# Measurement of Compliance Distance in Business Work Practice

Ruopeng Lu, Shazia Sadiq, Guido Governatori
School of Information Technology and Electrical Engineering
The University of Queensland, Brisbane, Australia
email: {ruopeng,shazia,guido}@itee.uq.edu.au

**Abstract**

Ensuring that work practice is compliant to regulations and industrial standards is an increasingly important issue in business systems. Whereas as an understanding of control objectives that stem from various legislative, standard and contractual sources may be found at strategic or tactical levels, an assessment of their effective adoption in operational practices is extremely hard. In this paper, we propose a method for assessing the level of compliance in business work practice. The method builds upon business process management platforms, and provides the ability to objectively measure the compliance distance of existing processes within the organization. This in turn empowers process designers and business analysts to quantify the effort required to achieve a compliant process.

**Keywords:** Governance, Risk and Compliance, Control Objectives, Business Process Management.

## 1 Background and Motivation

Recent high profile corporate scandals such as Enron (USA) and HIH (Australia) have created unprecedented pressures on compliance and risk management for practically all industry sectors, but particularly in financial services. In spite of mandated deadlines there is evidence that many organizations are still struggling with their compliance initiatives.

Compliance essentially means ensuring that business processes, operations and practice are in accordance with a prescribed and/or agreed set of norms. Compliance requirements may stem from legislature and regulatory bodies (e.g. Sarbanes-Oxley, Basel II, HIPAA), standards and codes of practice (e.g. SCOR, ISO9000) and also business partner contracts. Compliance directives are complex, vague and require interpretation. Often in legalese, these mandates need to be translated by experts in order to relate them to organizational contexts.

IS Management 25, 4: 344-355

The original publication is available at http://dx.doi.org/10.1080/10580530802384613.

Business will typically deal with a number of regulations/standards at one time which may have overlapping and even conflicting requirements.

Compliance is typically managed in conjunction with risk assessment, and is predominantly viewed as a burden, although there are indications that businesses have started to see the regulations as an opportunity to improve their business processes and operations. Industry reports (BPM forum, 2006) indicate that up to 80% of companies said they expected to reap business benefits from improving their compliance regimens.

In general, a compliance regimen must include three interrelated but at the same time rather distinct perspectives on compliance: corrective, detective and preventative procedures that collectively form a holistic approach to compliance management.

**Corrective** measures can be undertaken due to a number of reasons, ranging from the introduction of a new regulation, to breech reporting, to the organization coming under surveillance and scrutiny by a control authority, or in the worst case an enforceable undertaking. Corrective measures undertaken in a proactive manner positions the organization favorably with regulators or other control authorities.

**Detective** measures are typically based on reporting and traditional audits conducted for "after-the-fact" detection, often through manual checks. Recent tools are providing some level of automation wherein proposed solutions hook into variety of enterprise system components (e.g. SAP HR, LDAP Directory, Groupware etc.) and generate audit reports against hard-coded checks performed on the requisite system. Business intelligence (BI) and related technologies are complementary to this activity. However, this approach still resides in the space of "after-the-fact" detection. Although, the assessment time is reduced, and correspondingly the time to remediation and/or mitigation of control deficiencies is also improved. This improvement is much sought after as is evident from the heavy investment in compliance software during the last few years (Hagerty, 2006).

A major issue with the above approaches (in varying degrees of impact) is the lack of sustainability. Even with automated detection facility, the hard coded check repositories can quickly grow out of control making it extremely difficult to evolve and maintain them for changing legislatures and compliance requirements. In addition to external pressures, there is often a company internal push towards quality of service initiatives for process improvement which have similar requirements. The complexity of the situation is exasperated by the presence of dynamically changing collaborative processes shared with business partners. The diversity, scale and complexity of compliance requirements warrant a highly systematic and well-grounded approach.

We believe that a sustainable approach for achieving compliance should fundamentally have a **Preventative** focus, thus achieving compliance by design. One can observe that business process management (BPM) platforms may provide an ideal vehicle for such a model-driven approach. However, our study indicates that dealing with compliance may be a rather distinct activity within organizational structures from business process management (Sadiq, Governa-

tori, & Naimiri, 2007).

Historically, business process design has been driven by business objectives, specifically process improvement, whereas compliance is driven by control objectives. The source of objectives for the two will be distinct both from an ownership and governance perspective, as well as from a timeline perspective. Whereas businesses can be expected to have some form of business objectives, control objectives will be dictated by mostly external sources and at different times. Furthermore, there is likelihood of conflicts, inconsistencies and redundancies within the two, and hence the intersection of the two needs to be carefully studied.

This paper presents a means to study the relationship between compliance directives, and business work practice as depicted through the organization's process models and underlying execution histories. The work presented is based on a formal model of compliance directives on the one hand, and a typical BPM platform on the other. Specifically the paper presents a method to quantitatively measure the compliance distance for a particular business process within the organization against a set of compliance directives that apply on the process. This in turn allows the process owners and business users to better understand and analyze the impact of compliance related controls as well as the risk of non-compliance on the processes they manage. A re-design of the processes and the work practice they support and coordinate may or may not ensue from the above analysis depending on the associated risk appetite of the process owners. However, the path they decide to undertake is the consequence of an informed decision with a clearly articulated impact.

The remaining paper is structured as follows: In sections 2 and 3, the overall compliance by design methodology is discussed and background concepts and definitions are provided. Section 4 presents the proposed method for measurement of compliance distance between a given business process and the compliance obligations that impact on it. Section 5 and 6 respectively present reported research related to this topic and a summary of contributions derived from this paper.

