available at www.sciencedirect.com

SCIENCE DIRECT®

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Continuous auditing technologies and models: A discussion

*S. Flowerday, A.W. Blundell, R. Von Solms**

Centre for Information Security Studies, Nelson Mandela Metropolitan University, Port Elizabeth, South Africa

ARTICLE INFO

Article history:
Received 13 June 2006
Revised 13 June 2006
Accepted 13 June 2006

Keywords:
Continuous auditing
Real-time assurances
Information integrity
Internal controls
Technology-based prevention

ABSTRACT

In the age of real-time accounting and real-time communication current audit practices, while effective, often provide audit results long after fraud and/or errors have occurred. Real-time assurances can assist in preventing intentional or unintentional errors. This can best be achieved through continuous auditing which relies heavily on technology. These technologies are embedded within and are crucial to continuous auditing models.

1. Introduction

In today's fast paced business world, real-time information systems facilitate real-time accounting systems and real-time communication between entities. Current audit practices, while proving adequate, take too long to provide assurances. Current audit practices also uncover intentional and unintentional errors, however, only after it has possibly had a detrimental effect on the organisation. A method of prevention by detecting these errors early is desirable. It is time to provide real-time assurances to decision makers.

Real-time assurances can only be provided by continuous auditing technologies. This paper discusses a range of audit technologies within three continuous auditing models. Each of these models has a different focus and makes use of different technologies. The available technologies, tools and techniques used in continuous auditing, will be interrogated.

The aim of this paper is to reach a conclusion about how a generic, yet comprehensive continuous auditing system could make use of the available tools, techniques and technologies for testing internal control and performing tests on transactions. To achieve this aim, after discussing the history and definition of continuous auditing the reasons for using continuous auditing systems will be explored. Once this is clarified, the tools and techniques necessary to implement continuous auditing are discussed, these are then placed into context by discussing three prominent continuous auditing models. The three models are then compared in tabulated form, after which possible future technologies are suggested for addressing internal control issues and testing transactions within a continuous auditing system.

2. Definition and history of continuous auditing

There are several differing ideas of what continuous auditing (CA) systems are, and how they work. Each of the models

* *Corresponding author.*
E-mail addresses: sflowerday@telkomsa.net (S. Flowerday), ablundell@nmmu.ac.za (A.W. Blundell), rossouw@nmmu.ac.za (R. Von Solms).
0167-4048/$ – see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2006.06.004

(discussed later) has their own definition, differing slightly. The most widely accepted definition though, is one released in 1999 and reads as follows: ''*a methodology that enables independent auditors to provide written assurance on a subject matter using a series of auditors' reports issued simultaneously with, or a short time after, the occurrence of events underlying the subject matter*'' (CICA/AICPA, 1999). This is the definition used for the purpose of describing continuous auditing throughout this paper.

In the early 1990s, the business environment went through a series of substantial changes. The ''*Electronization*'' of business and the proliferation of e-business lead to paperless accounting systems (Bierstaker et al., 2001; Vasarhelyi, 2002). This move towards technologies such as Electronic Data Interchange (EDI) and Electronic File Transfer (EFT) caused the evaporation (or disappearance) of the traditional audit trail. Auditors could no longer look for source documents in paper form and increasingly had to perform tests and gather evidence electronically, thus their audit techniques had to undergo some changes (Bierstaker et al., 2001; Helms and Mancino, 1998).

The trend which caused the disappearance of the traditional audit trail continued and online systems and the Internet created an easier, cheaper way for data to be exchanged between systems. The *html* documents proved to be inadequate as it was difficult to extract and compare data because *html* only describes how data should be presented (Alles et al., 2004). Therefore, a language which could be intelligently manipulated and which would form a standard for data transfer was required. XML (eXtensible Markup Language) adds information about the document's content to its tags, thus it is easy to search, especially if digital agents are used. A subset of XML, XBRL (eXtensible Business Reporting Language) was created to describe business reporting information.

XBRL is useful for preparing, publishing, exchanging, acquiring and analysing accounting and business data, and provides a standardised method for transferring financial reporting information between different software applications (Alles et al., 2004; Srinivas, 2004). XBRL assigns tags to financial information, which allows computers to ''understand'' the data, while it can still produce human-readable reports. These tags are standardised by rules known as taxonomies. Taxonomies can contain rules (and tags) specific to certain industries or businesses as well as the Generally Accepted Accounting Principles (GAAP) rules. These may be region or country specific though (Pinkster, 2003).

