Conclusion

eSCARF worked as expected in this test scenario.

7.3.2 Test Scenario 2

Test scenario two consists of a more complex ruleset than scenario one. It is a simplified simulation of how a real store may work, and is aimed at demonstrating the capability of eSCARF's rulesets, as well as the ability for eSCARF to adapt to events as time passes. For this scenario, the values of certain products in the WebFashion catalogue were modified as WebFashion defaults all items to costing $25 (see inputs for values of goods).

This scenario will be tested using a parallel testing method (Simnett and Gay 2000). The input data will first be processed manually to determine a set of expected results. The input data will then be entered into eSCARF and the output recorded. The two sets of results are then compared to verify whether eSCARF is operating as expected.

7.3.2.1 Inputs

Inputs for this test will occur in two stages. An initial set of rules will be entered and then tested using an initial set of transactions. Following this, the existing ruleset will be modified in response to the transactions, and a follow-up set of transactions will be tested. Note that for credit cards, the default card number of sixteen zeroes is used in all transactions, in lieu of 'valid' numbers – cards are differentiated instead by expiry date.

Initial Set

RULES

Rule ID	*Node*	*Expression*	*Alert Lev*	*Time Period*	*TrueN*	*FalseN*
A	A1	`NUM_CCS = 2`	1	90 days	-	-
B	B1	`Any QUANTITY > 3`	0	1 min	B2	-
	B2	`Total AMOUNT >= 250`	2		-	B3
	B3	`Any QUANTITY > 3`	1		-	-
C	C1	`Total QUANTITY >= 20`	1	3 days	-	-
D	D1	`QUANTITY of 11016 > 1`	1	30 days	-	-

TRANSACTIONS

ID	*Login*	*Credit Card Details*	*Item Name*	*Price*	*Quantity*	*Item Subtotal*
1	joe@blog.com	VISA; Exp: 01/04 #0000000000000000	Men's Pleated Shorts	$25.00	8	$200.00
2	Unregistered User 1	VISA; Exp: 02/05 #0000000000000000	Men's Shirt	$25.00	2	$50.00
			Men's Wallet	$50.00	3	$150.00
			Men's Belt	$25.00	2	$50.00
3	joe@blog.com	VISA; Exp: 03/04 #0000000000000000	Men's Pleated Shorts	$25.00	6	$150.00
			Men's Wallet	$50	5	$250.00
4	Unregistered User 2	VISA; Exp: 10/06 #0000000000000000	Women's Fleece Shirt Jacket	$30.00	4	$120.00
5	jane@test.com	VISA; Exp: 07/03 #0000000000000000	Women's Snowflake Sweater	$35.00	1	$35.00

Follow-up Set

RULES

Rule ID	*Node*	*Expression*	*Alert Lev*	*Time Period*	*TrueN*	*FalseN*
E	E1	`login = joe@blog.com`	0	90 days	D2	-
	E2	`Total AMOUNT >= 500`	2		D3	D4
	E3	`NUM_CCS > 2`	2		-	-
	E4	`NUM_CCS > 2`	1		-	-

TRANSACTIONS

ID	*Login*	*Credit Card Details*	*Item Name*	*Price*	*Quantity*	*Item Subtotal*
6	joe@blog.com	VISA; Exp: 07/07 #0000000000000000	Men's Collared Shirt	$25.00	2	$50.00
			Men's Jacket	$100.00	1	$100.00
7	joe@blog.com	VISA; Exp: 10/04 #0000000000000000	Men's Collared Shirt	$25.00	2	$50.00
			Men's Jacket	$100.00	1	$100.00

Rationale for Rules

A. This rule checks if a user has used more than one credit card in the last 90 days to make an order.

B. For a clothing store, it would be considered unusual if a customer orders multiple, identical items. A customer ordering more than three of any one good would be flagged (three was arbitrarily chosen, based on two pieces of clothing for personal use, plus another piece for gift purposes). Additionally, if the order amounts to more than $250, then the order is especially noted. This rule has a time period of one minute to indicate that it should only apply to the transaction just processed.

C. Any customer ordering a large amount of items, 20 or more, over a three day period is noted.

D. This rule assumes that product 11016 (men's wallets) has historically been highly susceptible to fraud. Therefore, special attention is paid to this item where more than one wallet is ordered (in a month).

E. As the customer joe@blog.com has a highly suspicious ordering pattern (see expected results), this additional rule is set up to track that customer. It produces an onscreen alert if he orders more than $800 worth of merchandise over 90 days, and also if he is found to use more than 2 credit cards.

Alert levels have been chosen with the view that any onscreen alerts (alert level 2) generated mean that the auditor should closely inspect the transaction that generated the alert. Any alert logged to the database without an onscreen alert (alert level 1) should warrant a cursory check by the auditor, but is not considered very serious. Multiple level 1 alerts should be paid more attention, however.

7.3.2.2 Expected Results

Transaction	Result of Transaction (Node or nodes triggered)
1	• B1 • B3 (logged to database)
2	• D1 (logged to database)
3	• A1 (logged to database) • B1 • B2 (with onscreen alert) • D1 (logged to database)
4	• B1 • B3 (logged to database)
5	Nil
Note: At this stage, customer joe@blog.com has made two transactions which have generated six alerts in total. An auditor may regard this activity as highly suspicious and therefore choose to implement a rule that tracks that user closely. Rule D performs exactly this task.	
6	• C1 • E1 • E2 (with onscreen alert) • E3 (with onscreen alert)
7	Nil

7.3.2.3 Results

The results obtained from the test matched the expected results, as verified from onscreen alerts, as well as from the web reports generated by eSCARF.

7.4 Conclusion

This chapter has detailed internal system testing of eSCARF. Results showed that eSCARF functions correctly, in accordance with the requirements specification. These tests do not imply anything about the *quality* or *effectiveness* of the system (these aspects will be assessed in the next chapter), but merely verify its proper operation and conformance to the requirements specified.

Chapter 8. Evaluation Survey

This chapter contains the results and analysis of results for the evaluation survey. Firstly, a look will be taken at the sample demographics, followed by their perspectives on what is important in a continuous assurance system (addressing RQ5). Following this will be an analysis of the modules making up eSCARF, evaluating the quality of individual modules, as well as providing suggestions for improving them (also addressing RQ5). Finally, an analysis of eSCARF as a whole, and what makes eSCARF – and continuous assurance systems in general – useful will be undertaken (addressing RQ4). Following this are some suggestions for extra functionality in eSCARF that are not specific to any single module.

Please refer to appendix 7 for a copy of the questionnaire this chapter refers to.

8.1 Scales

The 5-point Likert scales used in the demographics section were transformed into interval scales according to this key:

Table 8.1.1: 5-point Likert scale scoring

Score	Scale (Knowledge)	Scale (Expertise)
1	None	None
2	Minimal	Basic
3	Adequate	Intermediate
4	Substantial	Advanced
5	Extensive	Expert

The 7-point Likert scales used in sections A (perceptions), B (component evaluation) and C (overall evaluation) were transformed into interval scales according to this key:

Table 8.1.2: 7-point Likert scale scoring

Score	Scale (Section A)	Scale (Section B, C)
1	Very Unimportant	Strongly Disagree
2	Unimportant	Disagree
3	Somewhat Unimportant	Disagree Somewhat
4	Neutral	Neutral
5	Somewhat Important	Agree Somewhat
6	Important	Agree
7	Very Important	Strongly Agree

The questions in the questionnaire will be referenced in a descriptive form by what they measure. The following list matches the question number with how it is referred to in this section:

Table 8.1.3: Question Abbreviations

Question	Referred to as…
Demographics: How would you rate the extent of your knowledge in…	
… Auditing	Knowledge of Auditing
… Information Systems	Knowledge of IS
… Information Systems Auditing	Knowledge of IS Auditing
… Continuous Assurance	Knowledge of Continuous Assurance
Demographics: How would you rate your expertise in…	
… Auditing	Expertise in Auditing
… Information Systems	Expertise in IS
… Information Systems Auditing	Expertise in IS Auditing
… Continuous Assurance	Expertise in Continuous Assurance
Section A – Perceptions	
1.	Perceived Accuracy
2.	Perceived Comprehensiveness
3.	Perceived Conciseness
4.	Perceived Timeliness
5.	Perceived Presentation
6.	Perceived Userfriendliness
7.	Perceived Customisability
Section B – eSCARF Component Evaluation	
8.	Rule Mgt Conciseness
9.	Rule Mgt Presentation
10.	Rule Mgt Userfriendliness
11.	Rule Mgt Customisability
12.	Rule Mgt Flexibility
13.	Console Accuracy
14.	Console Comprehensiveness
15.	Console Conciseness
16.	Console Timeliness
17.	Alerts Work as Expected
18.	Alerts Timeliness
19.	Web Reports Accuracy
20.	Web Reports Comprehensiveness
21.	Web Reports Conciseness
22.	Web Reports Timeliness
23.	Web Reports Presentation
24.	Web Reports Userfriendliness
Section C – eSCARF Overall Evaluation	
25.	Actual Accuracy
26.	Actual Comprehensiveness
27.	Actual Conciseness
28.	Actual Timeliness
29.	Actual Presentation
30.	Actual Userfriendliness
31.	Actual Customisability
32.	Usefulness Rating

8.2 Demographics

The final sample size for the survey was 15. 7 of the participants identified themselves as auditors with formal background in information systems (IS participants), and 8 identified themselves as auditors without an information systems background (Non-IS participants). Therefore, the survey covered two different groups of auditors.

Table 8.2.1: Frequencies – How many years experience have you had in auditing?

	Frequency			
Years Experience	**0**	**1-2**	**3-5**	**> 5**
	-	-	3	12

Table 8.2.2: Descriptive Statistics of Demographics

Question	**Participant**	***N***	**Mean**	**Std. Dev.**	**95% Confidence Interval for Mean** (*Lower, Upper Bound*)	**Min**	**Max**
Knowledge of Auditing	IS	7	4.14	.900	(3.31, 4.97)	3	5
	Non-IS	8	4.50	.535	(4.05, 4.95)	4	5
	Total	15	4.33	.724	(3.93, 4.73)	3	5
Knowledge of IS	IS	7	4.57	.535	(4.08, 5.07)	4	5
	Non-IS	8	2.88	.354	(2.58, 3.17)	2	3
	Total	15	3.67	.976	(3.13, 4.21)	2	5
Knowledge of IS Auditing	IS	7	4.57	.535	(4.08, 5.07)	4	5
	Non-IS	8	2.75	.463	(2.36, 3.14)	2	3
	Total	15	3.60	1.056	(3.02, 4.18)	2	5
Knowledge of Continuous Assurance	IS	7	3.43	.787	(2.70, 4.16)	2	4
	Non-IS	8	2.50	.535	(2.05, 2.95)	2	3
	Total	15	2.93	.799	(2.49, 3.38)	2	4
Expertise in Auditing	IS	7	4.00	.816	(3.24, 4.76)	3	5
	Non-IS	8	4.25	.707	(3.66, 4.84)	3	5
	Total	15	4.13	.743	(3.72, 4.54)	3	5
Expertise in IS	IS	7	4.29	.488	(3.83, 4.74)	4	5
	Non-IS	8	2.38	.518	(1.94, 2.81)	2	3
	Total	15	3.27	1.100	(2.66, 3.88)	2	5
Expertise in IS Auditing	IS	7	4.29	.488	(3.83, 4.74)	4	5
	Non-IS	8	2.13	.354	(1.83, 2.42)	2	3
	Total	15	3.13	1.187	(2.48, 3.79)	2	5
Expertise in Continuous Assurance	IS	7	2.71	1.113	(1.69, 3.74)	1	4
	Non-IS	8	2.13	.991	(1.30, 2.95)	1	4
	Total	15	2.40	1.056	(1.82, 2.98)	1	4

Table 8.2.3: ANOVA (Analysis of Variance) between IS and Non-IS Groups

Question	Sum of Squares	Mean Square	F	Significance
Knowledge of Auditing	.476	.476	.903	.359
Knowledge of IS	10.744	10.744	53.943	.000*
Knowledge of IS Auditing	12.386	12.386	50.093	.000*
Knowledge of Continuous Assurance	3.219	3.219	7.323	.018*
Expertise in Auditing	.233	.233	.404	.536
Expertise in IS	13.630	13.630	53.635	.000*
Expertise in IS Auditing	17.430	17.430	98.363	.000*
Expertise in Continuous Assurance	1.296	1.296	1.178	.297
** indicates a significant amount of variance between IS and Non-IS auditors (significance < 0.05)*				

Results from the demographics section show that all the participants matched the target demographic for this survey, with all having at least 3 years of experience, and 80% of those having more than 5 years. The average participant was found to have a substantial amount of knowledge and expertise in auditing (with means of 4.33 and 4.13, respectively), with an insignificant level of variation between IS and non-IS auditors. The minimum level of auditing knowledge or expertise claimed was "adequate". A 95% confidence interval places auditing knowledge within the 3.93 to 4.73 range, and auditing expertise within the 3.72 to 4.54 range. These figures clearly show all participants have a level of auditing experience that conforms with the sampling requirements for this survey.

As expected, knowledge of IS and IS auditing was significantly higher for IS participants, as shown by the shaded figures in table 8.2.3. Knowledge and expertise of continuous assurance recorded lower means than the other three fields of knowledge, indicating that the field of continuous assurance is not so well known, even amongst very experienced auditors. Furthermore, knowledge of continuous assurance is significantly different between the two types of participants (shown by a significance of 0.018), with the average non-IS participant claiming a minimal to adequate level of knowledge, and the average IS participant claiming an adequate to substantial level of knowledge. This implies that continuous assurance is a field connected with information systems, which would agree with the literature, which says that implementing continuous assurance systems requires auditors with knowledge of information systems (CICA 1999). However, expertise in continuous assurance did not vary significantly between the two groups, reflecting perhaps that practical experience in continuous assurance is lacking due to the field's relative newness.

8.3 Perspectives

The perspectives section attempts to gauge the importance of various attributes of a continuous assurance system.

Table 8.3.1: Descriptive Statistics of Perspectives ranked by Relative Importance

Question	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Perceived Timeliness	15	6	7	6.80	.414	(6.57, 7.03)
Perceived Accuracy	15	5	7	6.67	.724	(6.27, 7.07)
Perceived Customisability	15	5	7	6.40	.828	(5.94, 6.86)
Perceived Userfriendliness	15	4	7	6.40	.828	(5.94, 6.86)
Perceived Comprehensiveness	15	5	7	6.27	.704	(5.88, 6.66)
Perceived Conciseness	15	5	7	6.20	.676	(5.83, 6.57)
Perceived Presentation	15	4	7	5.87	.990	(5.32, 6.42)

All attributes, with the exception of presentation, were considered to be within the range of important to very important. Timeliness was considered the most important attribute with a mean of 6.8, which is in line with the major advantage continuous assurance has over traditional assurance or auditing. Timeliness was followed by accuracy, customisability, userfriendliness, comprehensiveness, conciseness and presentation.

An ANOVA test between groups did not show any significant differences in perceived importance of attributes between IS and non-IS auditors.

8.4 eSCARF Component Evaluation

This section takes a component-by-component view, quantitatively analysing the attributes of each eSCARF module evaluated, and qualitatively analysing participants' impressions of, and suggestions for them.

8.4.1 Rule Management

The rule management module was assessed for the attributes of conciseness, presentation, userfriendliness, customisability and flexibility.

Table 8.4.1.1: Descriptive Statistics of the Rule Management module

Question	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Rule Mgt Conciseness	15	6	7	6.20	.414	(5.97, 6.43)
Rule Mgt Presentation	15	5	7	6.13	.640	(5.78, 6.49)
Rule Mgt Userfriendliness	15	5	7	6.07	.458	(5.81, 6.32)
Rule Mgt Customisability	15	4	7	5.87	.915	(5.36, 6.37)
Rule Mgt Flexibility	15	4	7	5.87	.915	(5.36, 6.37)
Mean of Component Means				6.027		

Participants generally agreed that the rule management module exhibited the attributes listed above, with the attribute means ranging from 5.87 to 6.20 with an overall mean of 6.027.

Although naturally ease of customisation is an important factor when managing audit rules, one participant noted that one *"would not want to make [rules] too easy to customise. A degree of knowledge required for customisation is in itself a possible control."* System modification can significantly affect the way eSCARF works, and thus affect its effectiveness. By imposing that the user must be somewhat knowledgeable about the system in order to modify it, a control is created to prevent unskilled users from modifying the system improperly.

An ANOVA test between groups did not show any significant differences in opinion between IS and non-IS auditors.

8.4.1.1 Additional Features Suggested

- **Setting Alert Levels:** Alert levels should be selectable via radio buttons, or a drop down box, instead of having to manually type in a number.
- **Online Help:** Online documentation, accessible from a "Help" option on the menu bar would be an important feature for auditors to procure quick help using the rule management module.
- **New Rule Operators:** In addition to the current operators (eg: >, <, =, etc.), some new operators that may be of use include minimum and maximum values, and fuzzy operators, such as 'high value' and 'abnormal value'. For fuzzy operators, what would be considered high or abnormal would vary dynamically, depending on the nature of transactions flowing through the e-commerce system, and would also be adjustable by auditors.

- **Standard Rule-set:** A set of 'standard' or 'suggested' rules may be incorporated to provide auditors with some initial suggestions or guidance when first setting up rules for eSCARF.

8.4.2 Server Console Log

The server console log was assessed for the attributes of accuracy, comprehensiveness, conciseness and timeliness.

Table 8.4.2.1: Descriptive Statistics of the Server Console Log module

Question	**N**	**Min.**	**Max.**	**Mean**	**Std Dev**	**95% Confidence Interval for Mean**
Console Accuracy	15	4	7	6.00	.926	(5.49, 6.51)
Console Comprehensiveness	15	4	7	6.13	.834	(5.67, 6.60)
Console Conciseness	15	2	7	5.47	1.246	(4.78, 6.16)
Console Timeliness	15	5	7	6.33	.724	(5.93, 6.73)
Mean of Component Means				5.98		

Participants generally agreed that the console log was accurate, comprehensive and timely (the overall mean was 5.98). However, participants only somewhat agreed that the information the server console log delivered was concise.

The lower rating for conciseness was elaborated upon when the console log was described as confusing, due to two factors: poor formatting of log entries and poor phrasing of log entries. Currently, the log entries appear jumbled due to the poor formatting. The server log is designed for technical diagnostic purposes, so it was envisioned that non-IS participants may have had trouble understanding it. However, *both* IS and non-IS participants found the entries hard to understand, implying that the entries are phrased unclearly.

