Aiding the Audit: Using the IT Audit as a Springboard for Continuous Controls Monitoring

Ryan Teeter, Ph.D. Student, Rutgers University, email: teetery@rutgers.edu

Rod Brennan, Director of IT Audit, Siemens Corporation, email: rod.brennan@siemens.com

Work in Process

Abstract: Demand for continuous controls monitoring (CCM) has increased in recent years as software platforms have matured and ERP systems have become more robust. The question remains of how to effectively implement the CCM model. This study shows how an ERP Certification Audit can be used as a springboard for effectively creating rulebooks and exception reports that both fulfill requirements of the CCM model as well as aid the audit itself. The researcher followed the certification audit of a large, recently acquired software company and worked with the internal audit team to create an automation rulebook based on the parent company's existing audit program. By the end of the systems audit, rules for approximately 63% of the controls tests were successfully automated. Additionally, due to the homogeneity of the parent company's information systems, the rulebooks and reports created can be easily adapted for use with other subsidiaries without the need to repeat the entire creation process. This paper details the benefits and limitations of using the IT audit as a conduit for CCM implementation.

Keywords: Continuous monitoring, continuous controls monitoring, control settings, automation, reengineering.

1. Introduction

Information technology (IT) has become an essential driver in providing real-time accounting and business process information. Consequently, evaluation of controls for these critical information systems has received considerable attention by academia, the press, and governmental regulators, particularly in the wake of the Sarbanes-Oxley Act (see Brown et al, 2007 and Damianides, 2005). Under this recent scrutiny, constant monitoring of internal controls, defined by the Committee of Sponsoring Organizations (COSO) as policies and procedures intended to safeguard an organization's assets from fraud or error, ensures that the information being captured by these complex technology-driven systems is accurate (2008). The importance of this evaluation process is emphasized by the June 2008 exposure draft titled "Guidance on Monitoring Internal Control Systems" which was produced by the COSO in response to requests for direction from auditors and managers.

In their draft, the COSO (2008) provides the following model for monitoring internal controls:

Effective monitoring involves (1) establishing an effective foundation for monitoring, (2) designing and executing monitoring procedures that are prioritized based on risk, and (3) reporting the results, and following up on corrective action where necessary.

An important function of each of the steps outlined for internal control systems monitoring is provided by automated control monitoring tools, which can operate as controls and report on the integrity of other controls. These tools provide the support for *continuous controls monitoring* (*CCM*), which is the process of evaluating the effectiveness of internal controls in detecting errors and anomalies on an ongoing basis (COSO 2008). These evaluations are reported to management to help meet the firm's objectives and provide compliance with regulation. As accounting information systems have become more complex and able to provide real-time or near real-time financial information, the demand by management to evaluate risks and resolve issues as they happen, rather than solely at specific intervals, has led to the development of robust CCM platforms and tools (Brown et al, 2007; COSO, 2008).

When implemented and functioning properly, continuous controls monitoring "can enhance the efficiency and effectiveness of the whole internal control system" (COSO, 2008). Additionally, in the model proposed by Vasarhelyi et al (2004), CCM plays an important part of continuous assurance, providing the tools necessary for evaluation of business process effectiveness.

Although several models exist that demonstrate how CCM can provide meaningful feedback on the integrity of internal controls, actual large-scale implementation of CCM has been limited and remains an important and timely research problem. To provide insight for this problem, this paper presents a proof of concept implementation of CCM, based on work completed at Siemens Product Lifecycle Management (PLM), a large, recently acquired software division. This is an extension of the pilot study examined in Alles et al (2006).

When evaluating implementation of CCM, Vasarhelyi et al (2004) suggest that firms are more likely to adapt existing internal auditing programs than build a CCM model from scratch. Additionally, Alles et al (2006) indicate that utilizing the expertise of seasoned audit professionals can aid in creating effective and relevant controls tests. With an existing plan, a significant portion of the development of control tests needed for CCM has already been completed and shown to be generally effective on a periodic timescale; they need only be automated. Auditor expertise provides validation of the effectiveness of implemented controls tests, which can limit costs of reevaluating bad controls tests later. In light of these insights, performing a CCM implementation parallel to a manual information technology (IT) audit appears to be a suitable, as the manual audit provides the internal audit plan and auditor expertise necessary to create an effective CCM system.

