Detecting Compliance Failures in Unmanaged Processes

Yurdaer N. Doganata
IBM T. J. Watson Research, USA

ABSTRACT
The importance and the challenges of detecting compliance failures in unmanaged business processes is discussed and the process of creating and verifying internal controls as a requirement of enterprise risk management framework is explained. The effect of using automated auditing tools to detect compliance failures against internal control points in unmanaged business processes is investigated. Risk exposure of a business process due to compliance failures is analyzed and the factors that affect the risk exposure of a business process are evaluated.

INTRODUCTION
Detecting compliance failures help organizations better control their operations and remain competitive. The quality of product and services can not be ensured in a business if the processes do not conform to design goals and comply with the rules and regulations. Moreover, organizations may be subject to serious financial penalty as well as civil and penal consequences if they failed to comply with established guidelines, rules and regulations. Hence, the impact of non-compliance may have severe consequences that need to be managed either by reducing or eliminating the associated risk. Companies invest significantly on detecting compliance failures to ensure governance and manage risk. The cost of reducing the risk of being non-compliant could run into millions of dollars [1]. AMR Research survey reveals that the spending of companies on governance and risk management and compliance expected to grow to $29.8 billion in 2010, up nearly %4 over the $28.7 billion spent in 2009 [2].

Compliance can be managed relatively easy when the set of interrelated and interacting activities to achieve business goals are coordinated by business process management systems. This is the case where processes are well structured and documented. When the activities in a business process are structured enough, the transitions from one activity to another are automated by software systems. In a fully automated structured business process real time information about the status of various activities can be collected by business activity monitoring software [3]. Hence, compliance of processes against rules and regulations can be checked automatically. In such automated environments, the trace of the business operations is completely visible and it possible to know who did what and when.

In reality, business activities span multiple systems and organizations across modern enterprises, integrating legacy and newly developed software applications. There exists no single system or organization that controls the process end to end. Operations often depend on activities that rely heavily on human interaction without predefined control structures. Human actors decide what to do to achieve business goals. Since the transitions between human activities can not be fully automated or monitored by software systems, the visibility of end to end business operations is reduced. The processes that consist of such activities are called unmanaged processes. In the absence of business process management software with business activity monitoring that registers various aspects of the business operations, compliance check is usually performed manually by auditors, hence it is costly, time consuming.
There are primarily two challenges in ensuring compliance of unmanaged or partially managed processes. The first challenge is to increase the traceability of end to end operations. This requires tracking, capturing and correlating relevant aspects of the business operations. Once the visibility of the operations is increased, the second challenge is to create internal controls without depending on in depth knowledge of IT system and business application code. Creating and deploying new internal controls should be done without incurring additional IT cost. If the operations are tractable and the relevant business artifacts can be gathered, automated auditing systems and tools can detect compliance failures continuously and reduce the cost of employing auditors significantly.

In the absence of process automation software that can control and record who did what and when, the compliance check is a costly and time consuming task performed manually by auditors [2]. Automated continuous auditing systems, on the other hand, provide for an almost cost-free auditing opportunity if the initial cost of building such a system is excluded. Such a system can run continuously and performs evaluation for all process instances without adding to the cost of auditing. While continuous auditing systems eliminate or reduce the dependency on audit professionals, they are not infallible. The tools that are built to realize automated continuous auditing rely on information extraction from process events and information, including e-mail transactions between the people within the organizations. The extracted information about the processes may contain errors and due to these errors the decision on the compliance may be faulty. Moreover, the testing of a compliance condition may require a level of text analysis that is not yet available in automated systems. Hence, the automated systems can perform fast and extensive auditing of the internal control points at the cost of making mistakes. As a result, some compliance failures may be missed while some other cases that are compliant may be declared non-compliant.

The focus of this chapter is to discuss the factors that impact the effectiveness of continuous assurance with automated audit tools. The subject is important for organizations which need to determine how much they should invest to remain compliant. The chapter helps understanding the characteristics of the operational environment that affects the efficiency of automated tools and the conditions that necessitate hiring experts for manual auditing to avoid compliance failures. Ultimately, the companies expect to reduce the risk exposure at least as much as they spend for compliance assurance. Therefore, they need to know how they can optimize the return on their investment.

BACKGROUND

Businesses describe their operations in terms of activities that are performed to realize the defined business goals. These activities, their interrelations and interactions constitute the building blocks of the business processes of organizations. Traditionally business processes are enacted manually and guided by the knowledge of the people who established the business goals and the associated activities around the business goal. Manual enactment of business processes, however, are costly, time consuming and error prone since they are human centric.

Advances in computing technologies have led to the development of software systems that are used to automate the execution of some of these processes. These are called Business Process Management Systems (BPMS) [4]. While BPMS have been providing many benefits for coordinating activities within the enterprise and help improving the productivity and efficiency, a full automation of all activities in a modern enterprise does not seem realistic. This is mainly because today’s enterprise applications span across systems and organizations where newly developed software applications are integrated with legacy system. In addition, often business processes rely on human activities that cannot always be predicted and the information exchange is based on e-mail or attachments where the content is unstructured. In such environments, the next activity in a sequence of activities is not always determined by rules executed by automation software, but by the human decision. Before the advancements of computers, the business
interactions were always between people. Today, the transactions are between various resources that include both people and automated systems.