An ANOVA test between groups did not show any significant differences in opinion between IS and non-IS auditors.

8.4.2.1 Additional Features Suggested

- **Timestamp Console Entries:** Each entry in the console log should be timestamped (with a date and time).

- **Improve Clarity of Entries:** As the current entries made in the console log are currently quite cryptic to auditors, they must be rephrased to be more descriptive. Entries should also be formatted with extra spacing.
- **Window Resizing:** Resizing and maximising of the console window should be enabled.
- **Log Archiving:** The output from the server log should be logged to a file for storage, providing an archive of past events occurring on the server. (Also see section 8.5.1)

8.4.3 Rule Checking and Alerts

Rule checking and alerts were assessed for the attributes of correct operation (accuracy) and timeliness.

Table 8.4.3.1: Descriptive Statistics of the Rule Checking and Alerts module

Question	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Alerts Work as Expected	15	6	7	6.67	.488	(6.40, 6.94)
Alerts Timeliness	15	6	7	6.67	.488	(6.40, 6.94)
Mean of Component Means				6.67		

Participants strongly agreed that alerts worked as expected (indicating that rule checking worked as expected too) and were timely (all means were 6.67).

Two comments were made with regards to alerts and rule checking. Firstly, as the test was conducted using a one-tier configuration, no delays were experienced with the generation of alerts. However, if the eSCARF system and e-commerce system are geographically separated, and have to communicate over a network or the Internet, the timeliness of alerts may be impacted upon if communication delays are experienced. This is an issue that will have to be considered during the integration process of eSCARF. The link between eSCARF and the e-commerce system must be able to handle the volume of transactions that are expected to be sent over it.

Secondly, a problem may arise with alerts being *too* timely. For systems that process a very high volume of transactions, a correspondingly (relatively) high number of alerts could be generated. If the alerts are onscreen, this could lead to the auditor being confronted with a screen full of alerts. A possibility would be to collate a series of alerts and release them in an aggregated form every few minutes or hours. The system's alerts would not be real-time anymore. However, this is not an issue if the auditor cannot act upon information generated in real-time in a timely manner because there is too much information for him or her to process!

An ANOVA test between groups did not show any significant differences in opinion between IS and non-IS auditors.

8.4.3.1 Additional Features Suggested

- **Linking Onscreen Alerts to Reports:** For onscreen alerts, users should be able to click on a link that will bring them directly from the onscreen alert to the relevant web report for the purposes of further inspection.
- **Running Rules Retrospectively:** Having the capability to run new rules on historical sets of data would be a useful feature. Although not directly in the scope of a continuous assurance system, this feature would help auditors fine-tune their rulesets.

8.4.4 Web Reporting

Web Reporting was assessed for the attributes of accuracy, comprehensiveness, conciseness, timeliness, presentation and userfriendliness.

Table 8.4.4.1: Descriptive Statistics of the Rule Management module

Question	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Web Reports Accuracy	15	5	7	6.40	.632	(6.05, 6.75)
Web Reports Comprehensiveness	15	4	7	5.93	.961	(5.40, 6.47)
Web Reports Conciseness	15	3	7	5.87	1.060	(5.28, 6.45)
Web Reports Timeliness	15	6	7	6.47	.516	(6.18, 6.75)
Web Reports Presentation	15	5	7	6.20	.862	(5.72, 6.68)
Web Reports Userfriendliness	15	5	7	6.27	.704	(5.88, 6.66)
Mean of Component Means				6.19		

Participants generally agreed that the web reports were accurate, comprehensive, concise, timely, well presented and userfriendly. Out of these attributes, participants were in strongest agreement about the reports being timely and accurate. However, the attributes associated with content – comprehensiveness and conciseness were the lowest rated attributes (5.93 and 5.87, respectively), both with relatively high standard deviations (.961 and 1.060, respectively), implying that different auditors had varying views of what information they thought the reports should supply.

This module generated the most scrutiny and volume of comments from auditors, reflecting its relative importance as an eSCARF module. As noted above in the statistics, views appeared to differ between auditors about the comprehensiveness and conciseness of eSCARF's content. Correspondingly, many comments were made in reference to the content of the reports. The focus on content was not surprising, as it is the range of reports that can be generated by eSCARF, and the information those reports contain, which define what analysis an auditor can perform on the data that has been collected by eSCARF. Many suggestions were made regarding additional information that the reports could provide, or how existing reports could be modified for greater clarity and conciseness. These suggestions are detailed in the next section.

Another pertinent issue raised was how the conciseness and presentation of the reports may decrease with *"a large number of reported infringements"*. All tests and demonstrations so far have only dealt with small transaction sets, but work would have to be done to see how web reports would display a large amount of data, and how that could be effectively handled.

An ANOVA test between groups did not show any significant differences in opinion between IS and non-IS auditors.

8.4.4.1 Additional Features Suggested

- **More Statistics:** The output of various additional statistics would be helpful for auditors analysing system activity. For example, tracking the ranges and average prices of orders would help auditors determine a price at which an order would be considered suspicious if it exceeded it.
- **Rule Management Report Changes:** Instead of noting the rule ID next to rule names, the rule ID should be replaced with the rule version, which is more meaningful. The bar graph at the bottom of the page should be labelled as a bar graph, and the numbers currently to the right of the bars should instead be left-aligned for readability reasons.

- **Rule ID Numbering:** ID numbering currently starts at zero (0), which although common practice in information systems, may be confusing to those without information systems backgrounds (this opinion was expressed by such a person). Numbering starting at one (1) would be more logical
- **Cross-Referencing Parameterisation:** When viewing transaction detail reports, the option to cross reference a particular transaction with the previous 20 transactions should actually be user specifiable (that is, the user should be able to specify a number other than 20).
- **Viewing Transactions on the Transaction Summary Report:** A frames layout could be implemented on the transaction summary report. Instead of a new page loading up when an auditor views details for a specific transaction, the transaction details could load up within a frame on the same page as the transaction summary report, allowing the auditor to quickly browse between different transactions without having to alternate between transaction summary and transaction detail pages.
- **OLAP Analysis:** The ability for an auditor to perform online analytical processing (OLAP) on the data collected by eSCARF would provide a powerful tool for further analysis. This could be accomplished by implementing a viewing mechanism such as a pivot table on reports like transaction history.

8.4.5 Component Comparison

A comparison between components' relative usefulness is possible by comparing each component's overall usefulness score. Each component's overall usefulness score is a obtained by taking the mean of all its attribute ratings:

Table 8.4.5.1: Descriptive Statistics for Usefulness of Components

Question	N	Minimum	Maximum	Mean	Std Dev
Average for Rule Mgt	15	5.2	7	6.03	.453
Average for Console	15	4	7	5.98	.704
Average for Alerts	15	6	7	6.67	.488
Average for Web Reports	15	4.8	7	6.12	.675

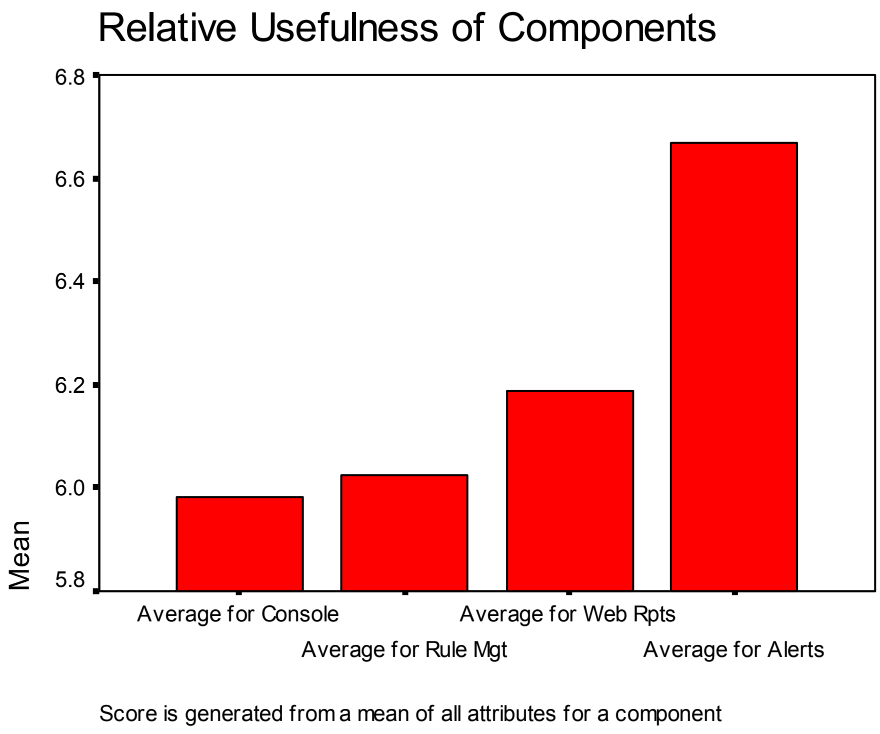


Figure 7: Ranking of Components' Relative Usefulness

As can be seen from the graph (figure 7), the average for the rule checking and alerts module was the highest at 6.67, given its timely and accurate operation. The overall score for the console was the lowest, when compared with other components, perhaps for its lack of clarity (it was the only component regarded as "confusing" by participants).

Finally, as shown by the 95% confidence interval for mean statistics in tables 8.4.1.1, 8.4.2.1, 8.4.3.1 and 8.4.4.1, no attribute for any eSCARF component had a *lower* bound confidence interval score of less than 5 ("agree somewhat"). This further confirms the effectiveness of each and every one of eSCARF's individual components.

8.5 eSCARF Overall Evaluation

The purpose of this section is to evaluate eSCARF as an entire system in an attempt to ascertain whether auditors believe it is a useful system.

Table 8.5.1: Descriptive Statistics of eSCARF's Attributes

Attributes	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Actual Accuracy	15	6	7	6.53	.516	(6.25, 6.82)
Actual Comprehensiveness	15	5	7	6.27	.704	(5.88, 6.66)
Actual Conciseness	15	5	7	6.13	.743	(5.72, 6.54)
Actual Timeliness	15	6	7	6.67	.488	(6.40, 6.94)
Actual Presentation	15	5	7	6.13	.743	(5.72, 6.54)
Actual Userfriendliness	15	5	7	6.07	.704	(5.68, 6.46)
Actual Customisability	15	4	7	5.87	.834	(5.40, 6.33)

The descriptive statistics show that participants agree eSCARF is comprehensive, concise, timely, userfriendly, customisable and presents information well. Participants tended to *strongly* agree that eSCARF was accurate and timely. All of these attributes were deemed to have been important in the perceptions section of the questionnaire.

The mean score for actual timeliness (6.67) similarly matches the rating for the highest rated module, rule checking and alerts. This module enables the timely provision of assurance information to auditors, so it makes sense that timeliness has also received a high score. This is an encouraging finding as the main advantage eSCARF offers to auditors, being a continuous assurance system, is the ability to provide real-time assurance for business.

The lowest score was customisability (5.87). Several participants noted that eSCARF's ability to assure was only as useful as the rules in place: *"Obviously, the benefits from [eSCARF's] use will be very tied to the rules set up."* The rule generation capabilities in this version of eSCARF are rudimentary, and while extensible by auditors, requires knowledge of Java programming. Showing participants how to customise rules in this way was not in the scope of the demonstration (it requires technical knowledge – that is, an auditor with a background in information systems). Neither was the ability to set up complex auditing rules in the scope of this thesis, but it explains why the system did not appear readily customisable to some participants.

An ANOVA test between groups did not show any significant differences between IS and non-IS auditors in how useful eSCARF was thought to be.

Table 8.5.2: Descriptive Statistics of eSCARF's Usefulness Rating

Usefulness Rating	N	Min.	Max.	Mean	Std Dev	95% Confidence Interval for Mean
Usefulness Rating	15	3	7	5.93	.961	(5.40, 6.47)

The mean usefulness rating of eSCARF was 5.93, which means that participants generally agreed that it was overall a useful system.

Impressions of the system were very positive from all participants, indicating they clearly understood the uses and usefulness of an e-commerce continuous assurance system. Comments expressing impressions of eSCARF's usefulness included:

> *"Can see a clear need and usefulness for eSCARF."*
> *"The system appears to be a useful tool."*
> *"Has great potential, especially as an internal audit tool."*
> *"Appears extremely relevant to the internal audit function within an organisation."*
> *"Very useful for controlling internal purchasing. Useful as a marketing tool."*

Participants also remarked positively on the overall usability and understandability of eSCARF, denoting that the system was usable by both auditors with IS and non-IS backgrounds (however, integrating eSCARF with an e-commerce system still requires expertise in IS):

> *"Appears very ease to tailor/modify to suit users' or organisations' own individual use – a very good feature."*
> *"Appears extremely flexible and user friendly."*
> *"I thought the system was easy to use and understand."*
> *"Very user friendly. Satisfied with the simplicity with which rules can be generated."*

Therefore, the general reaction to the system favours its continual development. Naturally, participants also had many observations and suggestions to make, which may be taken into account for the future. It was interesting to note that IS participants focused more on the IS issues surrounding a continuous assurance system, such as (technical) security features, maintenance and backup procedures, than the non-IS participants, which was expected given their respective backgrounds.

"Also, I would think that as a detection/control mechanism, it may be very useful as a management tool, used by senior management as well as auditors." One observation made was that implementing eSCARF in a business was a decision that would concern management. As eSCARF is a control for risk management (against the risk of fraud), the decision to implement such a system would be of strategic interest to businesses at the management level, not just those dealing directly with transaction processing. Furthermore, it was postulated that eSCARF would be more of interest to internal auditors than external auditors, because of its role in business. (Although, external auditors may be called in to set up an eSCARF type system if a business lacks the technical knowledge in-house to do it themselves.)

Another observation was how eSCARF's capability of detecting fraud could be extended for purposes of error detection as well. The participant who noted this suggested that 70% of controls currently used for fraud detection could, with a few minor modifications, also double as error detection controls. Conceivably, this could be possible with eSCARF, as the audit rules it uses are in effect detecting 'errors' or 'irregularities' in transactions. More research should be performed to explore the possibilities of expanding the scope of eSCARF.

Major observations were made by several participants regarding the business processes surrounding the use of eSCARF. These fell into the categories of training, maintenance and security. Naturally, as eSCARF is a specialised system (designed for use by auditors), users will need to be trained in its use. Research needs to be done into implementing effective audit rules so that this knowledge can be passed onto auditors through training.

The system will also need to undergo continual maintenance for the fine tuning of rules, and adapting the system to changes in patterns of transactions (for example, if a business begins expanding, the e-commerce system will subsequently process a higher volume of transactions, a fact that the auditor must adjust for in eSCARF). The fine tuning of audit rules is an iterative process, where auditors progressively adjust rules in order to reduce the occurrence of false positives as much as possible. Therefore, businesses should give due consideration to the resourcing issues surrounding the initial implementation and maintenance of eSCARF. Two responses expressing these points were:

> *"Adequate training is important. The system cannot just be implemented either, it constantly needs monitoring, additional customisation and tailoring; potentially high cost."*
> *"Monitoring seems to be a potentially time consuming activity, particularly until the rules are well customised and are working properly. This learning process would be an interesting additional area."*

Furthermore, the introduction of a new information system into a business raises many issues with security. Although eSCARF is designed to reduce risk in business, it brings risks of its own that must be considered. One instance of this is what should be done if the communications link between eSCARF and the e-commerce system fails (thereby rendering eSCARF unable to check any transactions)? If transaction processing continues on the e-commerce system in the event of such a failure, the ability to detect fraud will be lost. On the other hand, if transaction processing is made to halt, customers will no longer be able to order from the store. This translates into lost sales for the business. Although this means that no orders will escape from being monitored by eSCARF, it also adds an additional point of systems failure for the business. This may be an unacceptable risk for businesses which process high volumes of orders. Therefore, an acceptable solution to this must be found. For example, if the audit hook detects a communications failure, it buffers incoming transactions until a connection can be re-established with the eSCARF server, whereupon it sends the backlogged transactions. This allows the e-commerce system to continue processing orders, while also allowing eSCARF to (eventually) monitor all of them.

Businesses should be aware that internal fraud is still a major issue, because eSCARF is aimed at detecting only external fraud. One participant noted:

> *"I am concerned about the resolution process. It seems that the resolutions are not tracked and monitored. As an auditor, this could be a flaw in the controls. Eg: An order is placed which triggers an alert. It is an order placed by a friend of the*

> *auditor seeing the alert and they approve the transaction (which is fraudulent). There is a need to track the resolution procedure and subsequent success/failure rates."*

Business processes surrounding the use of eSCARF should be reviewed to consider threats arising from internal fraud. As in the quote above, the resolution process needs to be mapped out. Auditors should not be allowed to approve transactions, but rather report on alerts generated by eSCARF. Similarly, businesses should have timely access to auditor reports since they have ultimate authority over approving or rejecting transactions. However, business staff should not be able to modify the operation of eSCARF (which is the domain of the auditor). Special attention must be given to the audit hook, as it resides on the business' side, as opposed to the rest of eSCARF which is under the control of auditors. Future research should look into measures to prevent the audit hook from being maliciously manipulated by internal staff, such as obfuscating the audit hook Java code such that it cannot be usefully decompiled and modified.

Similarly, as eSCARF integrates with an operational e-commerce system, is it important that its security meets or exceeds the security standards imposed on the e-commerce system it is assuring. *"eSCARF needs an overall security framework to provide assurance to the auditor and to the 'owner' of the e-commerce store."* That is, a security review of eSCARF is required – the continuous assurance system itself needs to be assured, perhaps by a program such as SysTrust. Security controls need to be placed within eSCARF itself, such as audit logs of changes made to the system ('auditing the auditor'), password protection and encrypting sensitive data. Suggestions for constructing a security framework for eSCARF are noted in section 8.5.1, as it involves adding extra functionality.

Even though eSCARF is a relatively new system, it has been found to be useful, providing auditors with timely, accurate, comprehensive, concise and well presented information, accessible through a userfriendly interface. The system is easy to understand and can be customised to meet auditors' needs. However, because it is still relatively new, there are many future avenues for eSCARF development, namely through the comments detailed above, and the additional features suggested by auditors, listed in the next section.