XBRL enables continuous auditing by placing financial data in a format which is not proprietary to any specific software application, allowing any future continuous auditing system access to data on any software platform, running any software (which uses XBRL), in any country.

The concept of continuous auditing has been established in this section, this has explained *what* continuous auditing is. In the next section, *why* continuous auditing is necessary will be explained.

3. Motivation for CA technologies

In real-time accounting systems it has become desirable to have continuous assurances as to the condition of the information's integrity (Flowerday and von Solms, 2005). Furthermore, continuous assurances allow for corrective action to be taken sooner when a problem is found rather than in current auditing scenarios. To quote, ''*The focus of the audit will shift from manual detection to technology-based prevention*'' (Bierstaker et al., 2001).

Auditors aim to provide management with an opinion on subject matter for which management is responsible. In order to do this he/she will have to *validate* the *accuracy* of financial records and the *reliability* of the systems which store, transport and process those transactions. Looking at accuracy entails checking for fraud and error in transactions. There are well established auditing technologies which can assist in looking for *material* misstatements in financial records. Using these technologies within a continuous auditing system can extend their current effectiveness, as all transactions are analysed in real-time.

When assessing the reliability of the reports produced by the system, the auditor will look at confidentiality, integrity and availability and how this is ensured by the system of internal controls. Technology can be used to assess the internal control systems and see whether they are in line with prescribed norms. In the next section the nature of these technologies will be explored, explaining *how* the aims of continuous auditing can be met.

4. Technology aided tools and techniques

In order to verify that a real-time accounting system is producing reliable and accurate financial information, testing of controls must be done simultaneously with substantive tests of transactions (Helms and Mancino, 1999; Rezaee et al., 2001).

There are various tools and techniques which can aid in the analysis of transactions and internal controls. The tools are required to perform a variety of tasks. They can either be purchased software packages or auditor-designed routines (Rezaee et al., 2001). Collectively these tools and techniques are often referred to as Computer Aided Tools and Techniques or CATTs. An alternative acronym is CAATTs or Computer Aided Audit Tools and Techniques. CATTs have been used by auditors for many years and incorporate a wide variety of technologies, some of which are applicable to continuous auditing. In certain literature these have become known as Continuous Auditing Tools and Techniques. In this paper, ''CAATTs'' will refer to all of these collectively.

4.1. *Tools and techniques for analysing transactions*

To meet the requirements of an audit it is necessary to verify the accuracy of transactions to reveal fraud or error. Substantive tests of transactions must be performed. These will aim to obtain evidence showing possible material misstatements in the financial statements (South African Institute of Chartered Accountants, 2003). Two types of substantive tests are performed.

4.1.1. *Analytical procedures*

Analytical procedures involve performing comparisons of financial data to establish a relationship, often involving the

calculation of ratios. Analytical procedures not only indicate the possible existence of financial misstatements, they can also reveal how the client's industry and business function. When performed in the final phase of an audit, analytical procedures allow the auditor to comment on the reasonableness of transactions and the ability of the client to continue as a going concern.

CAATTs make analytical procedures more feasible and affordable than before (Rezaee et al., 2001). Many types of analytical procedures are too complex or time-consuming to be done manually. Using CAATTs also means that it has become possible to use larger sets of data when performing analytical procedures.

4.1.2. *Tests of transactions and balances*

The testing of transactions is often performed at the same time as testing controls. Transactions are tested continuously, throughout the financial year. This is done to discover whether material misstatements have occurred. In other words, to see whether erroneous or irregular processing of the transactions has taken place.

Testing transactions continuously throughout the year can help to reduce the number and/or complexity of tests of balances which need to be performed after balance sheet date (Rezaee et al., 2001). Tests done on balances are usually to collect evidence, on which the auditor can ground his or her opinion on fair representation of financial statements (Rezaee et al., 2002). When performing substantive tests of balances, Generalized Audit Software (GAS) tools are often used (Rezaee et al., 2001).