When all the transactions between activities and parties are fully controlled by automation software based pre-defined rules or process definitions, the associated process is called a “managed” process. In managed processes whenever an activity ends, the next activity is determined by the automation software. This implies that if human intervention is necessary to start or end a task, then the process cannot be fully automated, hence it cannot be called “managed”. A process which cannot be fully automated is called an “unmanaged” process. An important characteristic of unmanaged processes is that some of the transitions between its activities are non-deterministic. This means that when an activity ends, there is uncertainty about the next activity. This may cause integrity lapses in complex business environment since actual operations may differ from what they are originally designed for. In such an environment, the compliance of business processes to pre-established rules and regulation cannot be ensured.

In order to manage the risk associated with compliance failures and for the continuous assurance of business goals, organizations use Enterprise Risk Management framework such as COSO ERM [5] - [7]. ERM framework provides for a systematic way of creating internal control points as part of audit and compliance activities. Internal controls can provide a reasonable assurance that the goal of an organization is met and for some organizations they are required by law [8].

**INTERNAL CONTROLS**

Internal controls are primarily created to manage the execution of an activity or set of activities for the purpose of preventing business risk from occurring [7]. The steps of the process of creating internal controls can be outlined as identifying business process and the associated goal, assessing the risk of achieving the business goal, determining the control points to check to prevent business risk from occurring and testing if the internal controls are working as intended. A new position open process shown in Figure 1 is used to explain the steps of creating internal controls. The example can be found in IBM Websphere Lombardi manual [9]. In the example process shown in Figure 1, the hiring manager submits a job requisition for a position. If this is for a new job position, the requisition is routed to the general manager for approval. If this is for an existing position, the requisition is routed directly to human resources. The general manager evaluates the submitted requisition for a new position and either approved it or rejects it. This is represented by “Approve/reject requisition” task. If approved, the requisition is routed to human resources. Otherwise, it is terminated and the hiring manager is notified.

The business goal is to introduce an effective hiring process which requires the oversight of general manager who makes the planning for new job positions while bypassing the approval process for existing positions which have already been approved. Hence, the compliance goal associated with this business goal can be stated as: “Ensure that general manager approves every ‘new’ job positions opened”. According to the process definition, before human resources start searching for candidates for a new position, they must have the approval of the general manager. This is not necessary for existing positions. When the ‘Find job candidates” task is not automated by a process management system, i.e., enacted manually, and it is controlled by human, there may be errors. In a fully automated system, approvals are kept in a content management system and the “Find job candidates” activity may not be initialized until a notification is received from the content management system about the approval of the general manager. In contrast to fully automated systems, there is a risk of non-compliance in human centric systems where finding a candidate is a human action and the person who is responsible for this task may overlook the type of position and may ignore the rejection from general manager. Once the risk is identified, then the control point is determined to eliminate it. A potential internal control provides for a reasonable assurance for potential risk and hence, can be stated as: “There must be an evidence of general manager approval, before candidates are searched for the ‘new’ job opening”.

If the internal controls are represented in terms of the artifacts that can be captured during the execution of the business process, then automated tools can be used to check the compliance. In this example, if the approval or the rejection of general manager is linked to the job requisition id and stored, then it is possible to check if the task ‘Find job candidates’ is executed with proper approval or not. Automation of compliance checking enables continuous assurance of compliance, which requires capturing relevant aspects of the process around a control point from execution traces.

Traditional organizations use manual auditing for compliance assurance. Manual auditing involves the use of subject matter experts, but typically covers only a small set process instances because of time and cost constraints. The cost of improving the status of compliance and reduce the risk of being non-compliant by using experts could run into millions of dollars for many organizations [1]. In many cases, audits are performed in a quarterly or yearly basis, and cases are selected through statistical sampling. There is thus a trade off between the cost of sampling sufficient number of cases and the possibility of poor auditing which may cause missing opportunities for corrective action. While traditional audits are performed a few times a year, it is widely believed that compliance is an ongoing process that goes beyond testing and evaluating the internal controls of a sampled space. Thus many corporations focus on enhancing or implementing systems to ensure compliance on a continuous basis.

The COSO board recognizes the need to monitor internal controls effectively [6]. Effective monitoring of internal control point within the enterprise through internal controls requires the visibility of end to end operations. In a fully automated systems all the executions steps of the business processes are visible and execution steps are traceable. Pre-defined conditions that determine transactions between activities are embedded in the automation software and can be traced to control risk and compliance. In case of unmanaged processes, however, techniques are needed to increase the visibility of end to end operations based on the actual execution traces such as business provenance tracking.

Software systems that track the provenance is particularly important in case of unmanaged processes, since majority of compliance spending goes to manual labor of auditors and consultants who spend a lot of time to track the lineage of business tasks and items. Advances in data management, processing and
analysis systems have led to development of tools which can automate auditing process. Automated auditing took is based in business provenance technology explained that is developed to increase the traceability of end to end operations in a flexible and cost effective way.

A number of works [10], [11] advocate addressing control objectives early in design time and propose supporting mechanism for business process designers. A method is proposed in [10] to help the process designers to measure the compliance degree of a given process model against the set of objection. A language is introduced in [11] to express temporal rules about the obligations and permissions in order to help the designers at the process modeling time to validate and verify business contracts. In many traditional systems, however, the control objectives were not known at the time of process design. They are added to the risk management system later and the audit tools are customized based on the control objectives over existing business operations. In this chapter, in measuring the effectiveness of compliance tools, the control objectives are not assumed to be known at the time of process design.