8.5.1 Additional Features Suggested

- **Security Measures:** The issue of security was initially brought up in section 6.6.1. During the survey, its importance has been further emphasised by the participants. As noted previously, the introduction of eSCARF into a business introduces new risks which must be addressed. Some controls for these risks include:
  - Logging when rules are activated, deactivated, created, modified or deleted.
  - Logging the server console log output to a file.
  - Logging when the eSCARF server is started and stopped.
  - Password protecting access to the eSCARF main menu and web reports. Separate levels of authorisation could be implemented for different functions. For example, all auditors may have access to web reports, but only some may be allowed to perform administrative tasks such as edit rules and start/stop the server.

- **Database Maintenance:** Currently, the only database maintenance tool available is a batch file which clears out the eSCARF database. It became clear through the survey that a database maintenance module, containing a number of maintenance tools, was required. Users should be able to backup/archive and restore entries in the eSCARF database (such as transactions, alerts, rules, etc.) to a file or set of files. Although most major relational database management systems (RDBMS) come with built-in backup tools, eSCARF's backups should be performed in a manner which also makes them portable. That is, by using standard SQL commands to backup and restore data, system data can be transported in between different RDBMSes, such as IBM DB2, Oracle and Microsoft SQL Server.

  Furthermore, maintenance features allow auditors to keep past logs of transactions for future reference, useful for offline analysis and data mining, etc. Marking which transactions in the logs were fraudulent also allows later identification of fraudulent transactions with similar traits. These logs may be restored to new databases in other implementations of eSCARF for testing out different rulesets. Restoration should be allowed to occur in two ways – completely replacing the current database, or merging backed up data into the current database.

As the size of the eSCARF database can grow to be voluminous in time, backup files should be compressed to conserve disk space.

- **Web Interface for Rule Design:** If possible, future work should look at creating a web interface for designing rules. This would unify all of eSCARF's administrative functions, making them accessible via a web browser interface. The issue here is successfully translating the specialised GUI used for the rule designer into a web page interface.

  Because the use of the world wide web is so widespread, users are already familiar with the interface of a web browser. Making all of eSCARF's administrative features available through a web interface would allow centralised administration and enhance general ease of use as auditors would not have to learn any new interfaces.

- **Rule Plug-ins Interface:** Ideally, new audit rules and audit methods should be able to be easily added to eSCARF. A plug-in type mechanism could be used that would make this process as easy as selecting a rule module file. A rule module file could be a data mining module, or perhaps an interface to another system (such as an address verification system).

8.6 Exploratory Statistics and Relationships

As this survey is exploratory in nature, some correlation matrices and regression models were created in an attempt to discern any relationships between the concepts measured. Through the analysis above, two possibilities for relationships were determined:

1. That a relationship exists between eSCARF's seven individual attributes (table 8.5.1) and eSCARF's overall usefulness.
2. That a relationship exists between the usefulness of eSCARF's individual modules (table 8.4.5.1) and eSCARF's overall usefulness.

Because of the smallness of the sample size, the results below are tentative, but may still provide some valuable insights to guide future research work.

8.6.1 Correlations

8.6.1.1 Correlation Matrix – eSCARF Attributes to Usefulness

The first correlation analysis undertaken aimed at detecting if associations between eSCARF's attributes (accuracy, comprehensiveness, conciseness, etc.) and its overall usefulness, existed.

Table 8.6.1: Correlation Matrix of eSCARF's Attributes to its Usefulness Rating

		UR	**AA**	**ACm**	**ACn**	**AT**	**AP**	**AU**	**AC**
(UR) Usefulness Rating	Pearson Corr.	1	.365	.556*	.413	.558*	.213	.218	.166
	Sig. (2-tailed)	.	.182	.031	.126	.030	.445	.435	.553
(AA) Actual Accuracy	Pearson Corr.	.365	1	.367	.732**	.472	.732**	.485	.509
	Sig. (2-tailed)	.182	.	.179	.002	.075	.002	.067	.053
(ACm) Actual Comprehensiveness	Pearson Corr.	.556*	.367	1	.747**	.485	.337	.538*	.552*
	Sig. (2-tailed)	.031	.179	.	.001	.067	.220	.038	.033
(ACn) Actual Conciseness	Pearson Corr.	.413	.732**	.747**	1	.525*	.741**	.665**	.607*
	Sig. (2-tailed)	.126	.002	.001	.	.044	.002	.007	.016
(AT) Actual Timeliness	Pearson Corr.	.558*	.472	.485	.525*	1	.525*	.277	.410
	Sig. (2-tailed)	.030	.075	.067	.044	.	.044	.317	.129
(AP) Actual Presentation	Pearson Corr.	.213	.732**	.337	.741**	.525*	1	.801**	.492
	Sig. (2-tailed)	.445	.002	.220	.002	.044	.	.000	.063
(AU) Actual Userfriendliness	Pearson Corr.	.218	.485	.538*	.665**	.277	.801**	1	.503
	Sig. (2-tailed)	.435	.067	.038	.007	.317	.000	.	.056
(AC) Actual Customisability	Pearson Corr.	.166	.509	.552*	.607*	.410	.492	.503	1
	Sig. (2-tailed)	.553	.053	.033	.016	.129	.063	.056	.

n = 15
** shows a correlation that is significant at the 0.05 level (2-tailed).*
*** shows a correlation that is significant at the 0.01 level (2-tailed).*

Due to the small sample size, no correlation between eSCARF's attributes and its usefulness was expected. Despite the sample size of 15, the results were encouraging. The correlation analysis found that timeliness and comprehensiveness were significantly correlated with system usefulness, achieving scores of 0.558 and 0.556 respectively, which are significant at the 0.01 level (two-tailed).

It is logical that timeliness should correlate to usefulness, as the main benefit of a continuous assurance system over traditional audit techniques, is the timely matter in which assurance may occur. The correlation of comprehensiveness also implies that providing sufficient information to system users has a direct bearing upon the system's usefulness.

8.6.1.2 Correlation Matrix – eSCARF Components to Usefulness

The second correlation analysis undertaken aimed at detecting if associations between eSCARF's components and its overall usefulness, existed. Again, the same sample size constraints existed for this analysis as well.

Table 8.6.2: Correlation Matrix of eSCARF's Component Ratings to its Usefulness Rating

		Usefuln. Rating	Rule Mgt	Console Log	Rule Chkr. & Alerts	Web Reports
Usefulness Rating	Pearson Correlation	1	.201	.157	.558*	.259
	Signif. (2-tailed)	.	.472	.577	.030	.351
Mean for Rule Management†	Pearson Correlation	.201	1	.427	.625*	.574*
	Signif. (2-tailed)	.472	.	.112	.013	.025
Mean for Console Log†	Pearson Correlation	.157	.427	1	.555*	.583*
	Signif. (2-tailed)	.577	.112	.	.032	.022
Mean for Rule Checker & Alerts†	Pearson Correlation	.558*	.625*	.555*	1	.747**
	Signif. (2-tailed)	.030	.013	.032	.	.001
Mean for Web Reports†	Pearson Correlation	.259	.574*	.583*	.747**	1
	Signif. (2-tailed)	.351	.025	.022	.001	.

n = 15
** shows a correlation that is significant at the 0.05 level (2-tailed).*
*** shows a correlation that is significant at the 0.01 level (2-tailed).*
† See 8.4.5 for calculation of component means.

The quality of the rule checker and alerts module was found to correlate with overall system usefulness (scoring 0.558 which is significant at the 0.05 two-tailed level). Incidentally, this result matches the correlation findings in table 8.6.1, in that real-time rule checking and alerts are manifestations of eSCARF's *timeliness* in its provision of assurance. This somewhat reinforces the true existence of both correlations, despite the small sample size.

8.6.2 Regressions

Following the correlation analysis, linear regression models were produced for the same sets of relations (eSCARF's attributes as predictors for its overall usefulness; and eSCARF's components as predictors for its overall usefulness). Firstly, a normal regression model was created using all variables as predictors. Secondly, a stepwise regression model was created.

8.6.2.1 Regression Model – eSCARF Attributes and Usefulness

Table 8.6.3: Regression Model Statistics for eSCARF's Attributes and Usefulness Rating

R	R^2	Model Significance
.738	.545	.409
***Predictors**: (Constant), Actual Customisability, Actual Timeliness, Actual Userfriendliness, Actual Accuracy, Actual Comprehensiveness, Actual Conciseness, Actual Presentation* ***Dependent Variable**: Usefulness Rating*		

As expected with a small sample size, no meaningful regression model was produced, with an insignificant significance level of 0.409.

8.6.2.2 Regression Model – eSCARF Components and Usefulness

Table 8.6.4: Regression (Stepwise) Model Statistics for eSCARF's Attributes and Usefulness Rating

R	R^2	Model Significance
.558	.312	.030
***Predictors**: (Constant), Actual Timeliness* ***Dependent Variable**: Usefulness Rating*		

Variable	Unstandardised Coefficients		Standardised Coefficients	t	sig
	B	Std Error	Beta		
(Constant)	-1.400	3.029	-	-.462	.652
Actual Timeliness	1.100	.453	.558	2.427	.030

A stepwise regression model excluded all attributes except timeliness (found to be a correlating variable in table 8.6.1). The model has a significance of 0.030 and produces the following regression equation:

$$y = -1.4 * 1.1x$$

where: y is the usefulness rating; and x is the score for actual timeliness

Table 8.6.5: Regression Model Statistics for eSCARF's Component Ratings and Usefulness Rating

R	R^2	Model Significance
.633	.400	.233
***Predictors**: (Constant), Mean for Rule Management, Mean for Console Log, Mean for Rule Checker & Alerts, Mean for Web Reports* ***Dependent Variable**: Usefulness Rating*		

As expected with a small sample size, no meaningful regression model was produced, with an insignificant significance level of 0.233.

Table 8.6.6: Regression (Stepwise) Model Statistics for eSCARF's Component Ratings and Usefulness Rating

R	R^2	Model Significance
.558	.312	.030
***Predictors**: (Constant), Mean for Rule Checker & Alerts* ***Dependent Variable**: Usefulness Rating*		

Variable	Unstandardised Coefficients		Standardised Coefficients	t	sig
	B	Std Error	Beta		
(Constant)	-1.400	3.029	-	-.462	.652
Rule Checker and Alerts Module	1.100	.453	.558	2.427	.030

A stepwise regression model excluded all components except the rule checker and alerts modules (found to be a correlating variable in table 8.6.2). The model has a significance of 0.030 and produces the following regression equation:

$$y = -1.4 * 1.1x$$

where: y is the usefulness rating; and x is the score for the rule checker and alerts module

8.7 Survey Limitations

The major limitation of this survey is its low sample size. Essentially, this means that the survey is only a pilot and statistical inferences cannot *reliably* be made from the quantitative analysis. Nonetheless, impressions and insights can be gathered and the qualitative components of the survey may still be meaningfully analysed.

The validity and reliability of this questionnaire was not technically verified. For example, Cronbach Alphas were not calculated to give a measure of the questionnaire's reliability. Section 5.3.3 noted several limitations of the survey including self-efficacy issues and order bias (how recording participants' perceptions at the start of the questionnaire may affect their opinions of eSCARF which are recorded after perceptions).

The demonstration format for the survey also poses several constraints. The view participants get of the system is limited to what was shown in the demonstration procedure. Participants asked questions throughout the demonstration, and could explore other parts of eSCARF to satisfy their curiosity, but still the simulation environment was an artificial one. To gain a proper appraisal of eSCARF, a survey would have to be taken of people who have had the opportunity to use the system for themselves in a real world, natural setting.

Furthermore, some participants had a tendency to ask more questions about the system and probe deeper into its capabilities. Clearly, these participants would have gained a more thorough understanding of everything and have been better informed in their opinions of eSCARF. Some participants were time constrained, and thus had limited opportunity for questions and probing. These time constraints prevented the demonstration procedure used for all participants from being any longer (to ensure that all participants would at least get to see the key features of the system within the allotted time).

Nonetheless, despite these conclusions, the survey was ultimately informative and served its purpose to gain an evaluation of eSCARF from end-users, providing direction for future development work on the system, as well as gaining insight into an auditor's view of continuous assurance systems.

8.8 Conclusions

The evaluation survey collected data from 15 participants, all with significant experience in auditing. Two sub-groups existed within the sample, auditors with backgrounds in information systems, and auditors without. 7 participants were of the former group, and 8 of the latter. An ANOVA test between the two groups confirmed significant differences in the level of knowledge and expertise in information systems and information systems auditing, although auditing knowledge and expertise levels were both similar (averaging at "substantial" for knowledge and "advanced" for expertise).

Quantitative analysis of results occurred over the three sections of the questionnaire: user perspectives, component evaluations and overall system evaluation. For user perspectives, timeliness and accuracy were perceived to be very important to auditors, with the other attributes of customisability, userfriendliness, comprehensiveness, conciseness and presentation all regarded as important. Participants agreed that the four components of eSCARF demonstrated were all useful, with strong agreement that the rule checking and alerts module was useful. All lower bounds for 95% confidence interval of means were above 5 ("somewhat agree"), a very positive result concerning the build quality and usefulness of eSCARF's components.

Overall, participants strongly agreed that eSCARF was timely and accurate, with general agreement that it also possessed the other traits of comprehensiveness, conciseness, presentation, userfriendliness and customisability. This corresponds with the perceptions measured that found participants believed that a continuous assurance system should be most importantly timely and accurate. The holistic usefulness rating produced a positive overall result of 5.93, meaning that participants agreed that eSCARF was a useful system. These results answered RQ4 – whether auditors perceived eSCARF was useful for assuring e-commerce systems.

Correlation matrices were produced in an attempt to discern any relationships between attributes and overall usefulness, and also between eSCARF's component and overall usefulness. Despite the smallness of the sample size, analysis discovered a significant relationship between timeliness and usefulness, and accuracy and usefulness. That is, participants' perceptions of timeliness and accuracy positively correlated with their overall view of system usefulness. A positive correlation between eSCARF's rule checker and alerts module and overall usefulness was also discovered. As this module's task is to deliver assurance results in a timely manner, this result aligns with the other correlation matrix which deemed timeliness as a significantly correlating attribute.

Two regression models were created based on the same data sets as the correlation matrices. As expected due to a small sample size, no meaningful models were produced, except when stepwise regression was used. In the first stepwise regression model between attributes and usefulness, all attributes except timeliness were excluded. In the second stepwise regression model between components and usefulness, all components except the rule checker and alerts module were excluded. These models, however, are of limited use.

Qualitative analysis provided a considerable number of insightful suggestions for improving individual system components (such as improving the presentation of the server console log), as well as suggestions for added system functionality (such as the creation of a security framework for eSCARF and the addition of a database maintenance module). Expressed in words, participants displayed a clear understanding of eSCARF's purpose. They found the system was easy to use, was simple yet functional, and importantly determined it was a very useful tool, feasible in use for fraud detection in industry.

The relationships discovered in the correlation and regression analyses, combined with the qualitative data gathered, have provided a significant list of factors of what auditors regard as important in the design of a continuous assurance system, thereby answering RQ5.

Although the user evaluation survey was limited by the small sample size, both quantitative and qualitative results showed a clear indication that participants perceived eSCARF as a useful system. This shows eSCARF is a successful implementation of a continuous assurance system, and is feasible for use as a fraud detection instrument by businesses. The feedback received encouraged the continued development of eSCARF, and provided recommendations on how to further improve the system.

Chapter 9. Conclusions

This chapter summarises the findings of this thesis in terms of the achievement of the research objectives, and the answering of the research questions. Limitations of this research, and future avenues for research are also provided.

9.1 Achievement of Objectives

In this thesis, we improved our understanding of detecting fraud in e-commerce transactions by using continuous assurance systems and answered the overarching research question of this thesis by determining how eSCARF was useful as a continuous assurance system for detecting fraud in e-commerce transactions. The literature review focused on three major areas in addressing RQ1 and RQ2: the e-commerce environment, electronic fraud and continuous assurance systems. The strengths and weaknesses of e-commerce were identified, and through this, it was found that electronic fraud is a real and significant risk for businesses engaging in e-commerce today. As a result, it was important to consider methods by which businesses may manage or mitigate this risk (RQ1). If businesses are able to place more confidence in the security of e-commerce transactions, the adoption of e-commerce, and the benefits that e-commerce offers, become more attractive as the risks are controlled. Furthermore, being able to address the risk of fraud benefits auditors as well. Auditors have a responsibility, when providing assurance for business transactions and reports to consider and address the problem of fraud. Continuous assurance offers a method by which this fraud risk may be controlled. The term 'continuous assurance' refers to the process of providing assurance on information in a timely manner. As e-commerce transactions are often processed in real-time, a continuous assurance system is particularly well suited to detecting fraud as it occurs (RQ2). Such a system also provides auditors with a valuable tool for detecting fraud, enabling them to carry out their responsibilities in accordance with the Australian Auditing Standard 210 (AUS210 2002).

Activity Theory has been applied to continuous assurance and the fraud auditing of e-commerce systems in order to create a conceptual model that provides us with a generalised, holistic view of the e-commerce fraud auditing domain. By outlining the subjects, instruments, rules and other environmental factors (and the interactions of these things) involved in continuously assuring e-commerce transactions for fraud, the conceptual model enables us to better understand this domain (addressing both RQ1 and RQ2). Additionally, it places the eSCARF system within the larger context of the e-commerce fraud auditing environment, and highlights the need for its development as an instrument within the Activity Theory framework.

Continuous assurance systems that perform fraud detection do currently exist in industry, but they tend to be proprietary. Because of this, public information on how they work and how effective they are in detecting fraud is limited. This fact formed the motivation for improving our knowledge about continuous assurance systems (the motivation for asking RQ3). The rest of the thesis was directed at documenting the development and evaluation (via a survey) of such a system, called eSCARF.

eSCARF is a continuous assurance system using the SCARF audit technique. It was based on Ng and Wong's (1999) original version of eSCARF for Net.Commerce. As Net.Commerce is no longer being developed by IBM, eSCARF was successfully updated to work with the IBM WebSphere Commerce e-commerce system, a system in current use. eSCARF's primary purpose is to detect fraud occurring in e-commerce transactions by providing a mechanism by which transactions may be captured and monitored for assurance purposes. Suspected cases of fraud are reported in real-time.