## 2 Compliance by Design Methodology

Businesses have to proactively liaise with regulators, legal experts and company stakeholders in order to pursue a convincing intent to be compliant. As mentioned previously, regulations and other compliance directives need to be interpreted by experts into the business vocabulary. However, an interpretation of compliance directives to organization specific obligations is only the first step.

A recent report (BPM forum, 2006) identifies the gap between management focus on compliance related issues and IT's lack of ability to implement the critical policies and procedures. Hence it is important to note that subsequent to the identification of control obligations, the business has to set up a control infrastructure which connects with business operations. We present below an overall methodology for the manifestation of compliance related controls into

organizations' operational systems following a *preventative* approach (Sadiq, Governatori, & Naimiri, 2007).

Firstly, there is a need to provide a structured means of managing the various (expert) interpretations within regional, industry sector and organizational contexts. We identify this as a need for a **controls directory**. Control directory management could be supported by database technology, and/or could present some interesting content management challenges, but will be an essential component in the overall solution. There is some evidence in industry reports (e.g. SAP GRC Repository) that large solution vendors are producing repositories of control objectives (and associated parameters) against the major regulations.

Interpretation of regulations from legal/financial experts comes in the form of textual descriptions (see Table 1). Thus **establishing an agreement** on terms and usage between these descriptions and the business processes and constituent activities/transactions is a difficult but essential aspect of the overall methodology. Several controls may be applicable on a given business task, and one control may impact on multiple tasks as well.

A fundamental question in this regard is the appropriate formalism to undertake the task of **representing controls objectives**. Our observation is that a compliance requirement (or its translation into a control objective and subsequently internal controls) can be reduced to the identification of what obligations an enterprise has to fulfill to be deemed as compliant. The motivation to model control objectives is multifaceted: Firstly, a generic requirements modeling framework for compliance by design will provide a substantial improvement over current after-the-fact detection approaches. Secondly, it will allow for an analysis of compliance rules thus providing the ability to discover hidden dependencies, and view in holistic context, while maintaining a comprehensible working space.

Lastly, a precise and unambiguous (formal) specification will facilitate the systematic **analysis of business models** (business processes) with respect to the control objectives that impact on them. This may constitute visualization schemes (Sadiq et al., 2007), which facilitates a better understanding of the interaction between the two specifications for both stakeholders (process owners as well as compliance officers).

However, the visualization is only a first step. The new checks introduced within the process model, can in turn be used to analyse the model for *compliance distance* that can provide a quantification of the effort required to achieve a compliant process model. In this paper, we are focused on this aspect with the intention to assist process owners in analyzing and subsequently (re)designing compliant business processes. The presence of the previous phases of the methodology is assumed.

Eventually, a **redesign of process models** may need to be undertaken to include the compliance requirements. Furthermore, it is important to note that even though an approach with a fundamentally preventative focus (as described above) has several obvious benefits, it is naive to assume that there will never be a deviation from the prescribed models. Although not the focus of this paper, we mention **event monitoring** as a last aspect of the overall methodology in

order to provide a detective facility to complement the preventative approach of compliance by design.

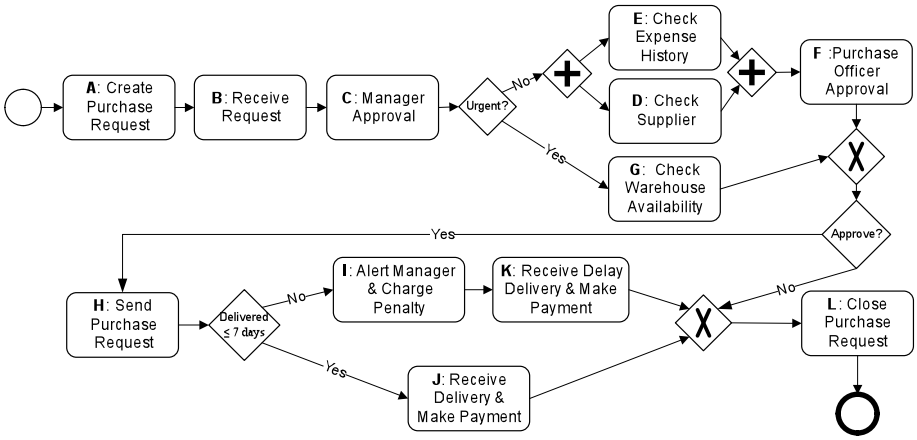


Figure 1: Example procurement process

In the remaining section, we first discuss the approach to model the control objectives and present an appropriate language for their representation, followed by a simple formalization for the business process models. We then introduce the technique to map the controls objectives and the process model into a canonical form, such that the degree of compliance in the process model can be compared with regard to the controls objectives. The subsequent discussion is based on a sample procurement process (cf. Figure 1).

The procurement process may be subject to a number of controls from various sources such as regulatory obligations, industrial standards and partner contracts. The controls will have a corresponding risk statement, and a translation to an internal control indicating effective implementation of the control objective. Typically, these internal controls cover multiple aspects of business process, such as model structure, e.g., task execution restrictions (every purchase order must be initially assessed before passing to the Manager for approval); Data integrity, e.g., every purchase request must contain a valid Purchase Order Number; Resource allocation, e.g., segregation of duty constraint (the creation and approval of purchase order must not be the same officer); Temporal restrictions, e.g., deadline (all purchase requests must be closed within $k$ days). Table 1 provides examples of such control objectives for the procurement process.