The method presented in this chapter to measure the effectiveness of automated audit tools does not depend on a particular process tracking technology or control point representation. There has been an extensive research on developing business rules approach to control and influence the behavior of business processes [9][13] and attempts to formally represent internal control points [14]. These studies are left outside the scope of this chapter since the chapter focuses on detecting compliance failures directly from process instances without making any prior assumptions about the structure of the business process.

The effectiveness of the tool is measured by its capacity to detect compliance failures during the execution of an unmanaged business process. This is accomplished by identifying an internal control point and comparing the number of non-compliance instances detected in the presence and in the absence of auditing tool. As a result of this comparison, how much the traditional auditing process performed by auditors under a budget constraint can be improved by employing auditing tool is quantified. The approach is based on inferring the prevalence of non-compliance and the performance of the automated auditing tool from a set of sample test results.

In the next section, the building blocks of an automated audit tool are described.

**AUTOMATED AUDIT TOOLS**

An automated auditing tool is a software system that captures information relevant to the internal control points of a business process, puts them into context and computes the compliance status for each control point. Auditing tools rely on correlating the data extracted from the underlying IT system to the relevant aspects of business control points effectively. Hence, relating the business goals to IT level data constitutes the core of this technology as described in [15]. Figure 2 outlines the step of building such a system which starts with converting business rules and regulations into compliance goals (Step 1). Compliance goals are identified by examining the business rules and deciding what action steps are needed. In other words, from the business rules expressed in the language of business people, compliance goals are identified. In the sample ‘new position open’ process above, the compliance goals is stated as “Ensure that general manager approves every ‘new’ job positions opened” (Step 1). This lays the ground work for setting up IT rules for compliance. Once the compliance goals are identified; tasks, activities, resources, artifacts and their relations that are relevant to the identified goal are determined and mapped onto a data model. As an example, the associated activities and artifacts related to the goal defined above are ‘Approve/Reject requisition’ task, approval or rejection messages and the type of job requisition (Step 2). Recording probes collect event data and the associated business artifacts related to the compliance goal from the underlying information system and map them onto provenance data (Step 3 and 4). In the example above, the event data related to opening a job requisition, sending the approval/rejection to content management system, activities of human resource constitute the relevant data related to the
process. A “provenance graph” is then formed with the data objects constituting the nodes and the relations among the data objects the edges. The data objects are correlated by using the compliance goals and the underlying data model (Step 5). Business control points are then expressed in terms of data entities extracted from the process execution trace as graph patterns (Step 6). Hence, control points provide a bridge between various components of the business operations and the actual data that could be consumed the IT system. A business control point that can be expressed in terms of the data produced and consumed by the IT system can be computed to check compliance in step 7. Root cause analysis of compliance failures can be done by querying the provenance graph in step 9. Provenance graph is formed by running correlation rules over the provenance data object. Root cause analysis is performed by querying the provenance data.

Figure 2 Steps for compliance checking

MEASURING EFFECTIVENESS OF AUDITING TOOLS

The problem of using automated audit tools to determine compliance failures is equivalent to determining the prevalence of a condition through screening the population by using test which is fallible. The tests that are not %100 accurate have been used extensively in health services to estimate the prevalence of a medical condition or a disease. Similarly, it is possible to determine the prevalence of non-conformance in process execution traces by using automated auditing tools. In partially managed processes, where transactions are not fully controlled and are based on human centric activities, automatic auditing may not be perfect. As an example, in the “new position open’ process, the internal control requires that the evidence of general manager’s approval must exist for every new job opening. If the general manager sends her approval to the human resources via e-mail, and the e-mail is stored as an evidence of approval then the text analysis of the e-mail is necessary to confirm that the message is indeed for the approval of a specific job opening. The text analysis, on the other hand, is not error free. Hence, the automated auditing tools are fallible in making classification for compliance.

The problem of identifying non-compliant instances of business processes is a binary classification problem where the process traces are grouped into two based on if certain internal controls are satisfied or not. There is always a trade off between the quality of classification and the cost. Audits performed by expert can be assumed always correct, provided that human errors in auditing are negligible. On the other hand, there is a significant cost associated with the manual auditing. Hence, the number of process instances that can be audited manually is limited by the budget constraints. In contrast to manual auditing, the cost of auditing by automated tools can be assumed negligible allowing all process instances to be
audited without incurring extra cost. The result of such automated classification, however, is fallible. Figure 3 illustrates the comparison of manual auditing with automated auditing based on speed, accuracy and cost. As depicted in Figure 3, automated auditing is less costly, fast but less accurate at the same time.

![Figure 3 Comparison of manual and automated audit in terms of cost, speed and accuracy.](image)

The compliance officer with a limited budget has to find an optimum way of detecting compliance failures. If he decides to spend his entire budget to hire expert for audit, then it may not be possible to cover all process instances. If he decides to save money and run the automated tool, on the other hand, he may cover all process instances but there may be a significant number of false negatives which may expose the risk of non-compliance. The optimum solution lies between utilizing automated tools along with hiring some experts. Regardless, effectiveness of the automated tool needs to be measured.

In order to measure the effectiveness of the automated audit tool, a methodology is considered that enables detecting the largest number of non-compliant instances within a budget constraint. The methodology is based on evaluating all process instances by using the automated audit machine and asking experts randomly re-evaluate some of the instances marked as compliant and noncompliant by the automated machine. The approach is modeled below.