Because of its role as a risk management and audit tool, the decision to implement eSCARF in business is likely to be a management initiative. It is, though, designed to be set up and used by auditors with information systems experience. eSCARF contains all the management tools necessary to set up and administer it.

eSCARF was designed to be userfriendly, highly usable and customisable to auditors using the system. However, an auditor should have a background in information systems and a knowledge of Java in order to initially integrate eSCARF into an e-commerce system, as well as customise it in more sophisticated ways such as interfacing it with external applications or modules. eSCARF is also a generalised implementation of the SCARF audit technique with a modular design which allows its functionality to be easily extended, and also to be easily modified to work with other e-commerce systems.

eSCARF was developed using a waterfall systems design life cycle, with the system's requirements detailed in section 6.3. The conceptual design documented the actual implementation process of eSCARF, including its architecture, and how all its modules work and interact with each other. Various enhancements were also made to eSCARF to increase its usefulness and capabilities, with this new functionality tested for correctness of operation. Some of these enhancements were made based on Ng and Wong's (1999) suggestions for future work, reflecting eSCARF's ongoing development. For instance, web reports were greatly improved from the elementary reports the old version of eSCARF offered. The updated reporting module allows auditors to examine data at a high level through summary reports, as well as drill-down to the transaction data level. System testing was then performed on the completed system to ensure it operated correctly, with functionality provided that corresponded to the requirements specification.

The documentation of this process addresses RQ3 and consequently furthers our understanding of how a continuous assurance system works, as well as providing information that may used when the system is further developed in the future.

eSCARF was developed by Ng and Wong (1999) without any input from system users, namely auditors, whatsoever. As user input is an important part of the development of any effective system, a user evaluation of eSCARF in the form of a survey was undertaken, with the aim of answering RQ4 and RQ5. eSCARF was demonstrated to experienced auditors and their input was received via a questionnaire in which their assessments of the system were gathered quantitatively and qualitatively.

The sample size for the survey was small, at 15 participants, but enough data was gathered for the purposes of exploratory research. The participants were all auditors with significant experience in industry, some with an information systems background, and some with only a traditional auditing background. The questionnaire asked for opinions relating to their perceptions of what attributes were important for a continuous assurance system, listing accuracy, comprehensive, conciseness, timeliness, presentation, userfriendliness and ease of customisation as attributes. The questionnaire also evaluated the individual components of eSCARF (rule management, the server console log, rule checking and alerts, and web reporting), followed by an evaluation of the whole system in order to gauge its overall usefulness.

Results from the survey were very encouraging in terms of the prospects for eSCARF's continued development and use in industry. By measuring participant perceptions, what auditors perceived as important in a continuous assurance system was ascertained (RQ5). The attributes of timeliness and accuracy were seen to be *very* important in a continuous assurance system. The result of timeliness being regarded as the most important attribute corresponds with the nature of a continuous assurance system, which is to provide assurance in a timely manner. Also answering RQ5 were the qualitative comments auditors provided which suggested extra functionality they desired in eSCARF (refer to the next section).

An evaluation of eSCARF's components found that participants tended to agreed about the usefulness and userfriendliness of each component (RQ4). The rule checking and alerts module scored the highest in the component assessment, with participants expressing strong agreement about its accuracy and timeliness. This assessment corresponds with the importance participants accorded to those two attributes.

In answering RQ4, participants viewed eSCARF favourably overall, agreeing upon its usability and general usefulness for auditors. This was signified by both the quantitative and qualitative data gathered. Interestingly, from the demonstration end-user perspective, there was no significant variation between the assessments of IS auditors and non-IS auditors, implying similar levels of understanding. While knowledge in information systems is still necessary for integrating eSCARF into an e-commerce system, once this has occurred, any auditor should be capable of using eSCARF's rule management and reporting functions.

Despite the low sample size, a correlation analysis was performed, and showed an interesting result where the attributes of timeliness and accuracy were found to correlate with the system's overall usefulness. Furthermore, when exploring if the usefulness of individual components had bearing upon the usefulness of the system as a whole, a correlation was found for the rule checking and alerts module. These results implied that what made eSCARF primarily useful was the timeliness with which it could report on information (therefore, also verifying eSCARF's status as a continuous assurance system). This shows a relationship between RQ4 and RQ5 in that there is a positive relationship between the provision of important factors (timeliness and accuracy) and overall system usefulness.

9.2 Limitations and Future Work

An information system is never truly 'complete' or 'finished'. Systems development is an ongoing, cyclical process where systems undergo a continuous cycle of development, testing, release, evaluation and then further development again as developers strive to implement improvements and bug fixes, as well as catering for the changing wants of users (Pfleeger 1998). This thesis' work on eSCARF represents only one iteration of this cycle. The evaluation survey performed determined that eSCARF was considered a useful tool by auditors, thereby encouraging its continuing development.

eSCARF received considerable positive feedback, but there are many features and improvements that can still be added into it in future versions. These proposed improvements were mentioned in section 6.6 and chapter 8. Apart from changes to aesthetics, and minor functionality modifications, the major suggestions included:

- Creating a security framework to ensure that introducing eSCARF into a business does create additional uncontrolled risks for a business. This is basically a process of ensuring that the assurance system is itself assured.
- Adding a database maintenance module, to provide for archiving of transactions. These archives may subsequently be used as test data for fine tuning rules, or for future research in fields such as data mining (after the transaction data has been sanitised so that any personal information attached to it is removed or anonymised).

- Unifying eSCARF's administration interface into a web browser. Because the use of the world wide web is so widespread, users are already familiar with the interface of a web browser. Making all of eSCARF's administrative features available through a web interface would allow centralised administration and enhance ease of use as auditors would not have to learn any new interfaces.

In the long term, what is ultimately envisioned for eSCARF is the creation of a library or suite of modules that plug in to the current eSCARF architecture. Such modules can be interchanged with others in order to easily customise eSCARF for whatever purpose it is being employed for. These modules would come in several categories:

- *Embedded audit modules* (audit hooks) for different e-commerce platforms (eg: IBM WebSphere, Oracle and iPlanet);
- *Database interface modules* so eSCARF can employ other database systems to store its data (such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, etc.);
- *Audit modules* for extending the auditing capabilities of eSCARF (eg: data mining modules, neural net algorithms and interfaces to external applications such as address verification systems or a bank's credit card verification system); and
- *Reporting modules* for extending the reporting capabilities of eSCARF (eg: pivot table views for OLAP, exporting reports to file formats in addition to the CSV format currently handled).

This modular design allows the eSCARF system to be used in different environments, with the ability to expand its functionality, all the while maintaining a standardised administration interface and architecture. For example, if an e-commerce system that needed to be assured was not IBM WebSphere but Oracle, all that would be needed to get eSCARF to work with Oracle would be to switch the audit hook used. How a module would plug-in with eSCARF is explained in figure 8. The figure is adapted from Anandarajah and Lek (2000), who designed a data mining module (an audit module) which interfaces with eSCARF.

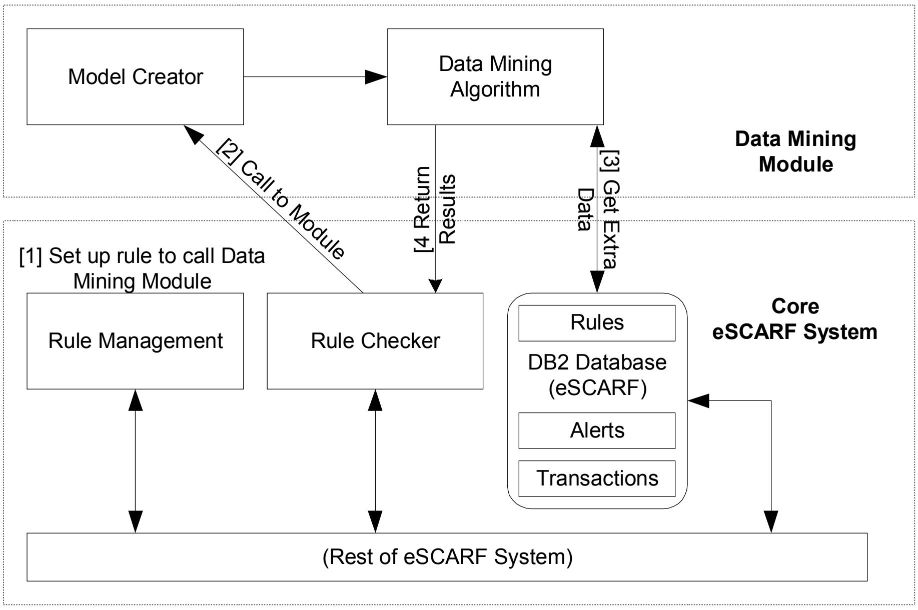


Figure 8: A Data Mining Module Interfaced with eSCARF

A data mining module may operate by taking a newly submitted transaction, performing some statistical analysis on it and eSCARF's database of past transactions, and return a score which indicates the amount of fraud risk the new transaction poses. This process could be achieved by calling the data mining module in a nodal expression ([1] in the figure above). For instance, if an auditor wanted to test for transactions with a fraud risk score of over 80%, an expression such as "`Module.Datamine > 80`" could be used. The rule checker would interpret this as a call to an external module and pass it the transaction details [2]. The data mining module would then create a mining model, perform statistical analysis on it, collecting extra data from eSCARF's database where needed [3], before returning the results as an integer score [4]. Naturally, the data mining module would need its own administration interface, but modular design means that this can be developed independently and without interference to eSCARF's code.

The major limitation of the evaluation survey performed is its low sample size and artificial setting. The evaluation, whilst enlightening, was still fairly cursory, and may be supplemented with other methods of systems evaluation in order to gain an even better assessment and understanding of eSCARF. Two other evaluative designs (both case studies) were considered for this thesis, but were unable to be carried out due to constraints noted below. It is hoped that these designs could be used in future research work regarding eSCARF or continuous assurance systems.

The first design involves a case study where eSCARF is implemented in a real world business environment. The study would document the process of integrating a continuous assurance system into a live B2C e-commerce system, including the interactions of the auditor with the business in order to determine an audit strategy. eSCARF would then be run over a period of a few weeks to see how effective it is in detecting fraud, with the data collected being periodically examined to refine the ruleset used. After this period, interviews with business personnel would establish what the business thought of eSCARF's usefulness. This would provide a more in depth response than what could be gained from a survey as the business will have had the chance to fully view and interact with eSCARF over a sufficient period of time. Unfortunately, this case study could not be performed for this thesis due to the currently low number of businesses adopting IBM WebSphere Commerce as their e-commerce system. As a result, a consenting business that used WebSphere could not be found in Sydney to use as a case study.

The other design considered was a case study of a business organisation to learn more about electronic fraud – how it occurs, its nature and its repercussions. By interviewing key personnel who deal with external fraud, and reviewing a database of past cases of fraud, it was hoped that a model of where and how fraud occurs in the business. This model would then be analysed to see where eSCARF may be integrated in order to provide fraud detection capability. The conceptual model in chapter 4 may be applied in order to aid the analysis. This case study was not carried out as management approval could not be obtained from the business for it. Nonetheless, a similar case study on an organisation would yield a better understanding of fraud and thus allow research to better address the problem of it.

Additionally, the same evaluation survey could be run, using the one in this thesis as a pilot survey. A large sample size would allow a more reliable quantitative analysis to be performed, including factor analysis to verify the validity of the questionnaire instrument.

Another point raised during the survey is that eSCARF is only as effective as its audit rules in operation. Although implementing sophisticated audit strategies in eSCARF was out of the scope of this thesis, effective strategies affect its overall ability to detect fraud. Therefore, building effective audit strategies is an important area of research. Data mining is a complementary field of research which aims at extracting meaningful information from large quantities of data using such techniques as statistical pattern detectors and neural networks. Naturally, this could be applied to the area of fraud detection, and future research adapting data mining techniques for eSCARF would increase its effectiveness and usefulness.

9.3 Closing Statement

E-commerce is a phenomenon growing in prominence and it is important to reduce the risks associated with it, especially the problem of electronic fraud. This thesis made inroads into addressing this problem with the development of eSCARF, a continuous assurance system that detects potential occurrences of fraud in e-commerce transactions. The evaluation of eSCARF found very encouraging results. Auditors thought the system was usable, useful and saw definite potential for its use in business, especially favouring its ability to assure information in a timely fashion. eSCARF's development and evaluation has demonstrated the feasibility of the system and provided the necessary information systems infrastructure for continued development of it as a valuable instrument in the e-commerce fraud auditing environment. The instrument is useful to business management, as a risk management tool for controlling fraud risk, as well as to auditors who are charged with the responsibility of considering fraud when auditing businesses. Given the significant number of suggestions collected from the 15 auditor participants, the opportunity for eSCARF to grow into a truly comprehensive fraud detection system is considerable. This thesis has laid the groundwork and created the information systems infrastructure for such future endeavours.

References

ACPR (2001), Electronic crime strategy, *Australasian Centre for Policing Research (Electronic Crime Steering Committee) Report*, March 2001

AICPA (2002), *Exposure Draft: Proposed Statement on Auditing Standards Consideration of Fraud in a Financial Statement Audit*, New York, NY: AICPA

Airports (2002), GAO outlines identity fraud risks, *Airports*, vol. 19 (27), p. 4

Alles, M. G., Kogan, A. and Vasarhelyi, M. A. (2002), Feasibility and economics of continuous assurance, *Auditing: A Journal of Practice and Theory*, vol. 21 (1), pp. 125-138

Alter, Steven (1999), *Information Systems: A Management Perspective*, 3rd edition, Addison-Wesley

Anandarajah, Benjamin and Lek, Monkol (2000), Using Data Mining to Detect E-Commerce Fraud, *Thesis (Hons)*, The University of New South Wales

ANAO (2000), Numbers on the run: Review of the ANAO Report No. 37 1998-99 on the Management of Tax File Numbers, *Australian Parliament Standing Committee on Economics, Finance and Public Administration Report*, August 2000

Armstrong, C. M. (2000), The Internet and E-Commerce, *White Paper* [http://www.ipservices.att.com/techviews/whitepapers/InternetE-Commerce.pdf] (last accessed: 21 November 2002)

AUS210 (2002), *Auditing Standard 210: The Auditor's Responsibility to Consider Fraud and Error in an Audit of a Financial Report*, Australian Accounting Research Foundation

AUSTRAC (2001), *Cost of Identity Fraud*, AUSTRAC Steering Committee on Proof of Identity

Baer, C. (2002), Incorporating a Forensic-Type Accounting Phase into the Financial Statement Audit: A Critical Analysis, *IIA Educator*, October 2002 [http://www.theiia.org/newsletter/index.cfm?news_id=209] (last accessed: 26 November 2002)

Bandura, A. (1997), *Self-efficacy: The Exercise of Control*, New York: Freeman

Bannon, Liam (1997), Activity Theory [http://www-sv.cict.fr/cotcos/pjs/TheoreticalApproaches/Actvity/ActivitypaperBannon.htm] (last accessed: 21 November 2002)

Blaikie, N. (2000), *Designing Social Research*, Oxford: Polity Press

Blumer, H. (1969), *Symbolic Interactionism: Perspective and Method*, Englewood Cliffs, NJ: Prentice-Hall

Brunker, Mike (2002), EBay's tough talk on fraud doesn't withstand scrutiny, *MSNBC*, 9 October 2002 [http://www.msnbc.com/news/809148.asp] (last accessed: 21 November 2002)

Cabinet (2002), Identity Fraud: A Study, *United Kingdom Cabinet Office Report*, July 2002

Camtech (1999), Merchant Server and Payment Gateway Functional Specification, *White Paper* [http://www.camtechcorporation.com/ecommerce/function.pdf] (last accessed: 21 November 2002)

Cerpa, N. and Jamieson, R. (2001), A Security Trust and Assurance Research Framework for Electronic Commerce, *Proceedings of the IFIP TC8 Working Conference on Electronic Commerce*, Salzburg, Austria 22-23 June 2001 (in Press)

Chen, M., Han, J. and Yu, P. (1996), Data Mining: An Overview from a Database Perspective, *IEEE Transactions on Knowledge and Data Engineering*, vol. 8 (6)

CICA (1999), *Continuous Auditing Research Report*, Toronto: The Canadian Institute of Chartered Accountants

Clarke, Roger (1993), EDI is but one element of electronic commerce, *6th International EDI Conference*, Bled, Slovenia June 1993

Clarke, Roger (1996), The SET Approach to Net-Based Payments [http://www.anu.edu.au/people/Roger.Clarke/EC/SETOview.html] (last accessed: 21 November 2002)

Clarke, Roger (2000), Appropriate Research Methods for Electronic Commerce, [http://www.anu.edu.au/people/Roger.Clarke/EC/ResMeth.html] (last accessed: 21 November 2002)

Comer, D. E. (2000), *Networking with TCP/IP*, 4th edition, Upper Saddle River, NJ: Prentice-Hall

CREC (2001), *Measuring the Internet Economy*, CREC, January 2001 [http://www.internetindicators.com/jan_2001.pdf] (last accessed: 21 November 2002)

Crotty, M. (1998), *The Foundations of Social Research*, Sydney: Allen and Unwin

Daigle, R. G. and Lampe, J. C. (2000), Determining the Market Demand for Continuous Online Attestation: Preliminary Work on a Ph.D. Dissertation, *2nd Continuous Auditing and Reporting Conference*, Newark 28 January 2000

Deloitte (1999), Computer Crime and Security Survey, Victoria Computer Crime Squad and Deloitte Touche Tohmatsu, Melbourne

Delzoppo, G. A., Mulholland, M., Hibber, D. B. (1993), A novel application of ripple down rules to selecting a method of chemical analysis for a variety of chemicals and their sample matrices, *1993 Australian Joint Conference on AI*, Melbourne 16-19 November 1993

Dertouzos, M. (2000), The Unfinished Revolution, *Andersen Consulting Outlook*, vol. 12 (2), Andersen Consulting

Devore, J. L. (2000), *Probability and Statistics for Engineering and the Sciences*, Pacific Grove, CA: Duxbury

Dey, I. (1993), *Qualitative Data Analysis: A User Friendly Guide for Social Scientists*, London: Routledge

Doll, W. J. and Torkzadeh, G. (1991), The measurement of end user computing satisfaction: Theoretical and methodological issues, *MIS Quarterly*, vol. 15, pp. 5-10

Donnelly, H. (2002), Credit industry combats growing losses to card skimming, *Stores*, vol. 84 (1), pp. 110-111

Doocey, P. (2002), Identity Crisis, *Bank Systems and Technology*, vol. 39 (6), p. 6

Duh, R. R., Jamal, K. and Sunder, S. (2001), Control and Assurance in E-Commerce: Privacy, Integrity and Security at eBay, *Working Paper*, 24 April 2001

Elliot, S. and Fowell, S. (2000), Expectations versus reality: a snapshot of consumer experiences with Internet retailing, *International Journal of Information Management*, vol. 20, pp. 323-336