Collected Papers of the Seventeenth Annual Research Workshop on: Artificial Intelligence and Emerging Technologies in Accounting, Auditing and Tax

Anaheim, CA USA, August 2, 2008.   129

An *enterprise resource planning* (ERP) or IT audit is performed periodically to provide risk assessment and test the controls of ERP systems, including SAP, Oracle, and other hybrid or legacy systems. The IT audit specifically tests controls that exist in the current business processes for the firm. Alles et al (2006) observe in a CCM pilot that 50% of the controls tests in their IT audit program are automatable with little or no alteration of the existing audit plan. An additional 25% of the tests have potential to be normalized and audited, but require significant reengineering in order to verify the functionality of the controls. Using the existing audit program as a base, experienced auditors can prioritize controls risks and directly test the CCM rules against typical and expected results from the periodic, manual audit.

The implementation at Siemens PLM, which is examined in this paper, provides insight into challenges and features of using continuous monitoring software platforms available from third-party vendors to automate the existing internal audit plan. A byproduct of the automation process, which is of particular interest to Siemens Corporation, is the ability to take the automation concepts and rulebooks that were created parallel with this IT audit and apply them to other divisions that have similar systems with similar audit and monitoring requirements. This paper provides additional insight into practical application of CCM.

In the next section, we examine the considerations of automating an internal controls audit plan that is typically performed as part of a traditional IT audit at Siemens, and includes an evaluation of CCM platforms that are currently available. In Section 3, we present the actual implementation. Section 4 shows our results and insights from the audit, and Section 5 presents our conclusion, including considerations for future CCM implementation.

2. Automating the IT audit

Siemens has some 70,000 employees which generate $20 billion dollars in annual sales across a variety of business sectors in the United States. In the US, the IT Internal Audit department of Siemens Corporation provides IT audit services for each company division, which include annual IT audits and system certifications. Siemens is heavily ERP and SAP R/3 centric.

As part of the IT audit, the internal auditors follow an audit plan that is comprised of approximately 300 audit action sheets (AAS). The audit action sheets identify control objectives and describe verifications and tests that should be performed to verify that the internal controls are in place and functioning properly on the ERP system. These tests include interviewing users and management, evaluating documentation and policies, verifying user access, and checking for correct separation of duties for certain functions. The AASs are divided into seven modules that follow the specific SAP business processes, ranging from Basis to Fixed Assets to Human Resources. The majority of these tests are consistent with the monitoring functions described in the COSO exposure draft: "comparing system parameters to pre-established requirements, comparing system results to pre-established tolerance levels, evaluating system access rights for possible segregation-of-duties issues, and evaluating propriety of administrator rights access" (2008).

Most of the tasks outlined on the audit action sheets were developed in-house, in many cases with the assistance of a consultant from a big four accounting firm. The internal audit team spends approximately 70 days manually pouring over tables and authorization requests within the ERP system. To aid the auditors at Siemens, a proprietary analytical tool extracts information from tables in the ERP and presents the results of individual control tests to the auditors for a small portion of the tests. The results from each test are used as evidence that the system meets or fails certification requirements.

In the case of Siemens PLM, the end result of a successful audit is an ERP systems certification, which means that a formal IT audit is not required for an extended period of time, from two to four years. The division performs annual self-assessments in the interim years. From a cost perspective, a successful audit translates into significant cost savings for Siemens Corporation, as travel and other expenses associated with the full, manual audit are mitigated until later periods.

Automating the IT audit provides a tool that fulfills two organizational objectives. For management, the IT audit already tests many specific controls and automation creates a CCM platform capable of reporting on weaknesses in a timely fashion. For the IT auditors, automated tests can be applied on systems throughout the company, resulting in time and cost savings. This last objective can be considered a bonus for organizations considering CCM implementation.