Our objective here is to be able to conduct a comparison between required internal controls and the current behavior of the business in terms of its process models and their execution. To allow for the comparison, the formal representations of compliance controls, and the business process behavior are required. The sections below present the respective formal foundations for both.

| Control Objective | Risk | Internal Control |
|---|---|---|
| Prevent illicit use of purchase orders | Unauthorized creation of purchase orders and payments to non-existing suppliers | Purchase request should be approved by both cost centre manager and purchase officer. |
| Product quality assurance | Inability to meet production quality targets | Orders to first time supplier must be approved by purchasing manager |
| | | Background check on first time suppliers must be conducted |
| Ensure adequate supply of materials | Production delays due to lack of resources or materials | Purchase requests not closed (declined or converted to Purchase Orders) within 28 days should raise an alert to purchasing manager |

Table 1: Control objectives of the procurement process

## 2.1 Modeling Control Objectives

The compliance controls can be represented in a formal language, such as Formal Contract Language (FCL) (Governatori & Milosevic, 2006). FCL is a combination of an efficient non-monotonic formalism (defeasible logic) and a deontic logic of violations. Although our work is primarily targeted at achieving compliance by design by adopting a preventative approach facilitated by business process models, the above work on formal modeling of control objectives has taken into account the violations and resultant reparation policies that may surface at runtime. We illustrate how to use this formalism to represent and reason about "normative" specifications relative to a business process. For detailed presentation of the rationale and formalism of FCL, we refer to (Governatori et al., 2006).

**Definition 1** *(FCL Rule) A rule in FCL is an expression of the form*

$$r : A_1, \ldots, A_n \Rightarrow B$$

*where $r$ is the name of the rule (unique for each rule), $A_1, \ldots, A_n$ are the premises (propositions in the logic), and $B$ is the conclusion of the rule (also a proposition of the logic).*

The propositions of the logic are built from a finite set of atomic propositions, and the following operators: $\neg$ (for negation), $O$ (for obligation), $P$ (for permission), and $\otimes$ (for violation/reparation). A simple proposition corresponds

to a factual statement. A reparation chain, for example $B_1 \otimes B_2$ captures obligations and normative positions arising in response to violations of obligation. Thus the expression above means that it is obliged to perform $B_2$, in case B1 is not fulfilled (i.e., the obligation is violated) then the "secondary" obligation B2 has to be fulfilled. The control objectives shown in Table 1 can be expressed in the following FCL rules:

*Purchase request should be approved by both cost centre manager and purchase officer for all cases.*

$$r_1 : \mathit{CreatePurchaseRequest}, \mathit{ReceiveRequest} \Rightarrow \mathit{ManagerApproval}; \mathit{PurchaseOfficerApproval}$$

*Supplier can be charged a penalty if goods not received within 7 days of receipt of goods shipment notice, while manager should be alerted.*

$$r_2 : \mathit{SendPurchaseRequest} \Rightarrow \mathit{ReceieveDelivery\&MakePayment} \otimes (\mathit{AlertManager\&ChargePenalty}; \mathit{ReceiveDelayDelivery\&MakePayment})$$

*If purchase order is not closed within 28 days the manager should be alerted.*

$$r_3 : \mathit{ReceiveDelayDelivery\&MakePayment} \Rightarrow \mathit{ClosePurchaseRequest} \otimes (\mathit{AlertManager\&CloseRequest})$$
$$r_4 : \mathit{ReceiveDelivery\&MakePayment} \Rightarrow \mathit{ClosePurchaseRequest} \otimes (\mathit{AlertManager\&CloseRequest})$$

For the ease of discussion, we use the letters associated with each task on Figure 1 to denote the tasks in the process model. $r_1$–$r_4$ can thus be denoted by:

$$r_1 : A, B \Rightarrow C; F, \quad r_2 : H \Rightarrow J \otimes I; K, \quad r_3 : K \Rightarrow L \otimes M, \quad r_4 : J \Rightarrow L \otimes M$$

## 2.2 Business Process Model

We provide below a formal definition for a simple business process model, as well as the execution sequences that can be derived from it.

**Definition 2** *Process Model A process model $W$ is a pair $(N, E)$, which is defined through a directed graph consisting a finite set of nodes $N$, and a finite set of flow relations (edges) $E \subseteq N \times N$.*

*$N = T \cup C$, where $T$ is the set of tasks in $W$, and $C$ is the set of coordinators of type $\{\mathit{Begin}, \mathit{End}, \mathit{AND\text{-}Split}, \mathit{AND\text{-}Join}, \mathit{XOR\text{-}Split}, \mathit{XOR\text{-}Join}\}$, which have typical workflow semantics.*

A sub-process is a special type of $W$, which is a fragment of a process model in which $\{\mathit{Begin}, \mathit{End}\}$ is excluded from its coordinator nodes. Given a process

model $W$ and a task $T_i \in T$, $Trigger(W, T_i)$ denotes the set of tasks that can be triggered by task $T_i$ in $W$ as the result of execution. E.g., $Trigger(W, A) = \{B\}$ (cf. Figure 1). For tasks followed by an *AND-Split* or a *XOR-Split* coordinator, we consider all subsequent tasks after the coordinator can be triggered. E.g., $Trigger(W, C) = \{D, E, G\}$, $Trigger(W, H) = \{I, J\}$. $Disable(W, T_i)$ denotes the set of tasks disabled as the consequence of executing $T_i$, which is defined to realize the semantics of the *Choice* coordinator. For example, $Disable(W, I) = \{J\}$, which means either $I$ or $J$ is executed but not both. $Initial(W)$ is a function returning the first task node in $W$.