Elliott, R. K. (2001), Introductory Remarks of Robert K. Elliott, *3rd Continuous Auditing and Reporting Conference*, Newark 1 June 2001

Elliott, R. K. (2002), Twenty-First Century Assurance, *Auditing: A Journal of Practice and Theory*, vol. 21 (1), pp. 139-146

Engeström, Y. (1987), *Learning by expanding: An activity-theoretical approach to developmental research*, Helsinki: Orienta-Konsultit

Engeström, Y. (1998), Cultural-Historical Activity Theory, [http://www.edu.helsinki.fi/activity/6a.htm] (last accessed: 21 November 2002)

Farhoomand, A. F. and Etezadi-Amoli, J. (1991), On end-user computing satisfaction, *MIS Quarterly*, vol. 15 (1), pp. 1-4

Frazer, L. and Lawley, M. (2000), *Questionnaire Design and Administration: A practical guide*, Milton, QLD: John Wiley & Sons

FTC (2000), Identity Theft Complaint Data: Figures and Trends on Identity Theft, January 2000 through December 2000, US Federal Trade Commission Report

FTC (2000b), Prepared statement on Identity Theft for a hearing before the House Committee on Banking and Financial Services, US Federal Trade Commission Report, 13 September 2000

Furnell, Steven M. and Dowland, Paul S. (2000), A conceptual architecture for real-time intrusion monitoring, *Information Management and Computer Security*, vol. 8 (2), pp. 65-74

Galetta, D. F. and Lederer, A. L. (1989), Some cautions on the measurement of user information satisfaction, *Decision Sciences*, vol. 20, pp. 25-34

Garceau, L. (1998), Auditing of Computerized Information Systems [http://www.csuohio.edu/accounts/AUDIT/] (last accessed: 21 November 2002)

Garner, B. A. (1999), Fraud, *Black's Law Dictionary*, 7th edition, St Paul, MN: West Group

Gay, Grant and Simnett, Roger (2000), *Auditing and Assurance Services In Australia*, Sydney: McGraw-Hill

Gelderman, M. (1998), Translation and validation of the Doll and Torkzadeh end user computing satisfaction instrument, In: Hugh J. Watson (Ed.), *Proceedings of HICSS 31, vol 6*, IEEE Computer Press, pp. 537-546

Gengler, B. (2002), PayPal's anti-fraud team, *Computer Fraud and Security*, vol. 2002 (3), p. 5

Geralds, J. (2002), E-shoppers still concerned about security, *Computer Fraud and Security*, vol. 2002 (2), p. 7

Glover, H. D. and Schleifer, L. F. (1995), Continuous Auditing: An idea whose time has come, *Corporate Controller*, pp. 30-34

Goodwin, N. C. (1987), Functionality and Usability, *Communications of the ACM*, vol. 30 (3), pp. 229-333

Groth, R. (2000), *Data Mining: Building Competitive Advantage*, Upper Saddle River, NJ: Prentice-Hall

Gul, F. A., Teoh, H. Y. and Andrew, B. H. (1991), *Theory and Practice of Australian Auditing*, 2nd edition, Sydney: Thomas Nelson

Hair, J. F., Anderson, R. E., Tatham, R. L. and Black, W. C. (1998), *Multivariate Data Analysis*, 5th edition, Upper Saddle River, NJ: Prentice-Hall

Harrison, A. W. and Rainer, R. K. (1996), A General Measure of User Computing Satisfaction, *Computers in Human Behaviour*, vol. 12 (1), pp. 79-92

Hasan, H. and Handzic, M. (2003), “Integrated KM Frameworks”, in Handzic, M. and Hasan, J., *Studies in Knowledge Management*, Australia: John Wiley & Sons, (book, forthcoming)

Hempel, C. E. (1966), *Philosophy of Natural Science*, Engelwood Cliffs, NJ: Prentice-Hall

Hendrickson, A., Glorfield, T. and Cronan, T. (1994), On the repeated test-retest reliability of the end-user computing satisfaction instrument: A comment, *Decision Sciences*, vol. 25 (4), pp. 655-667

IBM (2002), *IBM WebSphere Commerce Programmer's Guide: Version 5.4*, available from IBM Corporation

IFCC (2001), *IFCC 2001 Internet Fraud Report*, Internet Fraud Complaint Center [http://www1.ifccfbi.gov/strategy/IFCC_2001_AnnualReport.pdf] (last accessed: 21 November 2002)

Internet Fraud Watch (2001), 2001 Internet Fraud Statistics [http://www.fraud.org/internet/2001stats.htm] (last accessed: 21 November 2002)

InternetNews (1999), Visa, Cybersource launch new e-commerce fraud detector, *InternetNews.com* [http://www.internetnews.com/ec-news/article.php/4_193411] (last accessed: 21 November 2002)

Joppe, M. (2002), The Research Process, [http://www.ryerson.ca/~mjoppe/ResearchProcess/] (last accessed: 21 November 2002)

Kane, M. (2002), Will techs gain from auditor scrutiny?, *CNet News.com* [http://news.com.com/2100-1017-847222.html] (last accessed: 21 November 2002)

Kang, B. H., Preston, P. and Compton, P. (1998), Simulated expert evaluation of multiple classification Ripple Down Rules, *Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-based System Workshop*, Banff 18-23 April 1998

Kanter, H. A. (2001), Systems Auditing in a Paperless Environment, *Ohio CPA Journal*, 1st Quarter 2001 [http://www.ohioscpa.com/member/publication/Journal/1st2001/page7.asp] (last accessed: 21 November 2002)

Kennedy, D. (2000), Online credit card use draws blood, *Sydney Morning Herald*, 8 August 2000

Koch, H. S. (1981), Online Computer Auditing Through Continuous and Intermittent Simulation, *MIS Quarterly*, March 1981, pp. 29-41

Koch, H. S. (1984), Auditing On-Line Systems: An Evaluation of Parallel versus Continuous and Intermittent Simulation, *Computers and Security*, February 1984, pp. 9-19

KPMG (1999), 1999 Fraud Survey, KPMG, Sydney

Leyden, J. (2000), Blackmailer posts credit card details on the Net, *The Register*, 13 December 2000 [http://www.theregister.co.uk/content/archive/15446.html] (last accessed: 21 November 2002)

MacVittie, L. (2002), Online fraud detection takes diligence, *Network Computing*, vol. 13 (4), pp. 80-83

Maher, William (2002), Hacking: The big bucks industry, *APC*, April 2002 (280), pp. 58-64

Malhotra, N. K. (1996), *Marketing Research: An Applied Orientation*, 2nd edition, Englewood Cliffs, NJ: Prentice-Hall

Mappin, David A. (1999), Activity Theory, [http://www.quasar.ualberta.ca/edpy597/Modules/module15.html] (last accessed: 21 November 2002)

Marlin, Steven (1999), Visa develops Internet gateway, *Bank Systems and Technology*, vol. 36 (8), p. 22

Martin, C. (2002), Security and E-Commerce, *Presentation at the University of New South Wales*, 29 April 2002

Matejkovic, J. E. and Lahey, K. E. (2001), Identity Theft: No Help for Consumers, *Financial Services Review*, vol. 115, pp. 1-15

Mertl, S. (2000), Internet Security: A Growing Problem, *Canadian Press* in *E-CommerceAlert.com*, 22 January 2000

Mesmer, Ellen (2000), Online card fraud targeted, *Network World*, 21 August 2000

Mohrweis, L. (1988), Usage of Concurrent EDP Audit Tools, *The EDP Auditor Journal*, vol. 3, pp. 49-54

Myers, M. D. (1997), Qualitative Research in Information Systems, *MIS Quarterly*, vol. 21 (2), June 1997, pp. 241-242

Nardi, B. A. (1996), *Context and Consciousness: Activity theory and human-computer interaction*, Cambridge: MIT Press

Nehmer, Rob (2000), Agents for Continuous Auditing, *2nd Continuous Auditing and Reporting Conference*, Newark 28 January 2000

Newsted, P. R., Chin, W., Ngwenyama, O. and Lee, A. (1996), Resolved: Surveys have Outlived their Usefulness in IS Research, Panel presented at the *1996 International Conference on Information Systems*, Cleveland, Ohio, 17 December 1996

Newsted, P., Huff, S. and Munro, M. (1998), Survey Instruments in IS, *MIS Quarterly*, vol. 22 (4), pp. 553-554

Nezu, R. (2000), E-commerce: a revolution with power, *OECD Observer* [http://www.oecdobserver.org/news/printpage.php/aid/344/E-commerce:_a_revolution_with_power.html] (last accessed: 21 November 2002)

Ng, Brenda and Wong, Keith (1999), An Audit Review for Electronic Commerce, *Thesis (Hons)*, The University of New South Wales

Nguyen, D. (2002), Identity Theft: Perceptions and Problems, *Working Paper*, 15 October 2002

Nielsen, J. (1998), What is Usability?, *ZDNet DevHead,* [http://www.zdnet.com/filters/printerfriendly/0,6061,2137671-74,00.html] (last accessed: 21 November 2002)

O'Brien, I. and Mercer, P. (2002), Internet Security Risks and Exposures: Insights, Trends and Mitigation Strategies, *Presentation at the University of New South Wales*, 15 April 2002

Pajares, F. (1997), Current Directions in Self-efficacy Research, In: M. Maehr & P. R. Pintrich (Eds.), *Advances in motivation and achievement, vol 10*, Greenwich, CT: JAI Press, pp. 1-49

Pastore, M. (2002), U.S. E-Commerce Spikes in Q4 2001, *Internet.com* [http://cyberatlas.internet.com/markets/retailing/article/0,,6061_977751,00.html] (last accessed: 21 November 2002)

Pfleeger, S. L. (1998), *Software Engineering: Theory and Practice*, Upper Saddle River, NJ: Prentice-Hall

Pure Commerce (2002), Pure Commerce Fraud Detection, [http://www.purecommerce.com.au/e-com-services-risk-fraud.asp] (last accessed: 21 November 2002)

Rezaee, Z., Sharbatoghlie, A., Elam, R and McMickle, P. L. (2002), Continuous Auditing: Building Automated Auditing Capability, *Auditing: A Journal of Practice and Theory*, vol. 21 (1), pp. 147-163

Royce, W. W. (1970), Managing the development of large software systems: Concepts and techniques, *Proceedings of WESCON*, August 1970

Schonlau, M., Fricker, R. D. and Elliott, M. N. (2001), *Conducting Research Surveys via E-mail and the Web*, Santa Monica, CA: Rand

Schreft, Stacey L. (2002), Clicking with dollars: How consumers can pay for purchases from e-tailers, *Economic Review - Federal Reserve Bank of Kansas City*, vol. 87 (1), First Quarter 2002, pp. 37-64

Shields, Greg (1998), Non-stop auditing, *CA Magazine*, September 1998, pp. 39-40

Smith, R. (1999), Fraud: What response?, *CPA Australia*, November 1999, p. 39

Smith, W. C. (2001), Patent this, *ABA Journal*, vol. 87, pp. 48-57

Straub, D. W. (1989), Validating instruments in MIS research, *MIS Quarterly*, vol. 13, pp. 147-169

Turban, E., Lee, J., King, D. and Chung, H. M. (2000), *Electronic Commerce: A Managerial Perspective*, Upper Saddle River, NJ: Prentice-Hall

Udo, G. J. (2001), Privacy and security concerns as major barriers for e-commerce: a survey study, *Information Management and Computer Security*, vol. 9 (4), pp. 165-174

Vacca, J. (1995), CommerceNet: Open for Business, *Network World*, vol. 12 (35), pp. 18-20

van Krugten, P. and Hoogenboom, M. (2000), B2C Security – Be Just Secure Enough, *Computers and Security*, vol. 19 (4), pp. 348-356

Vasarhelyi, M. A., Kogan, A., Sudit, E. F. (2000), Continuous Online Auditing: A Program of Research, *2nd Continuous Auditing and Reporting Conference*, Newark 28 January 2000

Visa (2002), Verified by Visa: How does it work? [http://www.visa.com.au/verified/merchants/how.shtml] (last accessed: 21 November 2002)

Vygotsky, L. S. (1978), *Mind in society: the development of higher psychological processes*, Cambridge: Harvard University

Wang, Wenli, Bailey, Andrew D., Hidvégi, Zoltán and Whinston, Andrew B. (2000), A Framework for Proactive, Automated and Continuous E-commerce Control and Assurance, *Working Paper*, 17 September 2001

Wang, Wenli, Hidvégi, Zoltán and Whinston, Andrew B. (2000), Designing Secure Mechanisms for Online Processes, *Proceedings of the International Conference on Electronic Commerce 2000*, Seoul, Korea, August 2000, p. 312-318

Wang, Wenli, Hidvégi, Zoltán and Whinston, Andrew B. (2001), Designing Mechanisms for E-Commerce Security: An Example from Sealed-bid Auctions, *International Journal of Electronic Commerce* (forthcoming)

Wang, Wenli, Hidvégi, Zoltán and Whinston, Andrew B. (2001b), Shill Bidding in English Auctions, *Working Paper*, 6 September 2001

Weaver, A. C., Vetter, R. J., Whinston, A. B. and Swigger, K. M. (2000), Guest Editors' Introduction: The Future of E-Commerce, *IEEE Computer*, vol. 33 (10), pp. 30-31

Weber, R. (1999), *Information Systems Control and Audit*, Upper Saddle River, NJ: Prentice Hall

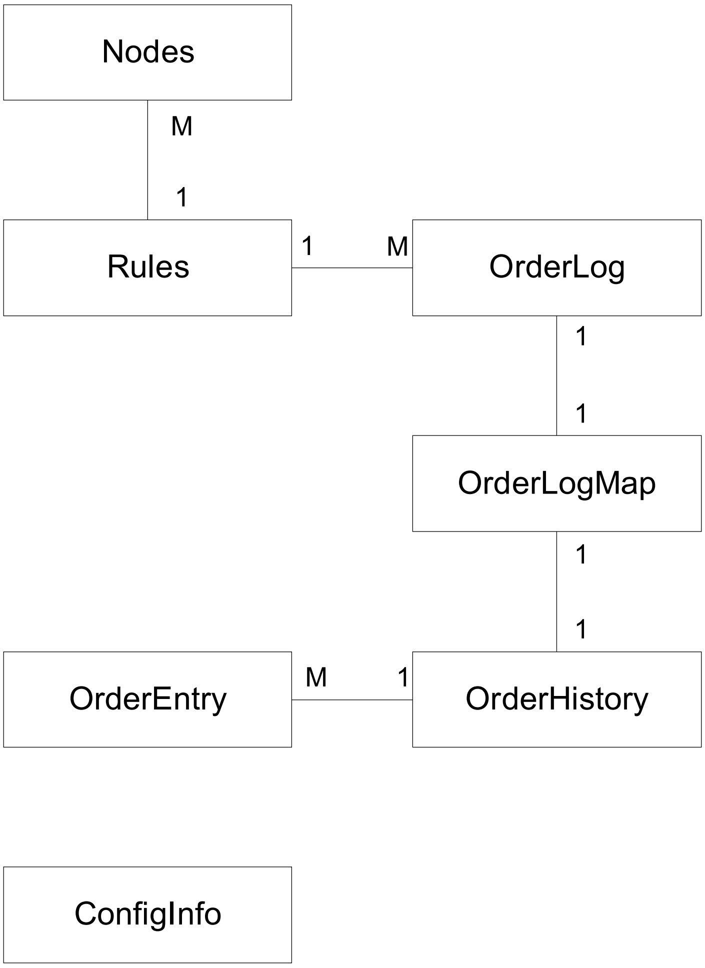
Weinberg, B. A. (2002), A model of overconfidence, *Working Paper*, March 2002 [http://economics.sbs.ohio-state.edu/weinberg/conf_mar02.pdf] (last accessed: 21 November 2002)

Yap, C. S. and Thong, J. Y. L. (1996), Information Systems Effectiveness: A User Satisfaction Approach, *Information Processing and Management*, vol. 32 (5), pp. 601-610

Yeomans, A. (2001), Learning a new language: The transition to XML, *Document Processing Technology*, vol. 9 (3), pp. 28-29

Appendices

Appendix 1: eSCARF Database Entity Relationship Diagram



Note that attributes for entities (that is, the fields for the database tables) have not been shown on this diagram. Please refer to the next appendix for the metadata.

Appendix 2: eSCARF Database Metadata (Data Dictionary)

PRI = Primary Key
FK = Foreign Key
NN = Not Null

Column Name	*Data Type*	*Properties*	*Description*
RULES Contains data for all the rules in the eSCARF ruleset			
rule_id	integer	PRI, NN	The primary key ID used to identify a rule.
name	varchar(30)	NN	The name of the rule.
node_start	integer	FK (Nodes)	Links to the ID of the node this rule starts processing from.
days	smallint		The days part of the time period for this rule.
hours	smallint		The hours part of the time period for this rule.
mins	smallint		The minutes part of the time period for this rule.
active	smallint		The active status of the rule. 0 = inactive; 1 = active
triggered	smallint		The triggered status of a rule, changed if any of the nodes for this rule has their expression evaluated as *true*. 0 = not triggered yet; 1 = triggered
version	smallint		The version number of the rule. New rules begin numbering at 1. New versions are created whenever a rule is triggered and has to be changed .
deprecated	smallint		Indicates a rule is no longer the latest version. The latest version always has a value of 0; else, 1.
last_update	timestamp		The timestamp of when this rule was last modified.
NODES Contains data for all the nodes used in the eSCARF ruleset. Nodes has a composite primary key.			
rule_id	integer	PRI, FK (Rules), NN	The ID of the rule this node belongs to.
node_id	integer	PRI, NN	The ID number of this node.
x_pos	smallint		The x- and y-coordinates of the node box on the rule management (rule designer) screen.
y_pos	smallint		
width	smallint		The width and height of the node box on the rule management (rule designer) screen, in pixels.
height	smallint		
rule	varchar(500)		The rule expression for this node.
alertlevel	smallint		The alert level for this node, triggered if the rule expression evaluates as true.
node_true	smallint	FK (Nodes)	Optional reference to another node, down the true path of this node.
node_false	smallint	FK (Nodes)	Optional reference to another node, down the false path of this node.
tstart_x	smallint		The x- and y-coordinates of the start of a line representing a true path.
tstart_y	smallint		
tend_x	smallint		The x- and y-coordinates of the end of a line

tend_y	smallint		representing a true path.
fstart_x	smallint		The x- and y-coordinates of the start of a line representing a false path.
fstart_y	smallint		
fend_x	smallint		The x- and y-coordinates of the end of a line representing a false path.
fend_y	smallint		

ORDERHISTORY
Stores all the historical transaction data captured by the eSCARF server.

trans_id	integer	PRI, NN	The primary key ID used to identify a transaction.
order_no	integer	NN	The order reference number the e-commerce system (WebSphere) uses.
login	varchar(32)	NN	The login username, or user ID of the person who placed the order.
merchant_no	integer	NN	The store ID from which the order came.
payment_type	varchar(5)		Credit card type (Visa, MasterCard, etc.)
card_no	varchar(64)		The credit card number.
card_exp	varchar(4)		The credit card expiry date in the format: MMYY.
order_success	smallint	default: 1	Denotes if the order was successful. Does not apply for WebSphere Commerce, defaults to 1, indicating success. Otherwise, 0.
log_timestamp	timestamp		The timestamp of when this transaction record was first inserted into this table.
last_used	timestamp		The timestamp of the last time this transaction was included in OrderLog. That is, there is a record in the OrderLog table with the same timestamp. This is used for cascading deletes.