The performance of such an auditing tool is measured by its sensitivity and specificity. Sensitivity measures the proportions of actual positives (that is non-compliant cases) which are correctly identified, while specificity measures the proportions of negatives (compliant cases) which are correctly identified. The probability that a randomly selected instance is actually compliant is defined as \( Pr (I = 0) \), the probability that a fallible auditing tool labels an instance compliant is defined as \( Pr (F = 0) \) and non-compliant as \( Pr (F = 1) \). Hence,

\[ \eta, \text{ Sensitivity: } \frac{TP}{TP + FN} = Pr (F=1|I=1) \]  
\[ \theta, \text{ Specificity: } \frac{TN}{TN+FP} = Pr (F=0|I=0) \]  

where \( TP \) (True Positive) is the number of non-compliant instances labeled as non-compliant, \( FN \) (False Negative) is the number of non-compliant instances labeled as compliant, \( TN \) (True Negative) is the number of compliant instances labeled as compliant, \( FP \) (False Positive) is the number of compliant instances labeled as non-compliant. Figure 4 depicts the regions labeled as compliant and non-compliant and illustrates \( TN, TP, FN, FP \) instances.
Given a fallible auditing tool with sensitivity and specificity \((\eta, \theta)\), and given a fixed budget to fund the use of audit experts, the goal is to find out how much the detection of non-compliant process instances can be improved. As discussed before, poor auditing may cause missing opportunities for corrective action; hence the number of non-compliant cases detected as a result of auditing should be maximized. On one hand, budget constraint limits the number of cases that can be audited by using an expert. On the other hand, a fallible automated audit machine can be used to evaluate every process instance without incurring extra cost. The goal is to device a methodology for enabling to detect the largest number of non-compliant instances possible under these constraints. One possible methodology is to evaluate all process instances by using the automated audit machine and ask experts randomly re-evaluate \(M_1\) cases among the ones marked as non-compliant (Region N) and \(M_2\) among the ones marked as compliant (Region C) by the automated audit machine. This way the sample space that the experts operate is reduced. It is assumed that the budget limits the expert evaluation of only \(M = M_1 + M_2\) cases. The effectiveness of the proposed methodology can be measured by comparing the expected number of non-compliant process instances detected. If the number is higher than what experts would have determined under budget constraint without using the methodology, then it can be concluded that the methodology improves the auditing process in general.

The probability that a randomly selected process instance is labeled non-compliant, \(P(F=1)\), by the auditing tool is \((p_{11} + p_{01})\). Given the condition that the auditors work only on instances labeled as non-compliant by the tool (Region N, all \(TP\) and \(FP\) instances), probability that the auditors detect a non-compliant case is

\[
\Pr(I = 1 / F = 1) = \frac{p_{11}}{p_{11} + p_{01}}
\]  
(3)

Similarly, the probability that an auditors detects a non-compliant cases among the ones labeled as compliant by the tool (Region C, all \(TN\) and \(FN\) instances) is

\[
\Pr(I = 1 / F = 0) = \frac{p_{10}}{p_{00} + p_{10}}
\]  
(4)

Hence, the average number of non-compliant cases detected by using this method can then be found as below where the function \(W\) is called the “worth” of this method.

\[
W = M_1 \cdot \frac{p_{11}}{p_{11} + p_{01}} + M_2 \cdot \frac{p_{10}}{p_{10} + p_{00}}
\]  
(5)
The worth function is maximized by making the experts work either in the region labeled as compliant (Region C) or as non-compliant (Region N) depending on the values of \( \frac{p_{11}}{p_{11} + p_{01}} \) and \( \frac{p_{01}}{p_{00} + p_{01}} \) provided that the budget constraint \( M \) is less than the size of both regions. This is a reasonable assumption since the size of process instances in both regions are usually much larger than \( M \). Hence,

\[
\max \{W\} = \begin{cases} 
M \frac{p_{11}}{p_{11} + p_{01}} & \frac{p_{01}}{p_{01} + p_{00}} \leq \frac{p_{11}}{p_{11} + p_{01}} \\
M \frac{p_{10}}{p_{10} + p_{00}} & \frac{p_{10}}{p_{10} + p_{00}} \geq \frac{p_{11}}{p_{11} + p_{01}} 
\end{cases} \tag{6}
\]

In the absence of auditing tool, we would only rely on the efforts of the audit experts. The average worth of this practice would then be the product of \( M \) and the prevalence of non-compliance, \( p \). Let \( W_0 \) be the worth of using only experts as auditors, the expected worth is then

\[
W_0 = Mp \tag{7}
\]