4.2. *Tools used in testing of internal controls and assessing risk*

In order to plan an audit, the auditor needs to be aware of the areas which carry the greatest risk and thus need the most scrutiny. This requires the auditor to look at the adequacy and effectiveness of internal controls within the system. According to the statement on auditing standards No. 80 (AICPA, 1996) CAATTs can be used for this purpose.

Testing of controls should also be ongoing. This allows the auditor to express an opinion as to how reliable the internal control system is. Knowing the reliability of the internal control system is important during the planning phase of an audit. The nature, timing and extent of substantive tests will be decided on accordingly (Rezaee et al., 2001).

In this section, the tools and techniques used for both analysing transactions and testing internal controls were elaborated, in the next section three of the models which use these tools and techniques will be explained.

5. Continuous auditing models

There are several suggested continuous auditing models, most are merely conceptual. Few seem to have been implemented in real-time systems. One of the early models was the Continuous Process Auditing System (CPAS) which was developed at AT&T Bell Laboratories. This is a methodology for internal auditing of large ''*paperless*'' real-time systems (Varsarhelyi and Halper, 1991). This model appears to have formed a basis for the later models.

In this paper, three of the better known models have been chosen for discussion. These all take somewhat different approaches and make use of different technologies.

5.1. *Continuous auditing building automated auditing capability (Rezaee et al., 2002)*

This is a conceptual framework for a continuous auditing system. It would be capable of running on a distributed client/server network and is also web-enabled for transmitting data to audit workstations. The model involves several steps.

Firstly, data are collected from transactional systems. This is done by linking to tables, via File Transfer Protocol (FTP), storage drives or via modem. The data are then stored on an audit server.

Once on the audit server, data are extracted from a variety of platforms and systems. Data standardisation is therefore required. Standards and formats are developed for storing data in the data warehouse/mart. The data are then transformed by cleaning, validating, restructuring the data and ''scrubbing'' with business rules.

An enterprise-wide data warehouse is not always needed, as it may be too expensive and complex. Instead, the required data could automatically be fed into several data marts. Data marts contain metadata which details the source transactions and the ETL (Extract Transform Load) process as well as the tests which take place. The metadata may for example include: detailed file definitions, business rules and transaction process flows.

Lastly, standardised tests are created to run within the data mart. The tests are created either to run continuously or at predetermined intervals. The tests are designed to automatically gather evidence and issue exception reports.

5.2. *Towards a paradigm for continuous auditing (Onions, 2003)*

To monitor the integrity of the data, Onions suggests *keystroke level* data examination. This basically involves monitoring database utilities and applications for commands which could cause fraud or error. This model addresses the testing of transactions in two ways.

Firstly, each transaction is audited and reported on as an isolated entity. This is done ''*ephemerally*'' – the transactions are tested at the time of entry. This is referred to as *transaction level* data examination. It ascertains whether each transaction fits the pre-specified rules for that transaction. These may be business rules or even rules dictating what actions are permissible for certain users. This is done in conjunction with performing certain analytical functions. Computer Assisted Audit Tools could be used. However, these operations would need to be performed on the transactions in real-time rather than batch. After the transaction has been examined it may be added to a data mine for possible further examination.

Secondly, the transactions are examined as a whole over a longer time period (perhaps even years). This examination looks for patterns in the transactions which could together result in fraud. This is known as the *transaction pattern level*

of data examination. Expert systems and rules based criteria would be employed. Rules would be similar to virus definitions and would be available for different industry types.

The problem when attempting to use expert systems is that each software package available has a different data schema. It would be very costly and time-consuming to create expert systems for each application. The solution would be to create a generic master file and transaction layout which could be used regardless of application data schema. This newly defined generic schema for a transaction would allow one expert system to trawl through the data mine. This schema would be defined using eXtensible Continuous Auditing Language (XCAL) which, similar to XBRL, is XML-based.

The model consists of four levels:

1. Transactions and data from various sources are entered for processing.
2. Transactions and keystrokes are mapped to XCAL schemas. This is done in real-time and is captured forensically on a daily basis.
3. Real-time CAATT processing is used to check transactions and keystrokes. Alerts may be sent to an Online Systems Audit Centre (OLSAC). Transactions are stored at this level for a day (but passed to level 4 where they are stored for years).
4. Expert systems look for patterns in the data.