ORDERENTRY
Stores the items listing (the products ordered) for a transaction in the OrderHistory table.

trans_id	integer	PRI, FK (OrderHistory), NN	The foreign key used to identify what transaction this record belongs to.
product_id	integer	PRI, NN	The product ID number of the product bought from the WebSphere Commerce system.
price	integer		The price of the product in *cents*.
quantity	smallint		The quantity of the product ordered.

ORDERLOG
Stores the log of transactions triggering alerts that log to the database.

transaction_id	integer	PRI, NN	The primary key used to identify a transaction which triggered an alert.
rule_id	integer	FK (Rules), NN	The foreign key used to identify what rule this transaction triggered.
login	varchar(32)	NN	The login username, or user ID of the person who placed the order.
merchant_no	integer	NN	The store ID from which the order came.
rulenode	varchar(500)		The rule expression from the node which triggered the rule.
order_success_cnt	smallint		Deprecated for WebSphere Commerce. Left in for e-commerce systems that may be able to implement this.
order_fail_cnt	smallint		Deprecated for WebSphere Commerce. Left in for e-commerce systems that may be able to implement this.
total_amt	integer		The total amount (price * quantity) of *all* orders made during the time period for the rule

			triggered.
total_qty	smallint		The total quantity of *all* orders made during the time period for the rule triggered.
log_timestamp	timestamp		The timestamp of when this transaction record was first inserted into this table.

ORDERLOGMAP

A simple table which matches transactions in the OrderLog table, to those in the OrderHistory table.

transaction_id	integer	PRI, FK (OrderLog), NN	The foreign key used to identify the transaction ID from the OrderLog table.
trans_id	integer	PRI, NN	The transaction ID from the OrderHistory table which corresponds to the same transaction in the OrderLog table.

CONFIGINFO

Stores miscellaneous configuration information for eSCARF

alert_mail_dest	varchar(50)		Contains an e-mail address. Used when an alert which generates an e-mail is triggered (configured as alert level 3).
smtp_server	varchar(50)		The SMTP server address to be used when sending an alert e-mail to the e-mail address in *alert_mail_dest*.
scarf_server_port	integer	default: 10002	The port the eSCARF server listens on for transactions sent in by the eSCARF audit hook(s).

Appendix 3: eSCARF Java Package Hierarchy

Bold, underlined entries represent package names. All other entries are Java classes.

scarf			
	scarf		eSCARF main menu
	scarf.commonobj		
		DecisionNode	Represents a rule node
		EventItem	
		MessageBox	
		OnlineAlert	
		TEA	TEA encryption algorithm class
		RuleSet	
		VariableStates	
	scarf.db		
		DBConn	Database connection library class
	scarf.ruleact		
		RuleActivator	
	scarf.rulechecker		
		OrderRuleProcessor	
		ProductOrder	
		RuleChecker	
		RuleCheckerDisplay	
		RuleProcessor	
	scarf.rulemaker		
		ActivePrompt	
		ChangeName	
		CreateNewVersion	
		DecisionDesignArea	
		DecisionNodeGUI	
		DeleteRule	
		DesignInternalFrame	
		OpenRule	
		OplistListener	
		OrderRuleValidator	
		PointUtils	
		RuleDesigner	
		RuleValidator	
		SaveQuery	
		SaveRule	
		SetFlags	
		SetRule	
		SetTimeInterval	
		VarlistListener	
		VersionChoice	
		VersionQuery	
	scarf.scarfserver		
		EventLogger	
		LogEventBuffer	
		OrderEventLogger	
		ScarfServer	
com.ibm.commerce.order.commands			
	HookOrderProcessCmdImpl		Embedded audit module (audit hook)

Appendix 4: eSCARF Installation Instructions

Install the eSCARF Files

1. Copy all files and subdirectories to: `<drive>:\scarf`
2. Ensure that the path to the Java Development Kit's bin directory is set. For example, the path can be set by entering the following at a command line:

```
set path=%PATH%;<drive>:\websphere\appserver\java\bin\
```

Create eSCARF Database

3. At a command line, type the following:

```
<drive>:\scarf\newdb.bat <name of WebSphere store database>
```

Adding the Audit Hook (Embedded Audit Module)

4. Copy the hook task command into WebSphere by copying

```
<drive>:\scarf\hook\wcsorder.jar
```

to

```
<drive>:\websphere\appserver\installedapps\<InstanceName>.ear\lib\
```

5. Restart WebSphere Application Server to ensure the changes are loaded.

Install eSCARF Reporting

6. To copy in web reports, copy all the files from

```
<drive>:\scarf\webreports\
```

to an existing .war WebSphere web application (eg: examples.war/)

To Run

1. Double-click the following file:

```
<drive>:\scarf\runscarf.bat
```

ADDITIONAL NOTES

name of WebSphere store database refers to the database node name of where WebSphere Commerce stores its information for the e-commerce store being audited. (eg: MALL)

InstanceName refers to the name of the instance of WebSphere Commerce being audited. (eg: demo)

For basic operating instructions and user manual, refer to: Ng and Wong (1999, appendix A).

Appendix 5: eSCARF File Details

Directory	File(s)	Description
/	scarf.jar	JAR (Java Archive) file containing the eSCARF Java program.
	runscarf.bat	Batch file used to start eSCARF running.
/hook/	wcsorder.jar	WebSphere Commerce JAR with the eSCARF audit hook added.
/hook/source/	*.java	Source code for the eSCARF audit hook.
/setup/	newdb.bat newdb2.bat	Creates a new eSCARF database for a DB2 database.
	refreshdb.bat refreshdb2.bat	Empties and resets the entire eSCARF database.
/source/	*.java	Source files for the eSCARF Java application.
	*.class	Compiled source files for eSCARF.
/sql/	audhook.sql	SQL used to add in the call to the audit hook to WebSphere.
	refreshdb.sql	SQL used by refreshdb.bat to empty and reset the eSCARF database.
	setupdb.sql	SQL used by newdb.bat to create and initialise a new eSCARF database.
/webreports/	*.java	Source files for the Java Beans used in the web reports.
	*.class	Compiled Java Beans.
	*.jsp	Java Server Pages which generate the web reports.
	*.html	Plain web pages which support the web reports.

These files are found in the eSCARF directory of the eSCARF CD. Please see appendix 10 for more details.

Appendix 6: Pilot Questionnaire

EVALUATION SURVEY FOR THE ESCARF FRAUD DETECTION CONTINUOUS ASSURANCE SYSTEM

Survey Overview and Introduction

This survey is designed to gather evaluation data of a software system called eSCARF (Electronic System Control Audit Review File) with regards to its usability and usefulness from an auditor's perspective. eSCARF is a continuous assurance system designed to detect fraud occurring in e-commerce system transactions. Firstly, some background information regarding continuous assurance and how eSCARF operates follows, in order to help you understand the eSCARF system. Secondly, a walkthrough demonstration of the eSCARF system will be presented to you. After this walkthrough (questions after walkthrough?), you will need to fill out the attached questionnaire. Please answer the questionnaire questions from an auditor's perspective, using your expertise in auditing.

What is Continuous Assurance?

Continuous Assurance is a rapidly developing field of research which extends upon the traditional accounting practice of auditing. Continuous Assurance can be defined as a methodology that permits independent auditors to provide assurance on subject matter, for which an organisation's management is responsible, using a series of assurance reports issued simultaneously (or a short period after) the occurrence of events underlying the subject matter. In other words, continuous assurance provides the ability for auditors to assure all types of information shortly after the occurrence of the events being assured.

Auditing has traditionally been performed on financial and accounting data to attribute a measure of trust and assurance to it. However, auditing can be performed on all types of data (called "assurance" when auditing is used in a non-financial context). A lot of non-financial data, though, is of a more time critical nature than financial data (which tends to be periodic), meaning that in the past it has not been practical to assure such data, due to the time that must be invested in the auditing process. A level of automation must be introduced in order to make the auditing process more timely, and thus allow all types of data to be audited. This idea is called continuous assurance.

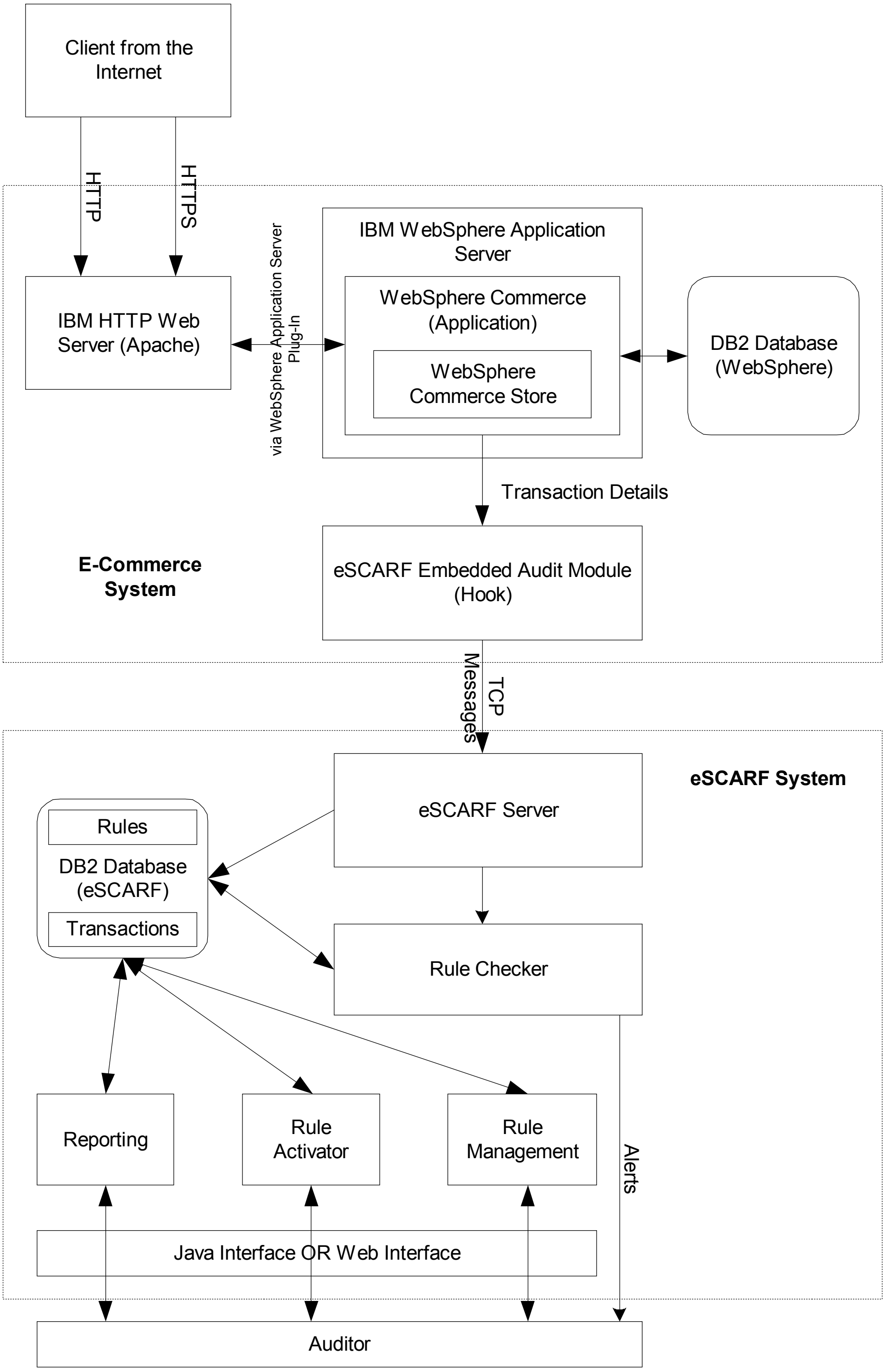
The advent and increasing use of information systems in society has allowed continuous assurance to be implemented, due to the data being assured being stored and managed on information systems, as well as auditing procedures being able to be automated via computers. Therefore, through the proliferation of information systems, it is now feasible to implement continuous assurance processes in organisations. The development of continuous assurance systems is particularly useful, as all types of data, especially data important to decision making in the organisation, can be assured.

eSCARF is an implementation of such a continuous assurance system. eSCARF assures e-commerce transactions, with the aim of detecting fraudulent transactions.

eSCARF: How it Works

eSCARF, as with all continuous assurance systems, work concurrently with the system they are assuring. eSCARF is designed to provide assurance for B2C e-commerce stores, with this version of eSCARF designed for IBM WebSphere Commerce 5.4, one such e-commerce system. eSCARF basically examines, in real-time, e-commerce transactions (an order placed by a customer) as they are made. Each transaction is compared to a set of rules that an auditor designs with the view to detect fraud. The rules allow eSCARF to look at attributes of the transaction currently being examined, as well as the attributes of previous transactions.

The following diagram shows the architecture of eSCARF:

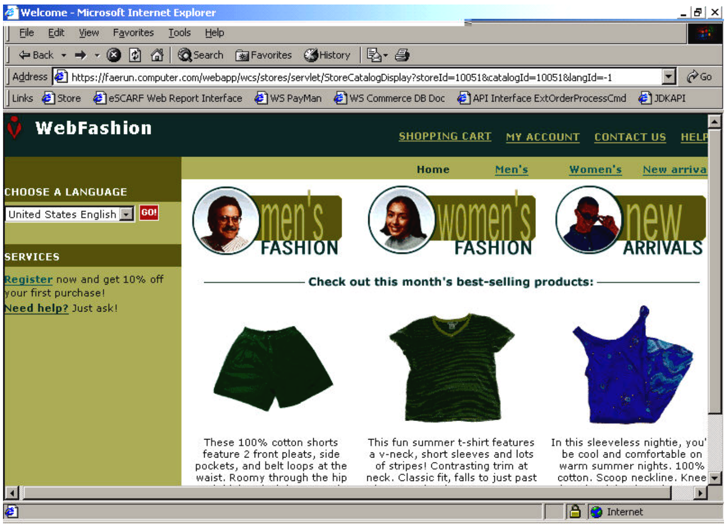


Because continuous assurance systems are designed to be as unintrusive as possible to the systems they assure, eSCARF is a distinct and separate system. It “hooks” into IBM WebSphere Commerce at one point only, where it captures transaction details and sends them to the eSCARF server. The server passes the transaction to a rule checker, which logs the transactions to the eSCARF database and produces any alerts if fraud is suspected.

Auditors manage the eSCARF system, with modules provided for each task they need to perform, all accessible via a main menu:

- Rule Management allows auditors to create and maintain audit rules which are aimed at detecting fraud.
- Reporting allows real-time reports to be generated. Auditors use these reports to analyse the data eSCARF collects. The reports are designed to provide information in summary form with the ability to drill-down for more detail.
- Alerts can be produced in response to a transaction which is highly suspicious. These alerts can consist of an onscreen alert, or even an SMS sent to an auditor’s mobile phone.
- The eSCARF server can be started and stopped.

The following screenshots illustrate various parts of eSCARF in action:


Welcome - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
WebFashion
SHOPPING CART MY ACCOUNT CONTACT US HELP
Home Men's Women's New arriva
CHOOSE A LANGUAGE
United States English
GO!
SERVICES
men's FASHION
women's FASHION
new ARRIVALS
Internet

https://faerun.computer.com/webapp/wcs/stores/servlet/StoreCatalogDisplay?storeId=10051&catalogId=10051&langId=-1

Register now and get 10% off your first purchase!
Need help? Just ask!

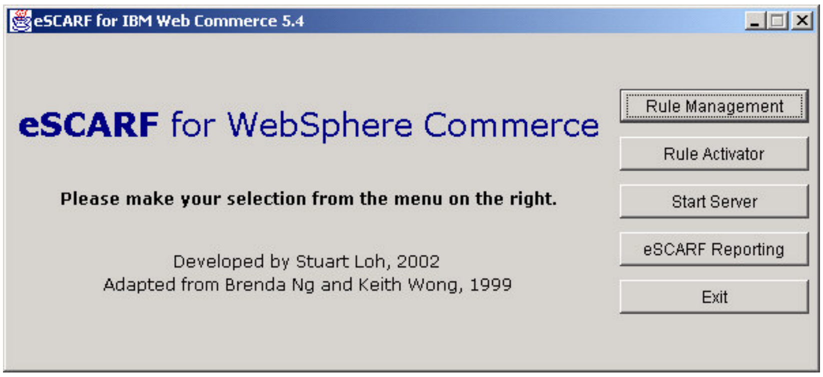
Check out this month's best-selling products:

These 100% cotton shorts feature 2 front pleats, side pockets, and belt loops at the waist. Roomy through the hip

This fun summer t-shirt features a v-neck, short sleeves and lots of stripes! Contrasting trim at neck. Classic fit, falls to just past

In this sleeveless nightie, you' be cool and comfortable on warm summer nights. 100% cotton. Scoop neckline. Knee

1. WebFashion is the e-commerce store running on IBM WebSphere Commerce that will be audited.

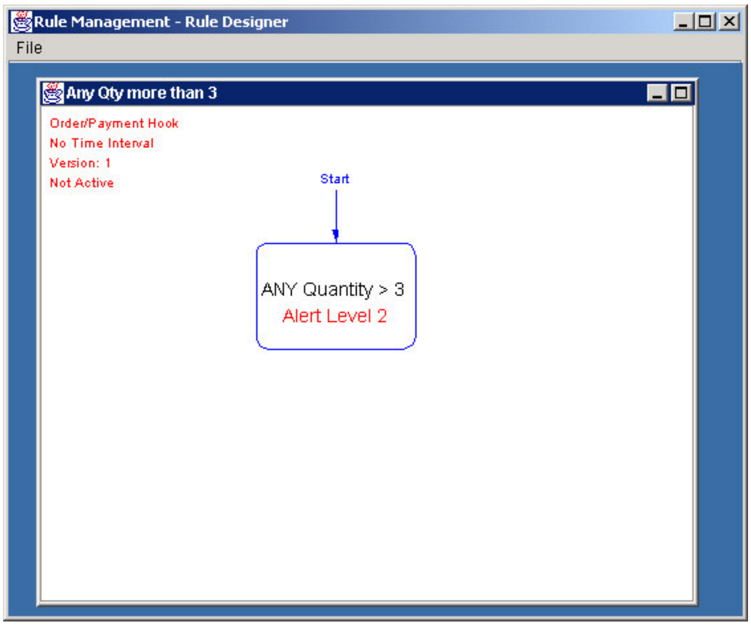


eSCARF for WebSphere Commerce

Please make your selection from the menu on the right.