Potential improvement of using auditing tool can then be measured by the ratio of the worth functions \( \max \{W\} \) and \( W_0 \), i.e., improvement, \( \Omega \), is equal to \( \frac{\max \{W\}}{W_0} \). From (6) and (7), the improvement function \( \Omega \) is found as:

\[
\Omega = \begin{cases} 
\frac{p_{11}}{(p_{11} + p_{01})p} & \frac{p_{01}}{p_{01} + p_{00}} \leq \frac{p_{11}}{p_{11} + p_{01}} \\
\frac{p_{10}}{(p_{10} + p_{00})p} & \frac{p_{10}}{p_{10} + p_{00}} \geq \frac{p_{11}}{p_{11} + p_{01}} 
\end{cases} \tag{8}
\]

where \( p \) is the prevalence of non-compliance among the process instances, that is the probability that a randomly selected process instance is actually non-compliant or \( Pr(I=1) \). Equation (8) shows that the improvement is the ratio the probability that a randomly selected process instance is found non-compliant by the expert and the prevalence. Depending on the region (labeled compliant or non-compliant) experts work the probability that a randomly selected process instance is found non-compliant by the expert is either \( \frac{p_{11}}{(p_{11} + p_{01})} \) or \( \frac{p_{10}}{(p_{10} + p_{00})} \). It is clear from the equation that there is always going to be improvement since the probability detecting non-compliant instances by employing only auditors in the specified region is greater than \( p \). The reason for this is that the prevalence of the region labeled by auditing tool is greater than \( p \) as long as labeling is not done randomly.

Equation (8) is an expression for the improvement factor, however not useful if the probabilities are not tied to sensitivity and the specificity of the automated audit tool. In order to simplify the calculations, the joint probabilities \( p_{11}, p_{01}, p_{10}, p_{00} \) are approximated with their mean values by using the sensitivity, \( \eta \), and specificity, \( \theta \), definitions given in equations (1) and (2)

\[
p_{00} = P(I=0, F=0) = \theta . (1-p) \tag{9}
\]
\[
p_{10} = P(I=1, F=0) = (1-\eta).p \tag{10}
\]
\[
p_{01} = P(I=0, F=1) = (1-\theta). (1-p) \tag{11}
\]
\[
p_{11} = P(I=1, F=1) = \eta . p \tag{12}
\]
It is a common practice to assume that prior information is in the form of a beta density distribution for prevalence, specificity and sensitivity following the Bayesian approach in the presence of misclassification [17]. The reason for selecting Beta distribution is that it is a flexible family of distribution and a wide range of density shapes can be derived by changing the associated parameters of a beta distribution [20]. It is also a conjugate prior distribution for the binomial likelihood which simplifies the derivation of the posterior distribution significantly. Let Beta(\(\alpha, \beta\)) denote a probability density function of a beta distribution with parameters (\(\alpha, \beta\)) and assume that that prevalence, sensitivity and specificity are independent beta distributions as Beta(\(\alpha, \beta\)), Beta(\(\alpha_1, \beta_1\)) and Beta(\(\alpha_2, \beta_2\)), then the following approximations can be derived from equations (9) – (12) and the fact that mean value of a Beta(\(\alpha, \beta\)) distribution function is given as \(\frac{\alpha}{\alpha + \beta}\):

\[
p_{00} \sim E(\theta,(1-p)) = \frac{\alpha_2 \beta}{(\alpha_2 + \beta_2)(\alpha + \beta)}
\]

(13)

\[
p_{10} \sim E((1-\eta)p) = \frac{\alpha \beta_1}{(\alpha_1 + \beta_1)(\alpha + \beta)}
\]

(14)

\[
p_{01} \sim E((1-\theta),(1-p)) = \frac{\beta_2 \beta}{(\alpha_2 + \beta_2)(\alpha + \beta)}
\]

(15)

\[
p_{11} \sim E(\eta,p) = \frac{\alpha \alpha}{(\alpha_1 + \beta_1)(\alpha + \beta)}
\]

(16)

The improvement factor is then calculated from equation (8) as a function of \(\alpha\) and \(\beta\) parameters of the beta distribution functions of prevalence of non-compliance, sensitivity and specificity of the audit tool. In order to better understand the effect of sensitivity and specificity on the improvement as a function of prevalence, improvement function can be approximated as follows by using (13)-(16) as follows:

\[
\Omega = \frac{1}{p(1-\psi) + \psi}
\]

(17)

where

\[
\psi = \begin{cases} 
\frac{1-\theta}{\eta} & \text{if } \frac{p_{01}}{p_{01} + p_{00}} \leq \frac{p_{11}}{p_{11} + p_{01}} \\
\frac{\theta}{\eta} & \text{if } \frac{p_{01}}{p_{01} + p_{00}} > \frac{p_{11}}{p_{11} + p_{01}} \\
\frac{1-\eta}{\theta} & \text{if } \frac{1-\theta}{\eta} \leq \frac{p_{01}}{p_{01} + p_{00}} \leq \frac{\theta}{\eta} \leq \frac{1-\eta}{\theta} \\
1-\eta & \text{if } \frac{\theta}{\eta} > \frac{1-\eta}{\theta} 
\end{cases}
\]

(18)
In Figure 5, the percentage improvement is plotted as a function of ψ for various prevalence values changing between 0.1 and 1. The performance of the auditing tool for each key control point is also mapped on the same figure. In general, the improvement percentage is significantly higher when both the prevalence and ψ are small and it converges to zero as ψ converges to 1. In order to explain this behavior, without lack of generality, let’s focus on the case where experts are asked to examine only process instances labeled as non-compliant (Region N) by the automated machine. Hence, equation (20) becomes \( \psi = (1 - \theta) / \eta \) and indicates that as the specificity and the sensitivity increases, the value of ψ decreases. In effect, the improvement percentage increases. This is expected since as the sensitivity of an audit machine improves, the likelihood of detecting non-compliant process instances by using the tool improves as well. This is why the improvement percentage is higher for smaller ψ values. On the other hand, as ψ approaches to 1, i.e., the sum of the specificity and the sensitivity approaches to one, improvement disappears. The reason for this is that in this case the likelihood of detecting non-compliant process instances approaches to, \( p \), the prevalence. This can be explained as follows: When sensitivity is equal to (1 – specificity), the likelihood of having false positives becomes equal to likelihood of having true positives. In other words, the following holds

\[
\eta = \frac{TP}{TP + FN} = 1 - \theta = 1 - \frac{TN}{TN + FP} = \frac{FP}{TN + FP}
\]