2.1 Evaluating monitoring software platforms

Prior to implementing CCM, Siemens auditors relied on a proprietary tool to analyze some of some of the controls and return results corresponding to tests found on the AASs. While this tool was helpful during the periodic audits, it was not robust enough to continuously analyze thousands of transactions and report anomalies. Even tools created by prior researchers suffer from performance issues (see Alles et al, 2006). Thus, choosing a monitoring platform that can handle CCM and provide the analytics that match the audit program is essential.

With the increased popularity and desirability of CCM, especially in response to the requirements of Sections 201 and 404 of Sarbanes-Oxley, many companies are looking for solutions, and software vendors have emerged and developed solutions to make it easier to create rules that perform the tests, in many instances by creating user-friendly interfaces for tools that perform complex SQL database queries. Development by these third-party vendors has also been spurred by the restriction of controls monitoring outsourcing on the large public accounting firms. Alles, et al (2006) indicate that as many large accounting firms develop CCM platforms and solutions internally, these solutions cannot be sourced to audit clients, due to these restrictions.

CCM tools and platforms typically analyze large databases that are periodic (typically daily) snapshots of the current ERP system. These tools function as part of a monitoring and control layer as performance and access considerations limit access to the production server (see Vasarhelyi, 2004). While most vendors tout their benefits of regulatory compliance and fraud detection, these tools vary in their applicability and implementation of CCM. Current tools and platform can be classified into three types: System-specific, Modular and mapping, and Custom.

System-specific platforms, such as Approva BizRights, provide comprehensive analysis of controls based on specific systems, such as SAP. These tools translate codes and tables from specific systems into a user-friendly interface that allows creation of system-specific rules for control checks. Rules created with these platforms can only be used on the systems for which they were created. While these tools are helpful in homogenous ERP environments, distinct instances and rules must be created for heterogeneous systems.

Modular and mapping platforms, such as ACL and OverSight, provide different CCM modules for specific business controls that meet specific business control objectives. An example of a module would be Purchase-to-Payment Cycle or Purchase Card Cycle. These platforms check for standard controls, based on widely-accepted risk frameworks, such as the one provided by COSO (1992). These modules are standardized to evaluate control objectives across different systems and data stores. By mapping existing systems to a common data model, universal rules and analyses are performed on the system. Custom libraries can also be developed, but they are not platform-specific. In some cases, ERP vendors provide monitoring modules

Custom platforms can be developed in-house to meet specific controls objectives. Siemens' proprietary tool is an example of a custom tool. Consulting firms may provide planning, implementation, and deployment of custom solutions. These platforms are particularly useful in environments where the ERP system is hybrid and requires a great deal of customization to allow sufficient control risk assessment.

A default installation of each of the different platforms provides the ability to create generic controls tests that can be applied to existing transactions. Because of the focus by vendors on SOX 404 compliance and the COSO control risk framework, many of these solutions provide monitoring functionality in the form of alarms and generic reports.

Where a manual IT audit may require evaluation of division- or company-specific transaction codes that must be explicitly evaluated, some automation tools provide support for variables and parameter lists, which can be used to create "generic" rules for use across multiple divisions or companies. When the rules are implemented elsewhere, the auditor need only enter the company-specific values into the parameter lists. While convenient, these parameter lists are a potential control weakness themselves, which we will discuss later in this paper.

In addition preset mapping of common ERP tables and codes and standard controls tests, many automation tools allow for customization. Using add-on tools provided by the vendors, such as Approva's InsightStudio, auditors have greater ability to create customized controls that are more complex and/or specific to the firm's needs. This customization ability can extend the system to scale the CCM platform over multiple systems and aid in the reengineering process.

While robust and helpful, CCM platforms aren't perfect. One drawback to any automated solution is incompleteness in the code itself. Incompleteness may include missing functionality which will not allow full automation of an audit plan. Most current software platforms suffer from incompleteness and minor, yet significant, bugs which are addressed in the results section of this paper. As more firms implement these software platforms, many of these flaws are being discovered, giving vendors an opportunity to address these issues in future revisions of their respective platforms.