An execution sequence of a process model referred to as the trace of execution in a process model, which reflects a possible order of task executions at runtime. Typically, a process model with parallel branches (*AND-Split*) or alternative branches (*XOR-Split*) contains more than one possible execution sequences.

For example, for tasks $G$, $D$, $E$ and $F$ in $W$ (cf. Figure 1), there are three possible execution sequences $\langle G \rangle$, $\langle D, E, F \rangle$ and $\langle E, D, F \rangle$, since $G$ and $D$, $E$, $F$ are in alternative branches, and $D$, $E$ in parallel branches.

We follow the general sequence definition to define an execution sequence: A finite sequence $s = \{s_1, s_2, \ldots, s_n\}$ is a function with the domain $\{1, 2, \ldots, n\}$, for some positive integer $n$. The $i$-th element of $s$ is denoted by $s_i$.

**Definition 3** *Execution Sequence An execution sequence $s^W$ of a process model $W$ is a finite sequence of tasks $T' \subseteq T$ in $W$, which is defined by the sequence $angle T_1, T_2, \ldots, T_n\rangle$, $n \geq 1$. An execution sequence $ss^W$ is a* subsequence *of $s^W$ if every element in $ss^W$ is an element of $s^W$, and the elements in $ss^W$ occur in the same order as in $s^W$.*

# 3 Idealness Semantics

We now introduce the concepts of Ideal Semantics (Governatori, Milosevic, & Sadiq, 2006) to provide a means of categorizing various degrees of compliance between process behavior and control rules (represented through FCL). FCL rules define a behavioral and state space which can be used to analyze how well different execution sequences comply with the FCL constraints. Our aim is to use this analysis as a basis for deciding whether execution paths of a business process are compliant with the FCL rules. The central part of this compliance checking is given by the notions of ideal, sub-ideal, non-compliant and irrelevant situations.

Intuitively an *ideal* situation is a situation where execution sequences do not violate FCL expressions, and thus the execution sequences are fully compliant with the control rule. A *sub-ideal* situation is situation where there are some violations, but these are repaired by executing additional, allowable tasks. Accordingly, processes resulting in *sub-ideal* situations are still compliant to a control rule even if they provide sub-optimal performance of the control objective. A situation is *non-ideal* if the process executes some tasks which are prohibited by the control objective, or the process fails to execute some required tasks. Finally a situation is irrelevant for a control objective if no rule is

applicable in the situation. For example consider the rule

$$r : A \Rightarrow B \otimes C$$

which means that, if $A$ occurred then it must be followed by $B$, or in alternative, in case $B$ does not occur, it must be followed by $C$. An *ideal* state for $r$ is the situation (a possible execution sequence) $s_1 = \langle A, B \rangle$. A *sub-ideal* situation can be $s_2 = \langle A, C \rangle$ where the first obligation $B$ is not fulfilled, but the obligation is repaired by $C$. We also consider $s_3 = \langle A, B, C \rangle$ a *sub-ideal* situation since it is not obliged to perform $C$ when $B$ is already in place. The *non-ideal* situation is $s_4 = \langle A \rangle$. An *irrelevant* situation is $s_5 = \langle B, C \rangle$, where $\langle A \rangle$ is not executed.

**Definition 4** *Idealness of execution sequence Let $S^W$ be the set of all possible execution sequences of a process model $W$, $r : A_1, \ldots, A_m \Rightarrow B_1 \otimes \cdots \otimes B_n$ be a control objective in FCL. Let $\Gamma$ denote the sequence of $A_1, \ldots, A_m$:*

- *A sequence $s \in S^W$ is* ideal *with respect to $r$ iff if $\Gamma$ is a subsequence of $s$, then $\Gamma; B_1^+; \neg B_2^+; \ldots; \neg B_i^+$ is a subsequence of $s$. With $\neg B_k^+$ we denote 0 or 1 occurrence of $\neg B_k$ in a sequence of tasks.*
- *A sequence $s \in S^W$ is a* sub-ideal *execution sequence with respect to $r$ iff if $\Gamma$ is a subsequence of $s$ then $\Gamma; B_j; \ldots; B_i$, $1 \leq j < i \leq n$, is a subsequence of $s$.*
- *A sequence $s \in S^W$ is "perfectly"* sub-ideal *with respect to $r$ iff if $\Gamma$ is a subsequence of $s$ and $\exists B_i$, $1 < i \leq n$ such that $\forall B_j$, $j < i$, $\Gamma; \neg B_1^+; \ldots; \neg B_j^+; B_i^+$ is a subsequence of $s$.*
- *A sequence $s \in S^W$ is a* non-ideal *execution sequence with respect to $r$ iff $\Gamma$ is a subsequence of $s$ and $s$ is neither* ideal *nor* sub-ideal.
- *A sequence $s \in S^W$ is* irrelevant *with respect to $r$ iff it is neither* ideal *nor* sub-ideal*, nor* non-ideal.

Given a control rule $r$, we denote the set of *ideal*, *sub-ideal* and *non-ideal* execution sequences as $S^r_{ideal}$, $S^r_{sub\text{-}ideal}$ and $S^r_{non\text{-}ideal}$ respectively. Table 2 shows the respective sequences for control rules $r_1$–$r_4$.