Equation (21) implies that

\[
\frac{TP}{FP} = \frac{TP + FN}{TN + FP}
\]
where \( TP, TN, FP \) and \( FN \) are the number of true positive, true negative, false positive and false negative observations respectively. Further manipulation of equation (20) yields:

\[
\frac{TP}{TP + FP} = \frac{TN + FP}{TN + FP + TP + FN} = p
\]

(21)

Here, \( TP, TN, FP \) and \( FN \) stand for the numbers of true positives, true negatives, false positives and false negatives. Note that this is also equal to the likelihood of detecting non-compliant instances in region \( N \):

\[
\frac{p_{11}}{(p_{11} + p_{01})} \approx \frac{TP}{TP + FP} = p \quad \Rightarrow \quad \Omega = 1
\]

(22)

Equation (22) shows that working in region \( N \) does not give any advantage since the detecting a non-compliant process instances in this region is equivalent to the prevalence. The same argument holds for the other region without lack of generality. This means that labeling process instances with the auditing tool does not improve the rate of detecting non-compliant instances if the sensitivity of the automated machine is equal to \( 1 – \text{specificity} \).

Using sample process instances to measure effectiveness

As seen above, the effectiveness of automatic compliance detection depends on the prevalence of non-compliance as well as the performance of the detection tool. In order to measure the effectiveness of using automated tools to manage the risk around a particular internal control, prevalence of non-compliance for the internal control, \( p \), and the associated sensitivity and the specificity of the automated tool need to be inferred. Inference about prevalence \( p \sim \text{Beta}(\alpha, \beta) \), sensitivity \( \eta \sim \text{Beta}(\alpha_1, \beta_1) \) and specificity \( \theta \sim \text{Beta}(\alpha_2, \beta_2) \) can be drawn by running a test using the auditing tool and observing true positives and false negatives as depicted in Table 1. The technique is well known in the literature as Gibbs sampler algorithm \([18]-[20], [23] \). Gibbs sampler is an iterative Markov-chain Monte Carlo technique developed to approximate intractable posterior distributions. The algorithm uses the observed data to compute the posterior distributions of prevalence, specificity and sensitivity by applying Bayes’ theorem and conversely computes the distributions of the observed data by using the prior distributions of prevalence, sensitivity and specificity as described in \([23] \).

Let the actual status of an internal control point is known for \( N \) sample process instances and the auditing tool marks \( n_{10} + n_{00} \) instances as non-compliant \((F=1)\) and \( n_{11} + n_{01} \) instances are marked as compliant \((F=0)\) where \( n_{11} \), \( n_{10} \), \( n_{01} \) and \( n_{00} \) are the number true positives, false negatives, false positives and true negatives respectively as shown in Table 1.

<table>
<thead>
<tr>
<th>Test (Automated audit)</th>
<th>Truth (Audit expert)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 1 Test results of \( N \) sample process instances
Gibbs sampler derives posterior probability distributions that best fit given prior distributions \( \text{Beta}(\alpha, \beta) \), \( \text{Beta}(\alpha_1, \beta_1) \) and \( \text{Beta}(\alpha_2, \beta_2) \) and observed data, \( n_{11}, n_{10}, n_{00} \) and \( n_{01} \). As described in [17], arbitrary starting values can be chosen for each parameter. Gibbs sampler converges to the true values of the posterior distributions after running tens of thousands of iterations.

Reference [22] provides for an on-line calculator to estimate the true prevalence based on testing of individual samples using a test with imperfect sensitivity and/or specificity. The input values required for the calculator are illustrated in Figure 6. That is if the test results over \( N \) sample process instances as shown in Table 1 are known then the input values for the calculator can be derived. These include the number of samples (process instances) tested, the number of samples labeled positive (non-compliant), \( \alpha \) and \( \beta \) parameters for prior prevalence, sensitivity and specificity distributions, number of iterations to be simulated in the Gibbs sampler, number of iterations to be discarded to allow convergence of the model and initial number of true positives \( n_{11} \) and false negatives \( n_{10} \). The initial \( \alpha \) and \( \beta \) values for prior distributions are selected by using the fact that \( \frac{\alpha}{\alpha + \beta} \) is the mean value of a beta distribution and by using the observed data given in Table 1. As an example, the mean value of the prior prevalence can be approximated as \( \frac{N_1}{(N_1+N_0)} \). Hence, the beta value for prior prevalence is approximated as \( N_0 \) and the alpha value is approximated as \( N_1 \). Similarly, the initial alpha value is \( n_{11} \), initial beta value is \( n_{10} \) for sensitivity and the initial alpha value is \( n_{00} \), initial beta value is \( n_{01} \) for specificity.