In addition to platform limitations, current audit plans are ultimately subject to non-formalized tests, such as auditor interpretation. As a result, rulebooks may not supply sufficient information for an automatic control evaluation without significant reengineering of the audit plan itself. For example, with Siemens' audit plan, evaluation of the AASs was provided with a value on an ordinal scale of 0-4. While 0 and 4 could be easily evaluated based on some threshold limit or other easily identifiable indicator, determining intermediate values is more subjective. An altered audit plan would possibly require additional rules or a separate evaluation of multiple rules at once.

3. Implementation

At Siemens PLM, a team of three researchers and two internal auditors, working alongside two principal internal IT auditors from Siemens Corporation who were conducting the manual IT audit, created and implemented rules and reports that would be used to perform CCM at PLM. Once again, the objective of Siemens Corporation was not only to provide CCM for the software company they had acquired one year prior, but also to create a universally-adaptable set of rules and control tests that could aid future audits and be easily implemented at other subsidiaries and divisions of the company.

Before the CCM could be implemented, the researchers evaluated and classified the audit requirements from the existing AASs into degrees of automation, defined in the next section. Later, creation of automated rulebooks based on those requirements, and reengineering manual controls into automatable controls was performed. Throughout the entire process, feedback was solicited from management, the auditors, and other researchers so that the rules created would produce reliable and accurate results.

Collected Papers of the Seventeenth Annual Research Workshop on: Artificial Intelligence and Emerging Technologies in Accounting, Auditing and Tax

Anaheim, CA USA, August 2, 2008.   131

3.1 Classifying audit requirements into degrees of automation

Prior to and during the implementation of the CCM rulebooks, an evaluation was conducted into the automation potential of each of the controls tests that appeared on the AASs (see Alles et al, 2006). A spreadsheet was maintained that identified each audit requirement and had comments as to whether it was automatable and what needed to be done to create a rule for that requirement. For example, see Figure 1. This evaluation was important to identifying "low-hanging fruit" or trivial automation concepts, partially automatable tests, rules requiring a degree of reengineering, and non-formalizable manual tests. Tests used in the Siemens audit program generally fit into one or more of the following categories: authorization, configuration, separation of duties, transaction, use as is, baseline, or manual.

Figure 1: Classifying audit requirements into degrees of automation: sample rules

Rule Type	AAS Ref #	Short Description	Description of rule to be made	Conditions used	Status
Authorization	1.02.X	Unauthorized access to SAP system – emergency user concept	Test these authorizations: 1. S_TCODE=SM18, S_ADMIN_FCD=AUDA 2. …	AI rules	Rule Built 1.02.X
Configuration	1.02.X	System admin/completeness verification	Set up 3 rules to test the following: 1. parameter rdisp/vbdelete=0 2. parameter rdisp/vbreorg=0 3. …	Parameters are listed in report RSPFPAR	Rule Built 1.02.X
Separation of Duties	5.06.X	Particular authorizations in SAP are only granted to appropriate personnel - Posting and master data maintenance authorization	Maintain conditions, invoices, and central data	V_KONH_VKS Act 01 OR 02; V_VBRK_FKA Act 01 OR 19;…	Rule Built 5.06.X
Transaction	5.04.X	Use of one time customers	Look in vendor rule set for one time customers	Table KNA1 and Field Name XCPDX. Value equaling "X" indicated one time customer	Rule Built 1.02.X
UAI	7.01.X	Report Analysis	Determine the reports available, the frequency of use, and whether the reports contain all the necessary information in order to manage projects.	Reports used (t-code used for each report): 1. Y_D01_1400014X - CSS Proj Fin Related Info GRP Currency 2….	Rule Built 7.01.X
Baseline	7.20.X	Project profiles	Configure the system to provide a notification if any changes are made to existing project profiles, or if new project profiles are created, or existing ones are deleted.	Table TCJ4X Baseline	Rule Built 7.20.X
Manual	1.02.X	System parameters for SAP*	Involves interviews and understanding. Interviewing the company, determining how passwords are assigned, etc. are manual and must still be performed.	N/A	N/A No rule required