5.3. *Continuous audit: model development and implementation within a debt covenant domain (Woodroof and Searcy, 2001)*

Woodroof and Searcy's model presents a conceptual model of continuous auditing. This model is limited in scope, as it is discussed in relation to debt covenant compliance. The model makes use of web-enabled technologies. It draws attention to the need for a reliable and secure system. The need for the production of *evergreen reports* is also discussed. Evergreen reports are reports which are generated on demand, usually viewed through a web site. The model is based on a database of transactions (journals and ledgers) on the client's system, with a web interface (on the auditors system) for the auditor to use.

This model is implemented in five stages:

1. A request is made for a report.
2. Agents and sensors within the client's system monitor the transaction data for exceptions to pre-specified rules. These exceptions are compared to the auditor-defined rules. This may trigger alarms, and alerts are sent to the auditor. The rules check the reliability of the system (possibly using continuous SYSTRUST), the fairness of the representation of financial reports and compliance to 3rd party contracts (like debt covenant agreements).
3. A digital agent on the auditor's system requests a digital agent on the client's system to retrieve the client's real-time balances of accounts via stored procedures in the client database.
4. If more information is returned than is needed, the digital agent extracts the information relevant to the ''contract'' (in this case debt covenant compliance). The information is checked for compliance, the actual event or process is checked against an acceptable standard for that event or process. If anomalies occur, these are flagged and the auditor is notified so that he/she may take action.
5. An evergreen report is generated and displayed to the loan officer. This details three levels of assurance. Level 1 is an assurance of reliability. If there is Level 1 exception, no further analysis is performed. Level 2 offers an opinion on the fairness of real-time financial statements. Level 3 provides an analysis of technical violations of 3rd party contracts (in this case debt covenant compliance is assessed).

Due to the reports being produced on demand (pull) (as opposed to being pushed to the user) this model is less suited to using XBRL-based reporting.

5.4. *Problems limiting use of CA systems*

One of the problems affecting continuous auditing solutions in real-time accounting systems is the varied data formats used. The ability to access and retrieve data from a variety of record sources, including legacy systems, is crucial to the creation of a continuous auditing system. This means that data will be in a variety of formats, with different file types and record systems. It becomes necessary to standardise these data. Unfortunately, this can be a complex and expensive process. Even more problematic is the risk of introducing errors such as duplicate records.

Technologies such as XBRL go a long way in creating a standard reporting format (Srinivas, 2004). Add to this, intelligent technology such as FRAANK (Financial Reporting and Auditing Agent with Net Knowledge) which can convert older reports into XBRL. This can create a way to compare non-XBRL data produced by legacy systems with newer XBRL reports (Kogan et al., 1998).

Until XBRL becomes widely implemented, using data marts to collect and assimilate data is an option. Onions (2003) also suggests adding XCAL, which would create a generic master file layout.

6. Evaluation of models

To evaluate these models one needs to consider at how accuracy and reliability are validated. In this context, *Accuracy* refers to how fraud and error in transactions are detected and how possible material misstatements in financial records are detected. *Reliability* is how confidentiality, integrity and availability of internal controls are examined. Further, these models will also be compared on *Real-time Processing* the *Reporting Method* used and the *Proposed Data Format* (Table 1).

It is apparent that the approaches of the three models differ slightly from each other, however, they all aim to function as close to real-time as possible. Some of the models use different technologies to achieve the same goal. For instance, detecting fraud and error may be accomplished by CAATS, digital agents or expert systems.

In the following section, suggestions will be made on how to draw together the tools and technologies used within these models to create comprehensive future CA systems.