![Prevalence Calculator](image)

Figure 6 Input and corresponding output values of the prevalence calculator

For the sample internal control point described in ‘new position open process’ and stated as “Evidence of general manager approval must exist, before candidates are searched for the ‘new’ job opening”, assume 135 process instances are examined and found that 8 of these instances are not compliant. The auditing tool runs over the same sample set and produces 7 true positives \( n_{11} \), 1 false negative \( n_{01} \), 8 false positives \( n_{10} \) and 119 true negatives \( n_{00} \). Hence, \( N_1 = n_{11} + n_{10} = 15 \); \( N_0 = n_{01} + n_{00} = 120 \). Once the initial values are calculated and entered into the prevalence calculator, the mean values of prevalence, sensitivity and specificity are found as

\[
E(p) = 0.074, \ E(\eta) = 0.826 \text{ and } E(\theta) = 0.934.
\]

From equation (17), the improvement factor, \( \Omega \), is calculated 6.75. Therefore, the percentage improvement is found as \( (\Omega - 1) \times 100 = 575\% \). This is the improvement obtained by using automated auditing tools to check the compliance of the internal control point which is stated as “Evidence of general manager approval must exist, before candidates are searched for the ‘new’ job opening” under budget constraint. In real life, there are thousands of process instances which cannot be audited manually under budget constraints. Here, after the automated tool labeled process instances, experts are asked to audit only those process instances marked as non-compliant (positive). For the sample set, automated audit tool marks only 15 cases as non-compliance. Hence, manual auditing cases is reduced from 135 to 15.
PRACTICAL CONSIDERATIONS

As mentioned in the introduction section, there is a cost associated with non-compliant process instances. The cost is determined by the amount of penalties that the company will pay for not complying with the rules and regulation as well as the cost of not being able to ensure quality and remain competitive. Compliance officer are responsible to determine the amount of risk exposure for being non-compliant and find ways to minimize the risk.

Risk exposure is the cost of being non-compliant for all process instances that are subject to auditing and it can be reduced by auditing internal controls for every process instance, detecting and eliminating the cause of non-compliance. While risk exposure can be completely eliminated by auditing every process instance, this may not be a cost effective solution. One of the challenges in reducing the risk exposure is to decide how much budget should be allocated for auditing. Budget allocation must be sufficient enough to justify the investment by reducing the cost associated with risk exposure. Risk exposure is proportional to the number of process instances, percentage of instances covered by audit and the penalty paid for every non-compliant case. In order for the investment to make financial sense, the return of investment must be at least positive. In this case, return is the amount that risk exposure is reduced. A company is expected to reduce the risk exposure at least as much as it spends for compliance assurance.

In the previous section, it was shown that by using auditing tools and limited manual auditing the detection of non-compliance instances can be improved by a factor, $\Omega$, as expressed in equation 17. Improvement factor depends on the prevalence of non-compliance, sensitivity and the specificity of the auditing tool. Note that this is an improvement over what can be done manually on limited set of process instances. Due to budget constraints, the set is usually much less than the total number of process instances. Hence, by using the methodology described in the previous section and the automated audit tool, the prevalence of non-compliance can be reduced by a factor of $\lambda \Omega p$ where $0 < \lambda < 1$ is the ratio of the process instances that can be audited manually within the budget constraint.

By detecting and fixing some of non-compliant cases within the set of all process instances, the prevalence of non-compliance is improved since there is less number of non-compliant instances after the detected non-compliant instances are fixed. As a result, the new prevalence of non-compliance is found as follows:

$$ p' = p(1 - \lambda I) = p(1 - \frac{\lambda}{p(1-\psi) + \psi}) \quad \text{for} \quad \lambda I \leq 1 $$

(23)

Since the risk exposure is proportional to the prevalence, then the percent of reduction in risk exposure, $\Phi$, can be found as $100 \times \lambda \Omega$

$$\Phi = 100 \times \left( \frac{\lambda}{p(1-\psi) + \psi} \right) \quad \text{for} \quad \lambda \leq p(1-\psi) + \psi$$

(24)

In other words, if the risk exposure has to be reduced $\Phi\%$ then (24) has to be satisfied. Equation (24) has practical implications. Desired reduction percentage may not be achieved due to the constraint on $\lambda$, the ratio of the process instances that can be audited manually within the budget constraint and $\psi$, the performance measure of the tool for a given prevalence of non-compliance, $p$. This means that for a given $p$, it may not be possible to build a tool that could reduce the risk at a desired level. Figure 7 shows the risk exposure reduction percentage as a function of $\psi = (1 - \theta)/\eta$ which is a measure of automated audit tool performance. Hence, in order to reduce the risk exposure to the desired level for a given prevalence
$p$, $\lambda$ and $\psi$ values should be selected as plotted in Figure 6. The values of $\lambda$ and $\psi$ determine the operating point where $\lambda$ is controlled by the number of manually audited process instances and $\psi$ is related to the performance measure of the tool which is tunable. Figure 6 shows that risk exposure is reduced more with higher $\lambda$ values and lower $\psi$ values. As an example, when the prevalence of non-compliance is, $p=0.3$, then the risk exposure can be reduced %20 provided that the operating point is $\lambda=0.1$ and $\psi=0.3$. Since $\lambda$ is the ratio of the process instances that can be audited manually within the budget constraint, it is directly proportional with the cost of hiring auditors for manual auditing. $\psi$, on the other hand, is the ratio of $(1\text{-sensitivity})$ to specificity of the automated audit tool and there is a cost associated with building tools with desired performance tooling. Hence, the reduction in risk exposure must be large enough to cover the cost of hiring experts and tuning automated auditing tool for the desired performance.