Collected Papers of the Seventeenth Annual Research Workshop on: Artificial Intelligence and Emerging Technologies in Accounting, Auditing and Tax

Anaheim, CA USA, August 2, 2008.   132

- *Authorization* tests verify which users currently have access to certain functions or screens in the ERP system. In results from these tests it is normal to see an appropriate number of users, based on their roles in the system. For example, a rule is created to see which users have authorization to maintain customer master data, checking any of a series of transaction codes in SAP, such as XD01, and create and edit attributes, such as activity 01 and 02.
- *Configuration* tests query the system to check that settings reflect the firm's policies. For example, checking that the production client is locked to changes (Table T000, field CCCORACTIVE is set to 2).
- *Separation of duties* tests identify users who have access to screens or reports that are in conflict with separation of duties controls. For example, a user who creates or edits a purchase order should not be able to approve it. With functioning controls, the results of these tests should be empty.
- *Transaction* tests were applied to see if transactions that appear in a table are appropriate and to see the frequency of specific transaction codes. An example of a transactional test is identifying one-time customers that are identified by an "X" in a specific field. Transactional tests may require verification as to what transactions are normal and expected for a particular function.
- *Use As Is (UAI)* tests check for timeliness and correctness of ERP functions. UAI tests generate reports if the test runs into errors.
- *Baseline* tests check for changes made to tables that are expected to have static values, such as Companies and Controlling Area. These tests generate reports any time these tables change.
- *Manual* tests consist of interviews and documentation verification that must be performed by the auditor. These tests are not automatable using the current system, and are likely to remain unautomated.

3.2 Creating rules from audit action sheets

After the tests from the AASs were classified and identified and manual or automatable, creating the actual rules in the CCM platform was fairly straightforward. An instance of the CCM platform was installed alongside the SAP R/3 production and human resources servers at Siemens PLM. The CCM server stored daily snapshots of the production and HR servers and allowed rule creation and report testing via a Web interface. Because the CCM platform used at PLM is an MC layer, it runs without an impact on performance for the production servers.

Easily automatable controls tests were identified as "low-hanging fruit" because they didn't require intense work to automate. These tests included authorization, configuration, separation of duties and UAI tests. Using the CCM platform's Web interface, creating these rules required selecting transactions from pull-down menus and entering values to be checked in the text boxes that appeared. In the system, rules were assigned numbers and names corresponding to the objectives found on the audit action sheet. Rules were also grouped into "rulebooks" based on the module they were part of. Once all of the rules for a module were completed, the entire rulebook would be executed and the results would be compared to the results from the manual audit.

More complex controls tests would be evaluated and, in most cases, partially automated. Baseline rules, for example, needed to be created using the vendor's add-on tools, because the CCM platform didn't provide automatic functionality for this type of rule. A number of the controls tests in the program were duplicated in other modules. In these cases, it was not necessary to recreate additional rules, but to identify the existing rule on the AAS so that it could be tested by the auditors.

At the request of the Siemens IT auditors, descriptions of the rules were also added, based on the description of the audit requirement outlined on the AAS. This provided the auditors with an easy way to identify which functions the rules were testing so they could minimize their own report creation. For convenience, using a vendor-provided conversion tool, rules created in the CCM platform could be exported and converted to a spreadsheet file, which made adding descriptions easier than clicking through each rule a number of times in the Web interface.

One of the tasks that posed a challenge to creating rules within the CCM platform was identifying all of the related tables and transaction codes. For example, some of the objectives on an AAS would identify a transaction code for a form that was accessed using a different code within the company. As a result, the original transaction code would not show any user instances, but a different transaction code would at this particular site. The rules, therefore, were adapted to look for both transactions.

3.3 Reengineering audit processes

At the beginning of the automation process, the automation team's efforts were focused on the "low-hanging fruit" or easily automatable objectives. Once those rules were created and tested, work shifted to reengineering of controls tests. Reengineering was essential to shift the focus from subjective controls tests that require auditor interpretation to objective tests that can be automated and produce reports that are useful to management. For example, a manual control may look like this: "Gain an understanding of X process. Verify Y function isn't allowed." The team would build a rule for the second half of the test based on a manual investigation of the first part of the test.