Developed by Stuart Loh, 2002
Adapted from Brenda Ng and Keith Wong, 1999

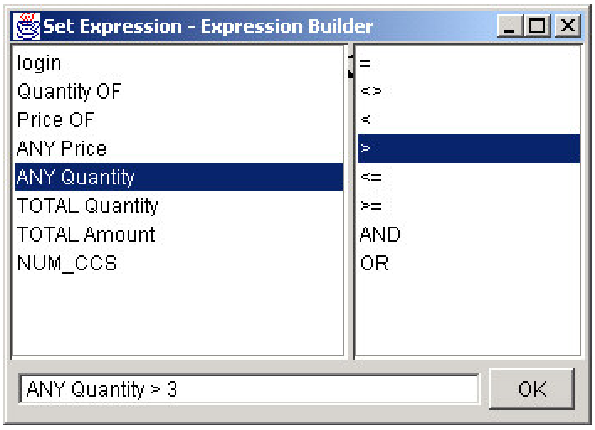
2. The main menu of eSCARF.



Order/Payment Hook
No Time Interval
Version: 1
Not Active

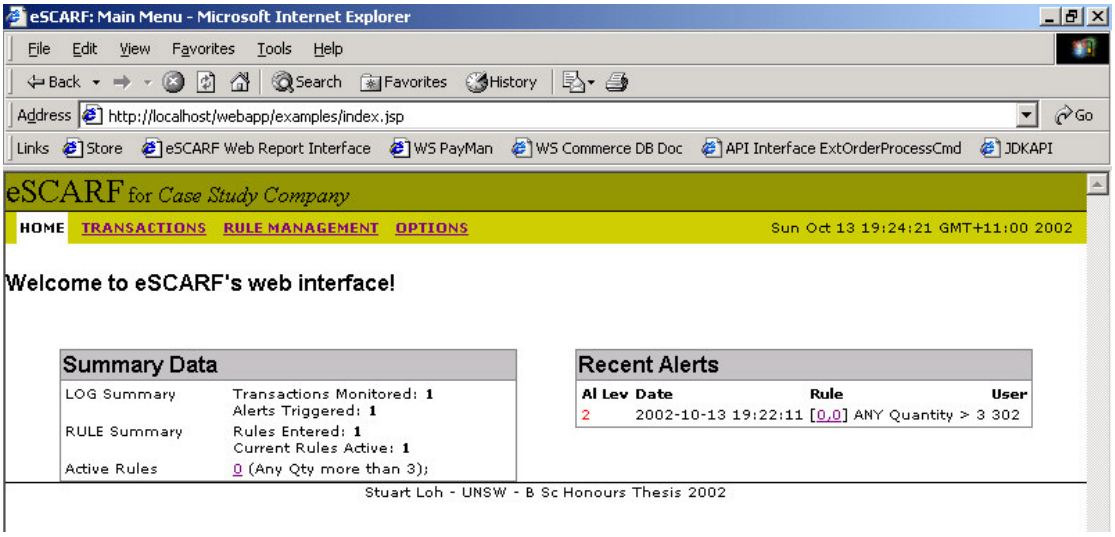
ANY Quantity > 3
Alert Level 2

3. The rule management module of eSCARF, showing a new rule being created.



ANY Quantity > 3

4. The expression builder - building the expression part of a rule.



eSCARF: Main Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
eSCARF for Case Study Company
HOME TRANSACTIONS RULE MANAGEMENT OPTIONS

http://localhost/webapp/examples/index.jsp

Sun Oct 13 19:24:21 GMT+11:00 2002

Welcome to eSCARF's web interface!

Summary Data

LOG Summary	Transactions Monitored: **1** Alerts Triggered: **1**
RULE Summary	Rules Entered: **1** Current Rules Active: **1**
Active Rules	0 (Any Qty more than 3);

Recent Alerts

Al Lev	Date	Rule	User
2	2002-10-13 19:22:11	[0,0] ANY Quantity > 3	302

Stuart Loh - UNSW - B Sc Honours Thesis 2002

5. eSCARF web reporting, where an auditor goes to view audit reports.

QUESTIONNAIRE FOR THE EVALUATION OF THE eSCARF FRAUD DETECTION CONTINUOUS ASSURANCE SYSTEM

Please ensure that you have read the information on the previous pages to familiarise yourself with the context of this questionnaire. If you require any clarification regarding the questions in this questionnaire, please direct them to the person administering this questionnaire. The questionnaire contains 41 multiple-choice questions (7 demographic and 34 directly concerning eSCARF) plus 6 free-format questions you should use to provide additional information (if you need more space for these, use the back of the page).

DEMOGRAPHICAL DETAILS

Name: ________________________________

What is the extent of your knowledge in…	None	Minimal	Adequate	Substantial	Extensive
… Information Systems?	❍	❍	❍	❍	❍
… Information Systems Auditing?	❍	❍	❍	❍	❍
… Continuous Assurance?	❍	❍	❍	❍	❍

What is the extent of your expertise in…	None	Basic	Intermediate	Advanced	Expert
… Information Systems?	❍	❍	❍	❍	❍
… Information Systems Auditing?	❍	❍	❍	❍	❍
… Continuous Assurance?	❍	❍	❍	❍	❍

How many years experience have you had in auditing (tick one)?

❍ Zero
❍ 1-2
❍ 3-5
❍ > 5

Please answer the following questions from an auditor's perspective, after you have had eSCARF demonstrated to you.

SECTION A – PERCEPTIONS

Please rate the importance of these factors in a Continuous Assurance System:	Very unimportant	Unimportant	Somewhat Unimportant	Neutral	Somewhat Important	Important	Very Important
1. Accuracy of Information (correctness)	❍	❍	❍	❍	❍	❍	❍
2. Comprehensiveness of Information (completeness)	❍	❍	❍	❍	❍	❍	❍
3. Conciseness of Information	❍	❍	❍	❍	❍	❍	❍
4. Timeliness of Information (how current)	❍	❍	❍	❍	❍	❍	❍
5. Presentation of Information	❍	❍	❍	❍	❍	❍	❍
6. User-friendly Interface (easy to navigate, intuitive to use)	❍	❍	❍	❍	❍	❍	❍
7. Ease of Customisation	❍	❍	❍	❍	❍	❍	❍

SECTION B – eSCARF COMPONENT EVALUATION

Please state how much you agree or disagree with the following statements:	Strongly Disagree	Disagree	Disagree Somewhat	Neutral	Agree Somewhat	Agree	Strongly Agree
Rule Management (Creation/Maintenance of Rules)							
8. The information on the rule management screen is comprehensive.	❍	❍	❍	❍	❍	❍	❍
9. The information on the rule management screen is concise.	❍	❍	❍	❍	❍	❍	❍
10. The information on the rule management screen is presented well.	❍	❍	❍	❍	❍	❍	❍
11. Rule management has a user-friendly interface.	❍	❍	❍	❍	❍	❍	❍
12. Customising and managing rules is easy.	❍	❍	❍	❍	❍	❍	❍
13. Rule management is flexible. (I can manage and customise rules how I want to.)	❍	❍	❍	❍	❍	❍	❍

Further comments/suggestions regarding eSCARF’s Rule Management:

Server Console Log

14. The information on the server console log is accurate.	❍	❍	❍	❍	❍	❍	❍
15. The information on the server console log is comprehensive.	❍	❍	❍	❍	❍	❍	❍
16. The information on the server console log is concise.	❍	❍	❍	❍	❍	❍	❍
17. The information on the server console log is current.	❍	❍	❍	❍	❍	❍	❍

Further comments/suggestions regarding eSCARF’s Server Console Log:

Rule Checking and Alerts

18. The rule checker functions as expected.	❍	❍	❍	❍	❍	❍	❍
19. The alerts generated are timely.	❍	❍	❍	❍	❍	❍	❍
20. The alerts generated are useful.	❍	❍	❍	❍	❍	❍	❍

Further comments/suggestions regarding eSCARF’s Rule Checking and Alerts:

Web Reporting

21. The information in web reporting is accurate.	❍	❍	❍	❍	❍	❍	❍
22. The information in web reporting is comprehensive.	❍	❍	❍	❍	❍	❍	❍
23. The information in web reporting is concise.	❍	❍	❍	❍	❍	❍	❍
24. The information in web reporting is current.	❍	❍	❍	❍	❍	❍	❍
25. The information in web reporting is presented well.	❍	❍	❍	❍	❍	❍	❍
26. Web reporting has a user-friendly interface.	❍	❍	❍	❍	❍	❍	❍

Further comments/suggestions regarding eSCARF’s Web Reporting:

SECTION C – eSCARF OVERALL EVALUATION

Please state how much you agree or disagree with the following statements:	Strongly Disagree	Disagree	Disagree Somewhat	Neutral	Agree Somewhat	Agree	Strongly Agree
27. eSCARF provides information that is accurate.	❍	❍	❍	❍	❍	❍	❍
28. eSCARF provides information that is comprehensive.	❍	❍	❍	❍	❍	❍	❍
29. eSCARF provides information that is concise.	❍	❍	❍	❍	❍	❍	❍
30. eSCARF provides information that is current.	❍	❍	❍	❍	❍	❍	❍
31. eSCARF presents information well.	❍	❍	❍	❍	❍	❍	❍
32. eSCARF has a user-friendly interface (easy to navigate, intuitive to use).	❍	❍	❍	❍	❍	❍	❍
33. eSCARF is easy to customise	❍	❍	❍	❍	❍	❍	❍

Overall Rating

34. eSCARF is useful for auditors of e-commerce systems.	❍	❍	❍	❍	❍	❍	❍

Please write any general comments about impressions you have about the eSCARF system:

Do you have any suggestions for other functionality in the eSCARF system?

Thank you for taking the time and effort to participate in this survey.

Appendix 7: Final Questionnaire

EVALUATION SURVEY FOR THE ESCARF FRAUD DETECTION CONTINUOUS ASSURANCE SYSTEM

Survey Overview and Introduction

This survey is designed to gather evaluation data of a software system called eSCARF (Electronic System Control Audit Review File) in regards to its usability and usefulness from an auditor's perspective. eSCARF is a continuous assurance system designed to detect fraud in e-commerce system transactions. Firstly, some background information regarding continuous assurance and how eSCARF operates follows, in order to help you understand the eSCARF system. Secondly, a walkthrough demonstration of the eSCARF system will be presented. After this walkthrough, fill out the attached questionnaire. Please answer the questionnaire questions from an auditor's perspective, using your expertise in auditing.

What is Continuous Assurance?

Continuous Assurance is a rapidly developing field of research which extends upon the traditional accounting practice of auditing. Continuous Assurance can be defined as a methodology that permits independent auditors to provide assurance on subject matter, using a series of assurance reports issued simultaneously (or a short period after) the occurrence of events underlying the subject matter. In other words, continuous assurance provides the ability for auditors to assure all types of information shortly after the occurrence of the events being assured.

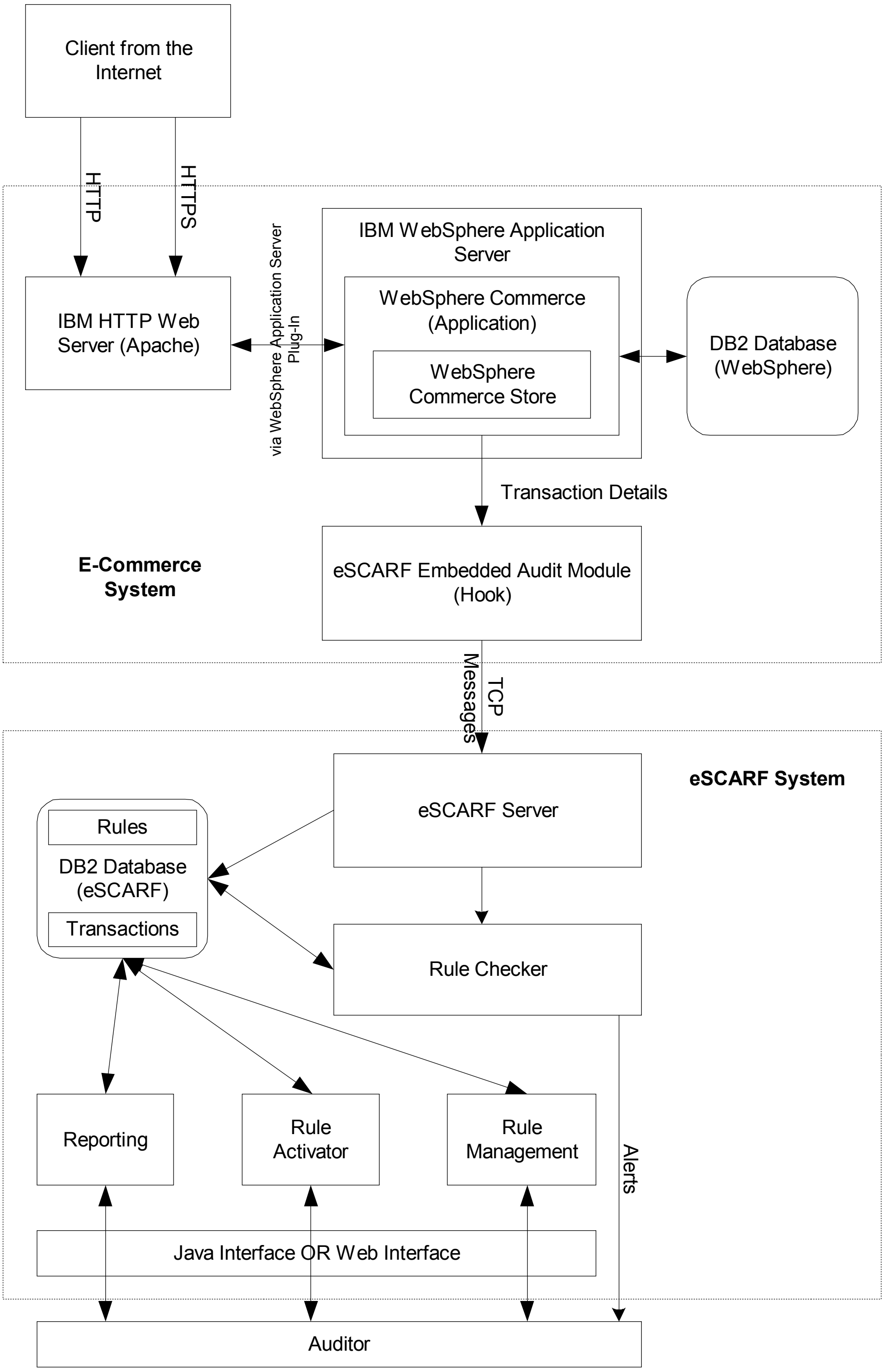
Auditing has traditionally been performed on financial and accounting data to attribute a measure of trust and assurance to it. However, auditing can be performed on all types of data (called "assurance" when auditing is used in a non-financial context). Normally, the auditing process is protracted, meaning that in the past it has not been practical to assure all forms of data, as non-financial data tends to be of a more time critical nature than financial data (that is, assurance for non-financial data must be provided in a more timely fashion than financial data). A level of automation must be introduced in order to make the auditing process quicker, and thus allow all types of data to be audited. This idea is called continuous assurance.

The advent and increasing use of information systems in society has allowed continuous assurance to be implemented. Data being assured is now highly accessible to auditors due to it being electronically stored and managed on information systems. Furthermore, auditing procedures are able to be automated via computers, providing the ability to assure this data quickly. Therefore, through the proliferation of information systems, it is now feasible to implement continuous assurance processes in organisations. The development of continuous assurance systems is particularly useful, as all types of data, especially data important to decision making in the organisation, can be assured.

eSCARF is an implementation of such a continuous assurance system. eSCARF assures e-commerce transactions, with the aim of detecting fraudulent transactions.

eSCARF: How it Works

eSCARF, as with all continuous assurance systems, work concurrently with the system they are assuring. eSCARF is designed to provide assurance for B2C e-commerce stores, with this version of eSCARF designed for IBM WebSphere Commerce 5.4, one such e-commerce system. eSCARF basically examines, in real-time, e-commerce transactions (an order placed by a customer) as they are made. Each transaction is compared to a set of rules that an auditor designs with the view to detect fraud. The rules allow eSCARF to look at attributes of the transaction currently being examined, as well as the attributes of previous transactions. The following diagram shows the architecture of eSCARF:

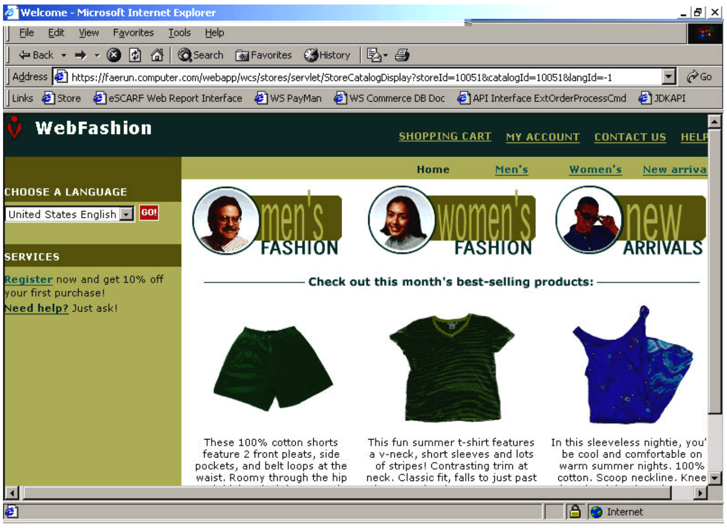


Because continuous assurance systems are designed to be as unintrusive as possible to the systems they assure, eSCARF is a distinct and separate system. It "hooks" into IBM WebSphere Commerce at one point only, where it captures transaction details and sends them to the eSCARF server. The server passes the transaction to a rule checker, which logs the transactions to the eSCARF database and produces any alerts if fraud is suspected.