![Graph](image)

*Figure 7 Reduction of risk exposure as a function of $\psi$ for different $\lambda$ values when prevalence is 0.3.*

Figure 7 shows that risk exposure is reduced most with higher values of $\lambda$. This is expected since $\lambda$ is related to the number of process instances manually audited by experts that are labeled as non-compliant by the auditing tool. As $\lambda$ increases, the number of actually non-compliant process instances in the system is reduced along with the risk. In the example depicted by Figure 7, when $\lambda=0.9$, i.e., 90% of all process instances labeled non-compliant examined by experts, and the $(1\text{-sensitivity})$/specificity of the audit tool is 0.9, the risk exposure is almost completely eliminated. This is the case when either the sensitivity or the specificity of the tool is very high, hence almost all the process instances labeled non-compliant are actually non-compliant and they are all detected and eliminated by experts.

Figure 8, on the other hand shows the effect of prevalence of non-compliance on the risk exposure reduction when $\lambda$ is constant. The designers of risk management systems need to know how the sensitivity and the specificity values of the automated tool should be tuned to reduce the exposure to the desired amount when the prevalence of non-compliance is constant.
Figure 8. Reduction in risk exposure as a function of $\psi$ for different prevalence values when $\lambda = 0.3$

Achieving the desired risk exposure reduction for a given prevalence of non-compliance may not be possible with an automated audit tool, if the sensitivity and the specificity measures of the tool cannot be tuned. Figure 9 demonstrates this fact for different $\lambda$ and prevalence, $p$, values. As an example for $\lambda = 0.3$ and $p=0.1$, represented as the solid line in the second group of curves, the risk exposure reduction cannot be more than 60% no matter how good the auditing tool is.

Figure 9. $\psi = (1\text{-sensitivity})/\text{specificity}$ as a function of desired level of risk exposure reduction.
On the other hand, if $p=0.1$, reduction in risk exposure can be increased to 70% by adjusting the $\psi$ level to 0.2. This means that even if the sensitivity of the tool is 1, i.e., the tool is capable of identifying all actual non-compliant cases, the specificity of the tool, i.e., the proportions of negatives (compliant cases) which are correctly identified must be larger than 0.8. Since $\psi \leq 0.2$ can only be satisfied when specificity is greater than 0.8 if sensitivity is 1.

**FUTURE RESEARCH DIRECTIONS**

Designing effective automated auditing tools reduces the auditing cost and the risk exposure associated with the non-compliant process instances. The effectiveness of an automated audit tool, however, depends on the sensitivity and the specificity of the tool in detecting the compliance failures. The question of how to design an automated auditing system with the desired performance measures is an area that needs further investigation. This should start with defining the factors around a control point that would affect the design of an effective automated audit system. There are many factors that cannot be controlled in partially managed or unmanaged processes. These factors are mostly related to human interactions supported by collaboration software such as e-mail, calendar system, and others. In some cases, detecting compliance requires the analysis of a document or the body of an e-mail to decide if the document is sufficient to satisfy the compliance. In such cases, the performance of the auditing tool is directly related to the accuracy of the text analysis software. Since it is not practical to build automated audit system for every control point, it is important to know how to tune an automated system for a particular control point. If tuning is not possible to achieve the desired risk exposure reduction, then the visibility of the underlying operations may need to be increased. There are no established standards for measuring the visibility of business processes. Hence, the impact of business process visibility on designing effective automated audit tools could not be established either. If the visibility of a business process can be measured in terms of the likelihood of detecting compliance failures, then the internal controls can be set more properly and the design of the automated audit systems may be improved more.

**CONCLUSION**

Detecting compliance failures effectively and eliminating them is important for the health of a business and a challenge for processes that are not managed. Processes with low levels of automation, which are essentially unmanaged processes, rely on an efficient auditing procedure as the only way to prevent systemic non-compliance. In order to increase auditing efficiency for unmanaged processes, automated auditing tools can be used to complement manual auditing by subject matter experts and expand the amount of process instances that can be audited within budget constraints. The effectiveness of an automated audit tool depends on both the prevalence of non-compliant cases as well as the performance of the tool. Measuring the effectiveness of auditing procedures in reducing the risk exposure of a business process help making smarter decision on employing subject matter experts and utilize automated audit tools. Organizations need to understand how effectively they can reduce the risk exposure with the allocated auditing budget. Reader of this chapter will find enough details and analysis for the factors that impact risk exposure reduction. In particular, the chapter is expected to help compliance officer to evaluate the return of their compliance investments in terms of reduced risk exposure.

**REFERENCES**

KEY TERMS & DEFINITIONS (SUBHEAD 1 STYLE)

Unmanaged business processes: Business processes where the transaction between activities cannot be automated by software.

Business process visibility: Ability to capture business artifacts related to business activities.

Business provenance: Lineage of the artifacts created during business process execution that includes resources, tasks, data and their relations.

Internal control point: A procedure to ensure that process adheres to policies, rules or regulations.

Compliance: Confirmation that the process meets the requirements of prescribed rules and regulations.

Automated audit tool: A software application that evaluates the process instances to provide an assessment of internal control points.

Risk exposure: The expected financial penalty to be paid for being non-compliant.

Prevalence of non-compliance: The ratio of non-compliant process instances in the set of all process instances.