# 4 Measurement of Compliance Distance

We propose to use the notion of compliance degree as a quantitative measurement for assessing the distance between control objectives and business work practice as depicted by the relevant process. The overall measure of compliance is determined firstly, by measuring the compliance distance of the process model and secondly, by calibrating the measure though execution histories by taking into account frequencies of execution sequences. This step is essential since the execution history may identify differences in the incidence of certain execution sequences.

| Control Rule | $S^r_{ideal}$ | $S^r_{sub\text{-}ideal}$ | $S^r_{non\text{-}compliant}$ |
|---|---|---|---|
| $r_1 : A, B \Rightarrow C; F$ | $\langle A, B, C, F\rangle$ | | $\langle A, B\rangle$, $\langle A, B, C\rangle$, $\langle A, B, F\rangle$ |
| $r_2 : H \Rightarrow J \otimes I; K$ | $\langle H, J\rangle$ | $\langle H, I, K\rangle$, $\langle H, I, J\rangle$, $\langle H, J, I\rangle$, $\langle H, J, K\rangle$, $\langle H, I, J, K\rangle$ | $\langle H\rangle$, $\langle H, I\rangle$, $\langle H, K\rangle$ |
| $r_3 : K \Rightarrow L \otimes M$ | $\langle K, L\rangle$ | $\langle K, M\rangle$, $\langle K, L, M\rangle$ | $\langle K\rangle$ |
| $r_4 : J \Rightarrow L \otimes M$ | $\langle J, L\rangle$ | $\langle J, M\rangle$, $\langle J, L, M\rangle$ | $\langle J\rangle$ |

Table 2: State of idealness of control rules $r_1$–$r_4$

## 4.1 Compliance Distance on Process Model

Compliance distance is measured to indicate the degree of match between a design time process model and a set of control rules. The concept of *support* is utilized (van der Aalst, Alves de Medeiros, & Weijters, 2006) to define compliance distance. Given a set of execution sequences $S$ and a process model $W$, the *support* of $W$ based on a sequence $s \in S$, is given by the proportion of tasks in $s$ that can be executed in $W$. The range of support is a real number between 0 and 1, where 0 indicates no support ($s$ is not executable in $W$ at all) and 1 complete match The entire sequence $s$ can be executed in $W$, i.e., it is possible to derive an execution sequence $s^w$ from $W$ such that $s = s^w$. The *support* of $W$ based on $S$ is the weighted sum of I from all sequences in $S$, which is also between 0 and 1.

The rationale of this technique is to quantitatively measure how a process model $W$ represents the *ideal*, *sub-ideal* and *non-ideal* situations in control rule $r$ by calculating the support for $W$ against the set of *ideal* and *sub-ideal* execution sequences representing $r$. We refer to the support for *ideal* and *sub-ideal* sequences as *ideal* and *sub-ideal compliance degree* respectively. The first measurement indicates whether the *ideal* situation (the exact sequence) can be fully or partially supported in $W$ (*ideal compliance degree* $= 1$, or between $[0, 1]$) respectively). Similarly, the latter measurement indicates whether $W$ allows *sub-ideal* situation(s) and by what degree.

We begin the procedure by first extracting a sub-process from the process model which contains only the relevant tasks as found in the set of *ideal* and *sub-ideal* execution sequences of a given rule $r$. To achieve this we use a technique called SELECTIVE_REDUCE (Lu, & Sadiq, 2006). For example, the procurement process model $W$ (*cf.* Figure 1) is reduced into $W_1$, $W_2$, $W_3$ and $W_4$ (Figure 2) against control rule $r_1$, $r_2$, $r_3$ and $r_4$ respectively.

The *ideal* and *sub-ideal* compliance degree can now be calculated through

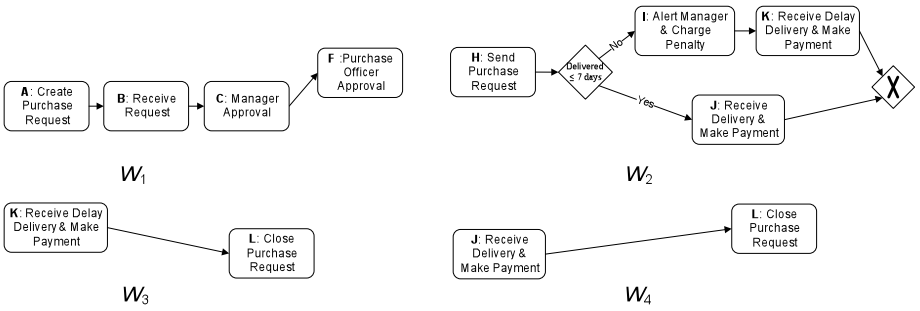


Figure 2: Sub-processes of the procurement process relevant to control rules $r_1$–$r_4$

the approach presented in Figure 3. The algorithm takes as inputs a process model $W$, a set of sequences $S$, and the control rule $r$, and produces the compliance degree. Functions $Trigger$, $Disable$ and $Initial$ given in Definition 2 are utilized. An additional function $SubInitial(W, r)$ returns the set of task node(s) which are immediately after the last task in the antecedent of $r$. For example, $SubInitial(W_2, r_2) = \{I, J\}$, where $H$ is the last task in the antecedent of $r_2$.