Auditors manage the eSCARF system, with modules provided for each task they need to perform, all accessible via a main menu:

- Rule Management allows auditors to create and maintain audit rules which are aimed at detecting fraud.
- Reporting allows real-time reports to be generated. Auditors use these reports to analyse the data eSCARF collects. The reports are designed to provide information in summary form with the ability to drill-down for more detail.
- Alerts can be produced in response to a transaction which is highly suspicious. These alerts can consist of an onscreen alert, or even an SMS sent to an auditor's mobile phone.
- The eSCARF server can be started and stopped.

The following screenshots illustrate various parts of eSCARF in action:


Welcome - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI
WebFashion
SHOPPING CART MY ACCOUNT CONTACT US HELP
Home Men's Women's New arriva
CHOOSE A LANGUAGE
GO!
SERVICES
men's FASHION
women's FASHION
new ARRIVALS
Internet

https://faerun.computer.com/webapp/wcs/stores/servlet/StoreCatalogDisplay?storeId=10051&catalogId=10051&langId=-1

United States English

Register now and get 10% off your first purchase!
Need help? Just ask!

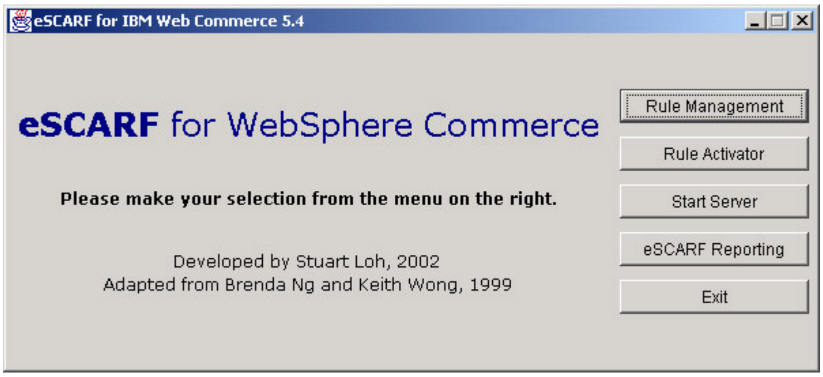
Check out this month's best-selling products:

These 100% cotton shorts feature 2 front pleats, side pockets, and belt loops at the waist. Roomy through the hip

This fun summer t-shirt features a v-neck, short sleeves and lots of stripes! Contrasting trim at neck. Classic fit, falls to just past

In this sleeveless nightie, you' be cool and comfortable on warm summer nights. 100% cotton. Scoop neckline. Knee

1. WebFashion is the e-commerce store running on IBM WebSphere Commerce that will be audited.

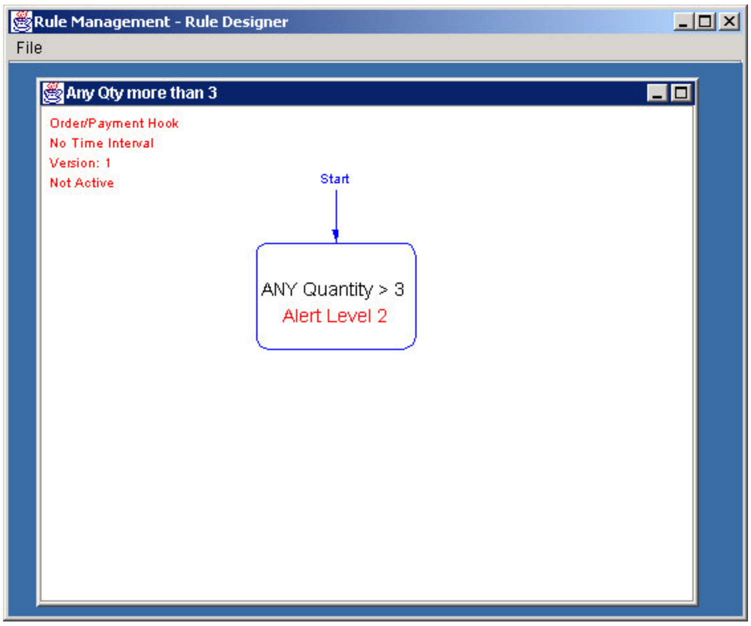


eSCARF for WebSphere Commerce

Please make your selection from the menu on the right.

Developed by Stuart Loh, 2002
Adapted from Brenda Ng and Keith Wong, 1999

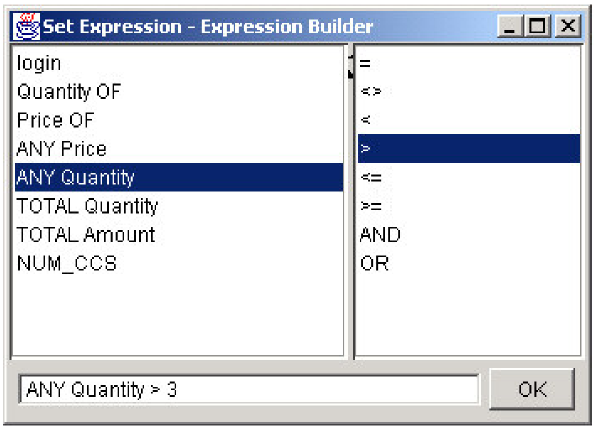
2. The main menu of eSCARF.



Order/Payment Hook
No Time Interval
Version: 1
Not Active

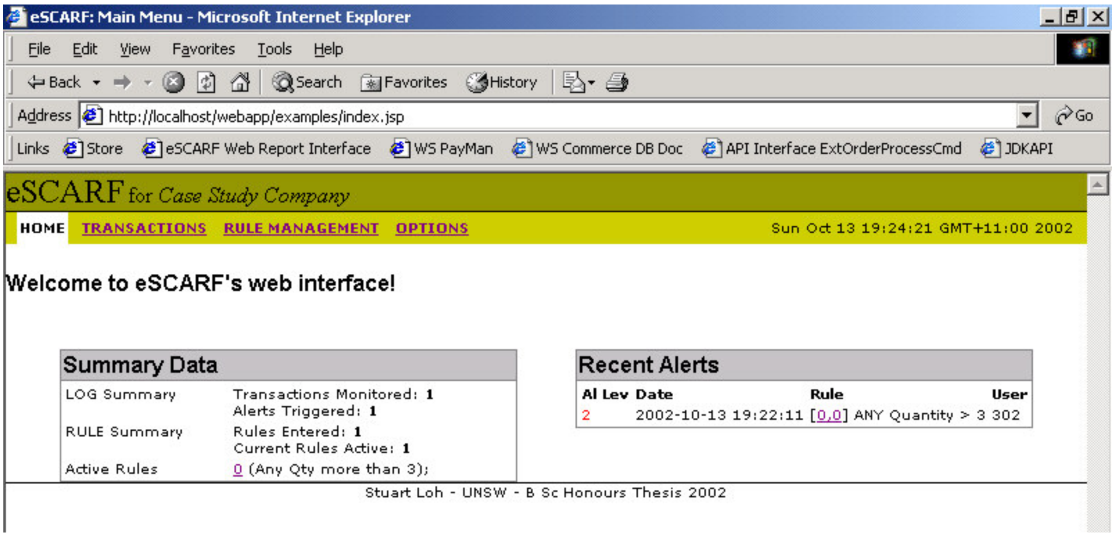
ANY Quantity > 3
Alert Level 2

3. The rule management module of eSCARF, showing a new rule being created.



ANY Quantity > 3

4. The expression builder - building the expression part of a rule.


eSCARF: Main Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links Store eSCARF Web Report Interface WS PayMan WS Commerce DB Doc API Interface ExtOrderProcessCmd JDKAPI

http://localhost/webapp/examples/index.jsp

eSCARF for *Case Study Company*

HOME TRANSACTIONS RULE MANAGEMENT OPTIONS Sun Oct 13 19:24:21 GMT+11:00 2002

Welcome to eSCARF's web interface!

Summary Data

LOG Summary	Transactions Monitored: 1 Alerts Triggered: 1
RULE Summary	Rules Entered: 1 Current Rules Active: 1
Active Rules	0 (Any Qty more than 3);

Recent Alerts

Al Lev	Date	Rule	User
2	2002-10-13 19:22:11	[0,0] ANY Quantity > 3	302

Stuart Loh - UNSW - B Sc Honours Thesis 2002

5. eSCARF web reporting, where an auditor goes to view audit reports.

QUESTIONNAIRE FOR THE EVALUATION OF THE eSCARF FRAUD DETECTION CONTINUOUS ASSURANCE SYSTEM

Please read the information on the previous pages to familiarise yourself with the context of this questionnaire. If you require any further clarification regarding the questions in this questionnaire, please direct them to the person administering this questionnaire. The questionnaire contains 39 multiple-choice questions (7 demographic and 32 directly concerning eSCARF) plus 6 free-format questions you should use to provide additional information (if you need more space for these, use the back of the page).

DEMOGRAPHICAL DETAILS

Name: ________________________________

How would you rate the extent of your knowledge in…	None	Minimal	Adequate	Substantial	Extensive
… Auditing?	○	○	○	○	○
… Information Systems?	○	○	○	○	○
… Information Systems Auditing?	○	○	○	○	○
… Continuous Assurance?	○	○	○	○	○

How would you rate your expertise in…	None	Basic	Intermediate	Advanced	Expert
… Auditing?	○	○	○	○	○
… Information Systems?	○	○	○	○	○
… Information Systems Auditing?	○	○	○	○	○
… Continuous Assurance?	○	○	○	○	○

How many years experience have you had in auditing (tick one)?

○ Zero
○ 1-2
○ 3-5
○ > 5

Please answer the following questions from an auditor’s perspective, after you have had eSCARF demonstrated to you.

SECTION A – PERCEPTIONS

Please rate the importance of these factors in a Continuous Assurance System:	Very unimportant	Unimportant	Somewhat Unimportant	Neutral	Somewhat Important	Important	Very Important
1. Accuracy of Information (correctness)	○	○	○	○	○	○	○
2. Comprehensiveness of Information	○	○	○	○	○	○	○
3. Conciseness of Information	○	○	○	○	○	○	○
4. Timeliness of Information (how current)	○	○	○	○	○	○	○
5. Presentation of Information	○	○	○	○	○	○	○
6. User-friendly Interface (easy to navigate, intuitive to use)	○	○	○	○	○	○	○
7. Ease of Customisation	○	○	○	○	○	○	○

SECTION B – eSCARF COMPONENT EVALUATION

Please state how much you agree or disagree with the following statements:	Strongly Disagree	Disagree	Disagree Somewhat	Neutral	Agree Somewhat	Agree	Strongly Agree
Rule Management (Creation/Maintenance of Rules)							
8. The information on the rule management screen is concise.	○	○	○	○	○	○	○
9. The information on the rule management screen is presented well.	○	○	○	○	○	○	○
10. Rule management has a user-friendly interface (intuitive to use).	○	○	○	○	○	○	○
11. Customising and managing rules is easy.	○	○	○	○	○	○	○
12. Rule management is flexible. (I can manage and customise rules how I want to.)	○	○	○	○	○	○	○

Further comments/suggestions regarding eSCARF’s Rule Management:

Server Console Log

13. The information on the server console log appears to be accurate.	○	○	○	○	○	○	○
14. The information on the server console log is comprehensive.	○	○	○	○	○	○	○
15. The information on the server console log is concise.	○	○	○	○	○	○	○
16. The information on the server console log is current.	○	○	○	○	○	○	○

Further comments/suggestions regarding eSCARF’s Server Console Log:

Rule Checking and Alerts

17. The rule checker functions appear to work as expected.	○	○	○	○	○	○	○
18. The alerts generated are timely.	○	○	○	○	○	○	○

Further comments/suggestions regarding eSCARF’s Rule Checking and Alerts:

Web Reporting

19. The information in web reporting appears to be accurate.	○	○	○	○	○	○	○
20. The information in web reporting is comprehensive.	○	○	○	○	○	○	○
21. The information in web reporting is concise.	○	○	○	○	○	○	○
22. The information in web reporting is current.	○	○	○	○	○	○	○
23. The information in web reporting is presented well (well formatted).	○	○	○	○	○	○	○
24. Web reporting has a user-friendly interface.	○	○	○	○	○	○	○

Further comments/suggestions regarding eSCARF’s Web Reporting:

Section C – eSCARF Overall Evaluation

Please state how much you agree or disagree with the following statements:	Strongly Disagree	Disagree	Disagree Somewhat	Neutral	Agree Somewhat	Agree	Strongly Agree
25. eSCARF provides information that appears to be accurate.	○	○	○	○	○	○	○
26. eSCARF provides information that is comprehensive.	○	○	○	○	○	○	○
27. eSCARF provides information that is concise.	○	○	○	○	○	○	○
28. eSCARF provides information that is current.	○	○	○	○	○	○	○
29. eSCARF presents information well.	○	○	○	○	○	○	○
30. eSCARF has a user-friendly interface (easy to navigate, intuitive to use).	○	○	○	○	○	○	○
31. eSCARF is easy to customise	○	○	○	○	○	○	○
Overall Rating							
32. eSCARF is useful for auditors of e-commerce systems.	○	○	○	○	○	○	○

Please write any general comments about impressions you have about the eSCARF system:

Do you have any further suggestions for other functionality in the eSCARF system?

Thank you for taking the time and effort to participate in this survey.

Appendix 8: eSCARF Demonstration Procedure

WALKTHROUGH DEMONSTRATION PROCEDURE

Estimated demonstration completion time: 30-60 minutes.

Clean Up Database for New Demonstration

- Reset the eSCARF database with `d:\scarf\setup\refreshdb.bat`

Walkthrough

1. Start eSCARF.
2. Enter rule management, create these rules:
    a. **`ANY Quantity > 3`**
       Alert level = 2
       No time limit
    b. **`NUM_CCS >= 2`**
       Alert level = 2
       10 minute time limit
3. Activate rules A and B.
4. Open up WebFashion in a web browser.
5. Logon as `joe@blog.com / bloggs1` (a pregenerated user account).
6. Make these orders:
    a. 4 of any item (observe alert and server console).
    b. 1 of any item, with a second credit card number (observe alert).
7. Start up web reporting and browse through the web reports.

8. Show rule versioning by creating a new version of rule A (revise to **`ANY Quantity > 5`**). View web reports to show old rule under "deprecated rules".

If a demonstration of a multi-node rule is required:

9. Enter rule management, create this two node rule:
    a. **`login = joe@blog.com`**
       Alert level = 1
       No time limit
       If true, goto b:
    b. **`Order Total > $100`**
       Alert level = 2
       No time limit
10. Activate new rule.
11. Make this order:
    a. 5 of any item (observe 2 alerts).
12. Re-enter web reporting and browse through the web reports.

Demonstration concludes. The respondents are free to ask further questions, or explore the system for themselves.

Appendix 9: IBM WebSphere Commerce and eSCARF System Requirements

IBM WebSphere Commerce 5.4 requires that the following *minimum* hardware requirements must be met in order to run it successfully:

A Pentium III 733 MHz IBM-compatible personal computer with:

- A minimum 512 MB of RAM (1 GB of RAM is recommended for smoother operation of WebSphere).
- A minimum of 1.6 GB of free hard disk space on the partition WebSphere is being installed, with an additional 300 MB needed on the C: drive.
- Double the amount of paging space as there is RAM (eg: 512 MB RAM should have 1024 MB paging space).
- A CD-ROM drive.
- A mouse or other pointing device.
- A graphics adapter capable of at least 256 colours.
- A LAN adapter that supports TCP/IP, or a Microsoft Loopback adapter.

WebSphere must be installed on Windows NT 4 (with Service Pack 6a) or Windows 2000 Server or Advanced Server (with Service Pack 2). WebSphere does not currently support any version of Windows XP.

eSCARF requires that the following *minimum* hardware requirements must be met in order to run it successfully:

- 2 MB of hard disk space, plus space required for eSCARF to store its transaction data.

eSCARF currently uses the following applications, which must also be available on the computer on which it is installed:

- IBM DB2 Database System
- A web server capable of interpreting JSP pages
- Java Runtime Environment (minimum version 1.3.1)

Appendix 10: CD Contents Details

This appendix describes the contents of the three compact discs that are attached to this thesis. eSCARF is found on the disc 1. WebSphere is found on discs 1, 2 and 3.

CD 1: WebSphere Commerce Installation Disc and eSCARF Installation Disc

Directory	File(s)	Description
/	readme_scarf.html	Installation instructions for
	readme_ws.html	Installation instructions for WebSphere Commerce.
	wc54bed1.zip	WebSphere installation zip file
/eSCARF/	*.*	eSCARF files. See appendix 5 for details.
/docs/	WCQuickBeginnings.pdf	Installing WebSphere Commerce 5.4.
	ProgrammersGuide.pdf	WebSphere Commerce 5.4 Programmer's Guide.
	wcs54archint.pdf	WebSphere Commerce 5.4: Architecture and Integration Guide.
	whatsnew54.pdf	What's New in WebSphere Commerce 5.4.

`readme_ws.html` gives information regarding the first procedure of installing WebSphere Commerce (this will involve unzipping `wc54bed1.zip`, and the zip files on the other two CDs on to the hard drive). You will then need to refer to **`/docs/WCQuickBeginnings.pdf`** to complete the installation procedure.

`readme_scarf.html` gives information regarding the installation of eSCARF.

The remaining two CDs in the set contain data files required for the installation of WebSphere Commerce.

CD 2: WebSphere Commerce Installation Disc 2 (DB2)

Directory	File(s)	Description
/	wc54dbdb.zip	Zip file containing the DB2 component of the WebSphere Commerce installation suite.

CD 3: WebSphere Commerce Installation Disc 3 (WebSphere Application Server and Fixpaks)

Directory	File(s)	Description
/	wc54wasa.zip	Zip file containing the WebSphere Application Server component of the WebSphere Commerce 5.4 installation suite.
	wc54bed2.zip	Zip file containing the fixpaks and patches for the WebSphere Commerce 5.4 installation suite.