Each of the tests classified as "manual" were evaluated for automation potential. In some cases existing rules provided the validation necessary. In other cases, limitations in the CCM platform would allow a rule to be created but the results would not be correct, so the rule was scrapped. Successful rules would be indicated in the spreadsheet and in the end a complete list of automated and manual rules was compiled for each module.

3.4 Feedback loop

Throughout the automation process, the audit action sheets were continuously evaluated for automation potential, reengineering, and applicability for the system audit. After the tests in a particular module were formalized, the rules were created and run on the exported database. Reports showing results from the automated rule were then checked against the manual evaluation results provided by the Siemens auditors. This process provided the necessary insight into the reliability and performance of the rules in detecting anomalies.

In the cases where the automated results did not match the manual results, the rules were re-evaluated, necessary changes were made and the rulebook was run again. In many cases, the researchers and auditors spent significant time troubleshooting the rule, platform, and settings to discover why results were inconsistent. This was also helpful in determining limitations in the platform itself and providing feedback to the vendor.

Due to the importance of this implementation, support staff from the CCM platform vendor worked directly with the audit team on multiple occasions to provide training and workarounds for many of the custom rules. Based on the auditors' feedback, many of the limitations were sent to the developers to incorporate into future releases of the platform. In many cases, however, rules could not be created because of bugs that would not be addressed until the new version was released, which was a source of frustration to the audit team.

4. Preliminary Results

The results of the large-scale implementation at Siemens PLM were consistent with the pilot study presented by Alles et al (2006), although not quite to the level of automation predicted by their model. By the conclusion of the IT audit, rules for approximately 63% of the tests found on the audit action sheets were created and implemented, including easily automatable tests as well as numerous reengineered processes. This is short of the 75% automation prediction by Alles et al but still quite considerable for testing controls at Siemens PLM.

This section addresses key components of the results from the CCM implementation at Siemens PLM, including resource commitments, successes, and challenges that were encountered along the course of the audit. A discussion of firm characteristics that may have an effect on the success of the audit itself is also presented.

4.1 Time and resource commitments

For this implementation effort and research case, the research team consisted of two professors and two graduate students working part time, one full-time internal auditor, one audit manager, and one support person from the CCM platform vendor, who helped out on a needs basis for rule development and platform support. The research team worked alongside two full-time internal auditors, who were also brought on-site to conduct the manual IT audit. Cost considerations for Siemens included transportation and lodging for off-site researchers who worked part-time on the site, as well as salaries for those performing the audit and working on the CCM implementation.

The IT audit itself was performed in a little less than 70 days. Prior to the audit being performed, evaluation of the existing audit plan had been extensively conducted by the IT audit head, three professors, and two doctoral students from Rutgers. The bulk of the evaluation had been conducted in the three months leading to the actual audit, although preliminary work preceded the audit by more than a year. Other resources included in the CCM project included support staff from the IT department at PLM, two CCM platform installations (one for the production server and one for HR).

4.2 Successes

As indicated previously, the rulebooks created during the IT audit at PLM provided evidence for approximately 63% of the audit action sheets, which the IT auditors indicated were sufficiently reliable sources for the IT audit objectives. With a little additional work, these rulebooks will be refined and generalized even further so they can be used universally at other Siemens divisions.

During and subsequent to the audit, alarms were set for most of the rules, providing one of the key benefits of CCM by allowing management to quickly see controls violations and instances. The auditors are currently working with management to determine controls priority and refine alarms to address the information overload problem, cited by Alles et al (2006).