Table 1 – Comparison of three continuous auditing models

	Rezaee et al.	Onions	Woodroof and Searcy
Accuracy (fraud and error) within transactions	Standardised audit tests are built into audit data marts. They run either continuously or at predetermined times. These gather evidence and then generate the relevant reports.	Transactions are checked both at time of entry and later. CAATTS (real-time, not batch). Expert systems (not in ''real-time'' but running continually).	Rule-based detection by digital agents. Data are analysed by devices integrated into the system.
Reliability of internal control system	CAATS are used. These include Integrated Test Facilities (ITFs) and parallel simulation. ITFs are used to verify correctness and completeness of processing. Parallel simulation tests assess effectiveness of control activities.	Parsing of keystrokes to detect database management utilities. Password control. Operating system's security. Audit logs. Web services verify information (e.g. new supplier's credit history checked).	Adapt and apply SYSTRUST principles. Web-based valuation sites. Must be in the auditor-defined rules for the digital agents.
Real-time	Real-time processing is the aim for this system.	All proposed systems run in parallel with operational systems in real-time.	Real-time reporting is one of the aims of this model. To this end, information must be collected and monitored in real-time.
Reporting method	Web-enabled data delivery of data to auditors' workstations, where reports can be generated (possibly by Generalized Audit Software).	Graded alerts sent through Virtual private networks (VPNs) to audit department/OLSAC. The alerts are graded by gravity (three levels).	Three levels of reporting, alerts are sent to the auditor via email. Level 1: reliability of the system or security of the transmission. Level 2: transactions and processes. Level 3: technical violation of 3rd party agreement. 3rd party and auditor notified by email. Evergreen reports are produced on demand through a web interface – information *pull* approach (as opposed to XBRL reporting this is push reporting method).
Proposed data format	Data mart Data warehouse XBRL	XCAL Data marts	Does not interface with legacy systems.

7. The future of CA technology

To meet the requirements of continuous auditing, any comprehensive CA model would need to address both *internal control testing* and *testing of transactions*. A dual-pronged approach, where both the system (internal controls) and data (transactions) are tested, simultaneously would be the ideal.

7.1. *Internal control issues*

There is a need to collect evidence on the quality and integrity of an electronic system in producing reliable and accurate financial information. Technology must aid in verifying the integrity of data, because the conclusions in auditors' reports must be based on accurate and reliable data in order to be deemed trustworthy (Wessmiller, 2002).

There is also a need to ensure the security of the system. A system which is not secure is not reliable. Ensuring security would involve examining internal controls. If the system is not reliable, the results from that system may not be viewed as trustworthy. Woodroof and Searcy (2001) suggest SYSTRUST (or a CA derivative of SYSTRUST). COBIT Guidelines in conjunction with ISO 17799 could also be used. COBIT could address internal control related issues (The IT Governance Institute, 2005). ISO 17799 would aid in addressing information security issues (ISO/IEC 17799, 2005). Thus, a number of best practices and/or standards exist to address the security issues.

Rezaee et al. (2002) mention Concurrent Audit Techniques for testing effectiveness of a client's internal controls. Concurrent Audit Techniques include SCARF (Systems Control and Review Facility) and the snapshot approach, where SCARF functions as an exception reporting system. It captures transactions meeting certain criteria (defined by the auditor) by using Embedded Audit Modules. The captured transactions are set aside for later review by an auditor. Embedded Audit Modules and the SCARF approach could be used to create alerts regarding the status of internal control systems by checking if controls are implemented.

7.2. *Transactions*

Processing of transactions should occur in several stages. Various technologies help throughout these stages. Firstly, transactions from a variety of sources are extracted. Not all records or fields may be required, digital agents and stored database procedures could be used to pull out only the necessary data. The creation of data marts and data warehouses may be desirable, a capable Database Management System (DBMS) would be required.

Once data have been collected and transformed, the transactions need to be assessed. Transactions are individually assessed for integrity (errors and fraud) and validity (business rules). CAATTS can be made to run as close to real-time as possible, while the data are flowing through the application system. Both analytical procedures and substantive testing should be applied to look for fraud and error. Common IT-based fraud schemes often involve the billing system, payroll system and check tampering. These schemes often need to be identified at the transaction data level, as they can depend on groups of transactions within the system. Examples include ghost vendors, ghost employees and exploiting voids and returns (Taylor, 2005).

Often a Database Management System (DBMS) forms an important part of a system. An example of a commonly used DBMS is Oracle. Oracle is a DBMS which allows ''triggers'' to perform tasks when certain criteria are met. Triggers are useful for creating logs for system events (Finnigan, 2003). Triggers in a database can be used in the same way as Data Query Modules (DQMs). DQMs are macros or programs built using audit software, they perform queries to answer a specific question posed by the auditor. For example, DQMs could be used to look for fraudulent travel allowance claims of employees. Employee swipe card data could be compared to dates on tour and travel reports. If the employee's card was swiped, and they were at the office, they are most likely claiming travel allowance for extra days (Dalal, 2000). If the entry of a new travel claim triggered the execution of the relevant DQM, an instant audit would occur.