For each sequence $s$ in $S$, $Tr$ is initially given the first task in $W$ (step 4). For each task $T_i$ in a sequence $s$ (in this case, $T_i = s_i$ where $s_i$ is the $i$-th element in $s$), $Tr$ is the current set of triggered tasks as the result of executing task $T_i$ in $W$. Step 6 checks whether the triggered tasks in $Tr$ includes $T_i$. Step 9 calculates the proportion of tasks in $W$ triggered by tasks in $s$. After all relevant sequences in $S$ have been accounted for, the final compliance degree is scaled according to the total number of sequences in $S$ and returned (step 10). The algorithm complexity is bound by the number of tasks in the sequence and the number of different sequences in $S$.

For example, to compute the ideal compliance degree of $W$ with regard to $r_1 : A, B \Rightarrow C; F$, we input $W_1$, the sub-process of $W$ relevant to $r_1$ (*cf.* Figure 2), and $S^{r_1}_{ideal}$, the set of *ideal* execution sequences of $r_1$, where $S^{r_1}_{ideal} = \{\langle A, B, C, F\rangle\}$. Since there is only one sequence in $S^{r_1}_{ideal}$, the ideal compliance degree is $(1 + 1)/2 = 1$ (step 9), because $\langle C, F\rangle$ is an exact execution sequence following $\langle A, B\rangle$ in $W_1$.

To compute *sub-ideal compliance degree* of $W$ with regard to $r_2$, we input $W2$ and $S^{r_2}_{sub\text{-}optimal}$, the set of *sub-ideal* execution sequences of $r_2$, where $S^{r_2}_{sub\text{-}optimal} = \{\langle H, I, K\rangle, \langle H, I, J\rangle, \langle H, J, I\rangle, \langle H, J, K\rangle, \langle H, I, J, K\rangle\}$. For each sequence $s$ in $S^{r_2}_{sub\text{-}optimal}$, we display in Table 3 the intermediate result of *degree*, which is the support of $W_2$ received from $s$. Sequence $\langle H, I, K\rangle$ has degree of 1 since $\langle I, K\rangle$ is an exact sequence executable in $W_2$ given $\langle H\rangle$. Sequence $\langle H, J, I\rangle$ has *degree* of 0.50 because after executing $H$ in $W2$, $I$ and $J$ are enabled; and after executing $I$, $J$ is disabled and $K$ enabled, which gives $(1+0)/2 = 0.50$ (step 9). Similarly, sequence $\langle H, I, J, K\rangle$ has degree of 0.67 ($(1 + 0 + 1)/3 = 0.67$). The overall *sub-ideal compliance degree* is 0.63, which is the average degree of

**Procedure** COMPLIANCE_DEGREE

---

**Input** $W$, $S$, $r$
**Output** *degree*
1. *degree*, *count*, *comp* $\leftarrow 0$
2. For each different sequence $s$ in $S^W$
3. $s \leftarrow s - \Gamma$ /* truncate $s$ to contain consequence tasks of $r$ only */
4. $Tr \leftarrow SubInitial(W, r)$
5. For each task in $s$ denoted by $T_i$, $i \leftarrow 1, \ldots, |s|$
6. If $T_i \in Tr$ /* check if $T_i$ is triggered in $W$ */
7. $count = count + 1$
8. $Tr \leftarrow (Tr - \{T_i\} - Disable(W, T_i) \cup Trigger(W, T_i)$
9. $comp \leftarrow comp + \dfrac{count}{|s|}$
10. **Return** $degree \leftarrow \dfrac{comp}{|S|}$ /* calculate overall compliance distance and return */

Figure 3: Computing compliance degree

| $S^{r_2}_{sub\text{-}ideal}$ | *degree* |
|---|---|
| $\langle H, I, K \rangle$ | 1 |
| $\langle H, I, J \rangle$ | 0.50 |
| $\langle H, J, I \rangle$ | 0.50 |
| $\langle H, J, K \rangle$ | 0.50 |
| $\langle H, I, J, K \rangle$ | 0.67 |
| *sub-ideal compliance degree* | **0.63** |

Table 3: Intermediate result for applying COMPLIANCE_DEGREE to $S^{r_2}_{sub\text{-}optimal}$ and $W_2$

| Control Rules | Ideal Compliance | Sub-Ideal Compliance |
|---|---|---|
| $r_1$ | 1 | 0 |
| $r_2$ | 1 | 0.63 |
| $r_3$ | 1 | 0.25 |
| $r_4$ | 1 | 0.25 |
| TOTAL | **1** | **0.28** |

Table 4: Compliance measurement for process model $W$

the five *sub-ideal* sequences.

Table 4 lists the *ideal* and *sub-ideal* compliance degree for control rules $r_1$–$r_4$ respectively. The overall compliance degree is the sum of the compliance degree of each control rule. The results show that W is compliant with all ideal situations according to control rules $r_1$–$r_4$, and $W$ supports *sub-ideal* situations to some extent.

We use the *ideal compliance degree* to evaluate how well the process model supports a given control rule. $degree = 1$ indicates all ideal situation(s) of the control objective are represented in the process model $W$, (i.e., it is possible to find out the exact ideal execution sequence(s) in the relevant sub-process of $W$, hence the process is an *ideal design* for the control rule $r$). While $degree = 0$ indicates none of the ideal situation(s) is represented in $W$, from which we can immediately conclude that $W$ is non-ideal with $r$. If none of the task in any sequence of *ideal* or *sub-ideal* execution sequences $S^r_{ideal}$ is presented in the process model $W$, then one can only derive an empty sub-process from $W$. Thus the algorithm returns 0 in this case, which is corresponding to a *non-compliant* situation. Lastly, having a number between 0 and 1 indicates $W$ represents part of some *ideal* situation (i.e., it is possible to find some partial ideal execution sequence(s) in the relevant sub-process of $W$).