It is yet to be seen if the cost savings identified by Alles et al (2006) will be realized as they predicted, but there the IT auditors felt that having the CCM rulebooks will significantly reduce the time required for their next IT audit. One of the key cost benefits expected by Siemens of the automated system was a reduction in the audit staff time and travel expense needed. While we expect to observe some of these savings in the future, an interesting question that was brought up by the auditors. With the creation of rulebooks, they argued that there would be a shift from an audit of the controls via querying and transaction testing to an audit of the rules themselves. The cost savings, therefore, may not be as significant as previously predicted as the audit effort is redirected, rather than simply reduced.

4.3 Challenges

Most of the challenges that were identified along the course of the CCM implementation at Siemens PLM can be classified into three main issues: audit priority, platform bugs, and properly functioning basic controls.

As we were implementing the CCM rules, the main priority of the IT auditors was to complete the audit of PLM's systems. Because of this focus, some rules that were not applicable to the site were ignored by the auditors and not given adequate attention due in part to the time constraints on the actual audit. These rules will need to be developed at a later time for other sites that deal with applicable line items, such as physical inventory, which have specific controls to be tested.

One of the primary challenges that existed as we followed the IT audit turned out to be bugs in the CCM platform itself. For example, when results from one rule were compared to the manual results of the auditors, users who were locked and/or inactive in the company did not appear on the automated list. The response from the vendor was that that functionality had not been seen as a risk issue, but that the problem would be addressed in the next version of the software.

Many of the bugs we found in the software had been identified as issues that would be resolved in the upcoming release. This brings up an important concern with any implementation of CCM as well as the development cycle of the CCM platform. During the implementation, the client firm may either alter the audit plan and create workaround rules to adapt for the shortcomings or ignore rules altogether. When vendors release bug fixes or new platform versions, the changes may fundamentally alter the results of some CCM rules. When the time comes to audit the CCM test themselves, it is likely that greater expense will be incurred to reevaluate rules affected by changes in the platform architecture. The differences in ERP installations across firms create a challenge for vendors to address all of the control issues across firms.

As mentioned previously in this paper, identifying platform weaknesses and bugs required significant troubleshooting and verification with the vendor. Most were identified after automated and manual results were compared and rules were fine-tuned. Fortunately, the ability to compare results and receive feedback is a primary benefit of implementing an effective CCM alongside the traditional IT audit. Such an issue is common in software implementing state-of-the-art process improvements. As classes of software mature, they are likely to become more reliable, bug-free, and easier to use.

Finally, functioning basic controls play an important part in successful implementation of continuous controls monitoring. From the controls and settings that must be observed manually, control failure in these areas leads to lack of support for the auditors and potentially failure of the audit itself. At the end of the implementation, the auditors discovered one control weakness that significantly impacted the reliability of the results of the audit, as well as many of the CCM rules we had created.

4.4 Firm characteristics

There were a number of attributes that made for a smooth implementation of the automated rulebooks. First, it was mentioned by one of the auditors that software and technology firms generally implement better tools and controls procedures from the get go because they have knowledgeable management, auditors, and IT developers.

One auditor noted that most acquired companies fail the IT audit the first time around. This is due in part because of the different ERP systems and business processes that exist at the two firms. For Siemens, PLM was a particularly good candidate for passing the initial IT audit because it was heavily SAP-centric before the acquisition matched a significant number of controls similar to those at Siemens were already in place.

These firm characteristics are particularly relevant to the objectives of Siemens to create a comprehensive CCM and automated audit program. At the same time, the degree of success in implementing CCM at single-unit firms may be dependent on the amount of IT systems and support available.

5. Conclusion

Based on our experience at Siemens PLM and previous theoretical models presented in the literature, the IT audit proves to be a feasible starting point for implementation of CCM at a firm. This is due in part to the existing audit plan of system controls provided by the IT audit, the knowledge of experienced auditors, and real-time performance comparison and testing of automated controls tests. These three elements facilitate the implementation of CCM, aid in creating a powerful tool for periodically evaluating internal controls, and potentially provide considerable cost and time savings for internal IT auditors.