At this stage, alerts could be produced. For example control agents may be used to alert auditors if transaction values change too much from the norm. The relevant transaction data need to be collected for future forensic analysis – possibly by moving the data to a secured partition or dedicated audit server. This may be achieved using FTP and tape or other large-capacity storage devices. Digital agents then examine transactions and select those that should be set aside for later analysis. CIS (Continuous and Intermittent Simulation) could also be used for deciding which transactions require more examination.

The data may then need to be standardised. This may be done either on the audit server or in the source application, depending on the cost of processing. For example, the cost of processing on mainframes is more expensive. The data can then be aggregated in data marts or data mines. Data mines are expensive and normally only larger organisations can afford them. Data marts are often used by smaller organisations, or in larger organisations for one specific focus area, for example, Human Resources, Accounting data, etc. (Rezaee et al., 2001). Smaller organisations could also make use of XCAL, as suggested by Onions (2003).

Stored data may then be checked for groups of transactions and the cumulative effects of a series of transactions. Expert systems and digital agents may be very useful for this purpose. Analytical procedures could also be used to create ''norms'' which can be used as a benchmark. The results of these tests are then used to create reports and alerts, which are sent to auditors by encrypted email. The alerts may also be sent via VPNs. A grading system for alerts, showing the possible impact of the anomaly, may be important to the auditors.

It may also be necessary to be in contact with an outside auditing bureau for verification and validation. For example the Online Systems Audit Centre (Onions, 2003). The Online Systems Audit Centre is a group of auditing professionals, which monitor and investigate alerts online. VPNs can be used for this purpose. Web-enabled agents like FRAANK, and Web Services may also provide useful, secure ways to communicate (Kogan et al., 1998; Murthy and Groomer, 2004).

8. Conclusion

Within real-time accounting systems, real-time assurances are not only desirable, but are possible. Technologies which enable the provision of real-time assurances are becoming commonplace. This includes technologies which test internal controls and those related to testing transactions. Many of these technologies are not new, for example CAATTS, including GAS, EAMs and ITFs, which are being applied in new ways to achieve continuous auditing. Some innovative technologies, such as AI technologies allow every transaction to be inspected, instead of just a sample. These available technologies need to be brought together in a way which makes full use of each of them.

Models have been suggested for this purpose, however, most are only conceptual. These models can be adapted and

adjusted in order to provide the auditor with the reliable and accurate results he or she desires. A possible reason for the lack of comprehensive continuous audit models may be the result of the problems related to the variety of data formats which exist and the availability of audit data. Technologies such as XBRL contribute to solving the problem, but further enhancements are definitely envisaged.

A comparison of the three most prominent continuous auditing models is tabulated. The models are compared according to a set of criteria. These include: how accuracy and reliability are evaluated, what the reporting method is, and how close to real-time the model functions, and the proposed data format. These findings highlight the core aspects of the three models and can be used as a foundation on which to build future continuous auditing solutions.

REFERENCES

Alles M, Kogan A, Vasarhelyi M. Real time reporting and assurance: has its time come? Available from: http://raw.rutgers.edu/continuousauditing/Real_Time_Reporting_-_ICFAI1.doc; 2004 [retrieved 12.05.2005]

American Institute of Certified Public Accountants (AICPA). Amendment to statement on auditing standards no. 31, evidential matter: SAS 80; 1996.

Bierstaker JL, Burnaby P, Thibodeau J. The impact of information technology on the audit process: an assessment of the state of the art and implications for the future. Managerial Auditing Journal 2001;16(3):159–64.

CICA/AICPA. Continuous auditing: research report. Canadian Institute of Chartered Accountants; 1999.

Dalal C. Advanced use of audit software in audit and fraud detection – audit software: an indispensible tool in the new millennium. Ineternal Auditors 2000;3(February 1).

Finnigan P. Introduction to simple oracle auditing. Available from: http://www.securityfocus.com/print/infocus/1689; 2003 [retrieved 10.04.2006].