In addition, from the *sub-ideal degree* we can find out whether the process model may contain some *sub-ideal* situations. *Sub-ideal* will always be less than *ideal*, and always more than *non-ideal*. Although there are many interpretations, here we consider *sub-ideal compliance degree* as an auxiliary measurement to examine the expressiveness of the process model, in terms of expressing both *ideal* and *sub-ideal* executions.

## 4.2 Calibration with Execution History

Calculation of compliance distance of control rules from a process model provides only part of the measurement. The measurement should be further calibrated with respect to actual execution sequences found in process execution logs.

The procedure to measure the overall compliance of the process to the set of control rules, is undertaken by calculating the percentage of relevant execution sequences among all execution sequences in the log. This number indicates the relevance of the process model to the considered control rules. Suppose a set

| $\mathbf{s_i}$ | | ***count*** |
|---|---|---|
| $s_1$ | $\langle A, B, C, D, E, F, L\rangle$ | 40 |
| $s_2$ | $\langle A, B, C, D, E, F, H, J\rangle$ | 5 |
| $s_3$ | $\langle A, B, C, D, E, F, H, J, L\rangle$ | 25 |
| $s_4$ | $\langle A, B, C, E, D, F, H, I, K, L\rangle$ | 15 |
| $s_5$ | $\langle A, B, C, G, L\rangle$ | 5 |
| $s_6$ | $\langle A, B, C, G, H, J, L\rangle$ | 10 |
| $s_7$ | $\langle A, B, C, G, H, I, K, L\rangle$ | 20 |
| TOTAL | | **120** |

Table 5: The list of all execution sequences $S$ and their counters. In this case $S$ contains 7 relevant execution sequences, from 120 process instances

| **Type** | $S^{r_1}$ | **count** | $S^{r_2}$ | **count** | $S^{r_3}$ | **count** | $S^{r_4}$ | **count** |
|---|---|---|---|---|---|---|---|---|
| *ideal* | $\langle A, B, C, F\rangle$ | 85 | $\langle H, J\rangle$ | 40 | $\langle K, L\rangle$ | 35 | $\langle J, L\rangle$ | 25 |
| *sub-ideal* | | | $\langle H, I, K\rangle$ | 35 | | | | |
| *non-ideal* | $\langle A, B, C\rangle$ | 35 | | | | | $\langle J\rangle$ | 5 |
| TOTAL | | **120** | | **75** | | **35** | | **30** |

Table 6: The list of relevant execution sequences $S^r$ with respect to control rules $r_1$–$r_4$ and their counter

of execution sequences $S^W$ for a given process model $W$ can be extracted from an execution log ($S^W$ can be denoted by $S$ if there is only one model $W$). We group by each different execution sequence $s_i^W \in S$ and aggregate the number of its occurrence, which forms $\Delta$. Table 5 shows an example of $\Delta$ for the process models $W$ in Figure 1.

In order to examine the compliance degree, for each control rule $r_i$, we need to first extract the relevant sequences, including *ideal*, *sub-ideal* and *non-ideal* sequences from $\Delta$. An execution sequence $s_i$ in $\Delta$ is relevant to $r_i$, iff $s_i$ is a subsequence of some $s \in S^r$, where $S^r$ is the set of idealness sequence of $r_i$. The relevant sequences and the frequency of their occurrence in $\Delta$ is denoted by $\Delta^{r_i}$. Table 6 shows $\Delta^{r_i}$ for each control rules $r_1$–$r_4$ after filtering the list of relevant sequences and their counters. For example, $s_1$–$s_8$ are relevant to $r_1$, while $s_1$–$s_4$ contains the *ideal* sequence $\langle A, B, C, F\rangle$ and $s_5$-–$s_7$ contains the *non-ideal* sequence $\langle A, B, C\rangle$. It can be concluded that $\Delta$ is 100% relevant to $r_1$–$r_4$, since every executions sequence in $\Delta$ is relevant to at least one $r_i$.

The result indicates that there exist non-ideal situations during execution with respect to $r_1$ and $r_4$. In particular, $r_1$ requires that each purchase request must be approved by both manager (task $C$) and purchase officer (task $F$). From $\Delta^{r_1}$ it can be noticed that among total 120 process instances, for 85 times (71%) the ideal sequence $\langle A, B, C, F\rangle$ has occured. While the non-ideal sequence $\langle A, B, C\rangle$ has occurred 35 times. It can be checked that the process model (*cf.* Figure 1) allows for urgent cases to bypass purchase officer approval (task $F$), which leads to the non-ideal situations. Furthermore, $\Delta^{r_2}$ shows that there are

only 75 occurrences of sent purchase requests among 120 process instances. This is because there are 45 instances of declined purchase request (i.e., $count(s_1+s_5)$ in Table 6). For the case of $r_4$, it requires to perform an additional check if purchase request is not closed within 28 days. If so, the manager should be alerted. The additional task $M : AlertManager\&CloseRequest$ is not included in the process model, which led to 5 cases in $\Delta^{r_1}$ where delivery has been received but purchase request was not closed in time.