The progress made in this project with Siemens reveals several of the key benefits of CCM as an aid to the audit itself and as a springboard for implementing a platform for continuous controls monitoring. The implementation presented here is not, however, without weaknesses. Site audit priority among the auditors, system bugs, and weak controls posed challenges to creating a complete "golden" rulebook that can be used throughout Siemens Corporation by the end of the audit. However, the bulk of the work has been completed and is functional for the installation at Siemens PLM. A revision of the rules to fit the broader scope of the corporation as a whole appears to be a worthwhile, yet potentially time-consuming process.

Although not evaluated in this study, weighting of the automated rules may provide additional insight into controls effectiveness. Weighting has the potential to focus management on more important risks while not abandoning "lesser" weaknesses. The creation of a theoretical basis for attributing weights to controls and methods of control combination are very important issues

Collected Papers of the Seventeenth Annual Research Workshop on: Artificial Intelligence and Emerging Technologies in Accounting, Auditing and Tax

Anaheim, CA USA, August 2, 2008.   135

that have been extensively examined in the literature (Cushing 1974, Vasarhelyi 1980, Srinidhi and Vasarhelyi 1986, Vasarhelyi and Srindhi1989, Cash, Bailey, and Winston 1977) using reliability theory and other methods. These issues now with Sarbanes 404 and this class of automation tools become crucial. This may also aid in the reengineering process by helping redefine, combine, or eliminate some manual controls checks.

From the audit perspective, evaluating the time investment required to perform the remaining tests would be valuable. One very important insight gained from this CCM implementation came at the very end of the audit. While cost savings of the automated controls over the traditional audit were significant, there will very likely be a shift of focus from testing controls to evaluating the rulebooks used to test the controls. Additional audit requirements will be required to evaluate authorization controls to the CCM platform, appropriate access, alteration of rules and reports, etc. Platforms that utilize parameter lists will have to be evaluated on a site-by-site basis to determine if the correct parameters are used. Alteration of these parameters can affect the outcome of the rules without requiring a change to the rule itself.

Future research into CCM and continuous audit implementation will provide additional insight, including an evaluation of the effectiveness of the revised IT audit program and a closer estimate of the amount of the cost savings realized. Of particular interest is the portion of cost savings that will be reallocated to evaluating and "auditing" the CCM platform itself.

References

Alles, M., Brennan, G., Kogan, A., & Vasarhelyi, M. A. (2006). Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at siemens. *International Journal of Accounting Information Systems, 7*(2), 137-161.

Brown, C.E., Wong, J., & Baldwin, A.A. (2007). A review and analysis of the existing research streams in continuous auditing. *Journal of Emerging Technologies in Accounting*. *4*(1), 1-28.

Cash, J. I. Jr., Bailey, A.D. Jr., and Whinston, A. B. (1977). A survey of techniques for auditing EDP-based accounting information systems. *The Accounting Review* 52 (4): 813-832.

COSO (2008). Guidance on monitoring internal control systems: exposure draft. http://www.coso.org/guidance.htm. Accessed June 30, 2008.

COSO (1992). Internal control – integrated framework. http://www.coso.org/guidance.htm. Accessed June 1, 2008.

Cushing, B.E. (1974). A mathematical approach to the analysis and design of internal control systems. *The Accounting Review, 49*, 24-41.

Damianides, Marios. (2004). Sarbanes-Oxley and IT governance: new guidelines on IT control and compliance. *Information Systems Management*, *22*(1), 77-85.

Srinidhi, B.N., and Vasarhelyi, M.A. (1986). Auditor judgment concerning establishment of substantive tests based on internal control reliability. *Auditing: A Journal of Practice and Theory, 5*(2) 64.

Vasarhelyi, M.A. (1980). A taxonomization of internal controls and errors for audit research. *Proceedings of the 1980 Touche Ross University of Kansas Symposium on Auditing Problems*.

Vasarhelyi, M.A., Alles, M.A. and Kogan, A. (2004). Principles of analytic monitoring for continuous assurance. *Journal of Emerging Technologies in Accounting*, *1*(1), 1-21.

Vasarhelyi, M.A., and Srinidhi, B.N. (1989). Adaptation and Use of Reliability Concepts in Internal Control Evaluation. *Advances In Accounting*, supplement 1.