Flowerday S, von Solms R. Real-time information integrity = system integrity + data integrity + continuous assurances. Computers and Security 2005;24:604–13.

Helms GL, Mancino J. Wave good-bye to the paper trail. Electronic auditor. Available from: http://www.aicpa.org/pubs/jofa/apr98/helms.htm; 1998 [retrieved 7.03.2005].

Helms GL, Mancino JM. The CPA & the computer: information technology issues for the attest, audit, and assurance services functions. Available from: http://www.nysscpa.org/cpajournal/1999/0599/departments/cpac.html; 1999 [retrieved 3.05.2005].

ISO/IEC 17799. Information technology – security techniques – code of practice for information security management. International Organization for Standards. Available from: http://www.iso.org/iso/en/ISOOnline.frontpage; 2005.

Kogan A, Nelson K, Srivastava R, Vasarhelyi M, Bovee M. Design and applications of an intelligent financial reporting and auditing agent with net knowledge. Available from: https://kuscholarworks.ku.edu/dspace/bitstream/1808/141/1/srivastava.pdf; 1998 [retrieved 10.05.2005].

Murthy US, Groomer SM. A continuous auditing web services model for xml-based accounting systems. International Journal of Accounting Information Systems 2004;5:139–63.

Onions RL. Towards a paradigm for continuous auditing. Available from: http://www.auditsoftware.net/community/how/run/tools/Towards%20a%20Paradigm%20for%20continuous%20Auditin1.doc; 2003 [retrieved 1.04.2005].

Pinkster R. XBRL awareness in auditing: a sleeping giant? Managerial Auditing Journal 2003;18(9):732–6.

Rezaee Z, Elam R, Sharbatoghlie A. Continuous auditing: the audit of the future. Managerial Auditing Journal 2001;13(3):150–8.

Rezaee Z, Sharbatoghlie A, Elam R, McMickle P. Continuous auditing: building automated auditing capacity. Auditing: A Journal of Practice and Theory 2002;21(1):147–63.

South African Institute of Chartered Accountants. SAICA handbook – auditing. 2003/2004 ed., vol. 2; 2003.

Srinivas S. Road map to XBRL adoption as a new reporting model. Information Systems Control Journal 2004;1.

Taylor P. The perils of systems-based fraud. IT Audit 2005;8(January 15).

The IT Governance Institute. (COBIT) Control objectives for information and related technology. 4th ed. USA: The IT Governance Institute; 2005.

Vasarhelyi MA. Concepts in continuous assurance. Available from: http://raw.rutgers.edu/continuousauditing/conceptsincontinuousassurance13final.doc; 2002 [retrieved March, 2005].

Varsarhelyi MA, Halper FB. The continuous audit of online systems. Auditing: A Journal of Practice and Theory 1991;10(1).

Wessmiller R. Facing the data integrity challenge. Available from: http://www.theiia.org/itaudit/index.cfm?fuseaction=print&;fid=440; 2002 [retrieved 20.04.2005].

Woodroof J, Searcy D. Continuous audit: model development and implementation within a debt covenant compliance domain. International Journal of Accounting Information Systems 2001;2:169–91.

Stephen Flowerday is currently a final year full-time doctoral student at the Nelson Mandela Metropolitan University in South Africa. His research focus is on providing real-time assurances for information integrity. This is within the domain of corporate governance and information security management. In addition to his studies he lectures part-time and before entering the academic field he had a successful career in management consulting.

Adrian Blundell is presently completing a full-time Master's degree in Information Technology at the Nelson Mandela Metropolitan University in Port Elizabeth, South Africa. He is researching the roles of various technologies within continuous auditing systems. His qualifications include a National Diploma IT from the Port Elizabeth Technikon and a B. Tech IT from Nelson Mandela Metropolitan University.

Professor Rossouw von Solms is the Director of the Institute for ICT Advancement at the Nelson Mandela Metropolitan University in South Africa. He holds a PhD from the Johannesburg University. He has been a member of the International Federation for Information Processing (IFIP) TC 11 committee since 1995. He is a founder member of the Technikon Computer Lecturer's Association (TECLA) and is an executive member ever since. He is also a vice-president of the South African Institute for Computer Science and Information Technology (SAICSIT). He has published extensively in international journals and presented numerous papers at national and international conferences in the field of Information Security Management.