The data produced in Table 6 can be further used as weights to calibrate the compliance measurement of the process model as given in Table 4. For example, although the support for *ideal* sequence is 1, the *ideal* sequence for $r_2$ is found in only $(85/120) = 71\%$ of instances.

## 4.3 Towards Achieving Compliance

The measurement of compliance distance provides a quantitative means of analyzing process models as well as execution behavior (actual work practice) with respect to a given set of control rules representing the relevant compliance requirements. Using these measurements, process owners can make an informed decision on what changes (if any) they would like to make to their processes.

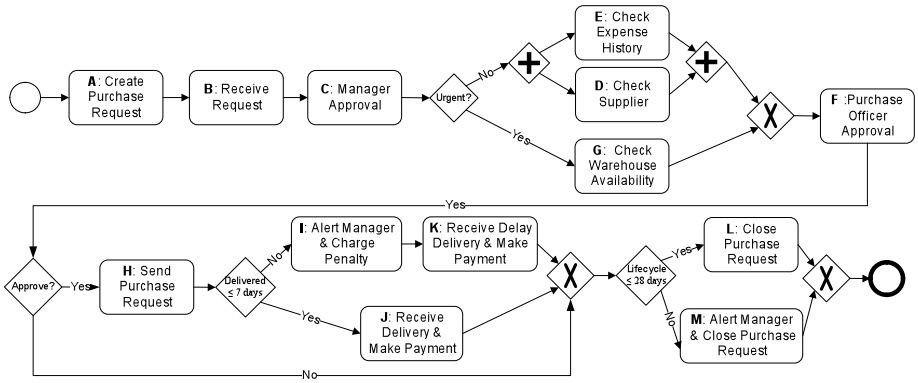


Figure 4: Revised process model

A major contributing factor towards the decision to change the affected processes is the risk associated with non-compliance. Risk of non-compliance is one of many risks that an organization faces which include fraud, malicious damage, theft, equipment failure etc. Effective risk management requires a clear understanding of the organization's risk appetite. Risk analysis techniques are highly advanced and not the focus of this paper. However, what decision a process owner will take after an analysis of compliance distance will depend on such an analysis. For example the process owner may decide not to deploy the process anymore (*avoid risk*), or define and implement requisite controls (*mitigate risk*), or take additional measures to transfer risk e.g. insurance (*share risk*) etc.

As an example, consider the scenario that by analyzing the runtime compliance reflected in $\Delta$, the original process model is modified to improve the compliance degree. Figure 4 shows a possible design for the improved process model where *non-ideal* situations with regard to $r_1$ and $r_4$ are mitigated.

Subsequently, simulation techniques (e.g. ASIM tool from IDS Scheer (www.ids-scheer.com)) where available, can be used to further analyze the modified process. Note that the method for measurement of compliance distance can be utilized in this regard and also applied to simulated execution data.

# 5 Related Work

Governance, risk and compliance (GRC) is an emerging area of research which holds challenges for various communities including information systems, business software development, legal, cultural, & behavioral studies and corporate governance.

In this paper, we focus on compliance management from an information systems perspective, in particular the modeling and analysis of compliance requirements. Both process modeling as well as modeling of normative requirements are well studied fields independently, but until recently the interactions between the two have been largely ignored (Desai, Mallya, Chopra, & Singh, 2005), (Padmanabhan, Governatori, Sadiq, Colomb, & Rotolo, 2006).

It is obvious that the modelling of control objectives will be undertaken as rules, although the question of appropriate formalism is still under studied. A plethora of proposals exist both in the research community on formal modelling of rules, as well as in the commercial arena through business rule management systems (see e.g. ilog.com). We have proposed FCL as a candidate which has proved effective due to its ability to reason with violations, but we acknowledge that further empirical study is necessary to effectively evaluate the appropriateness of FCL.

As discussed previously, the approach of this paper takes a preventative focus. In terms of detective approaches to compliance management, a wide range of supporting technologies can assist, which include several commercial solutions (business activity monitoring, business intelligence etc). Noteworthy in research literature is the synergy with process mining techniques which provide the capability to discover runtime process behavior (and deviations) and can thereby assist in detection of compliance violations (van der Aalst, van Dongen, Herbst, Maruster, Schimm, & Weijters, 2003), (van Dongen, de Medeiros, Verbeek, Weijters, & van der Aalst, 2005).

There have been recently some efforts towards support for business process modelling against compliance requirements. In particular, the work of (zur Muehlen & Rosemann, 2005) provides an appealing method for integrating risks in business processes. The proposed technique for "risk-aware" business process models is developed for EPCs (Event Process Chains) using an extended notation. (Goedertier, & Vanthienen, 2006) presents a logical language PENELOPE, that provides the ability to verify temporal constraints arising from

compliance requirements on effected business processes, and (Kuster, Ryndina, & Gall, 2007) provide a method to check compliance between object lifecycles that provide reference models for data artefacts e.g. insurance claims and business process models. On a similar note, (Giblin, Muller, & Pfitzmann, 2006) provide temporal rule patterns for regulatory policies, although the objective of this work is to facilitate event monitoring rather than the usage of the patterns for support of design time activities. Furthermore, (Agrawal, Johnson, Kiernan, & Leymann, 2006) has presented a workflow architecture for supporting Sarbanes-Oxley Internal Controls, which include functions such as workflow modeling, active enforcement, workflow auditing, as well as anomaly detection.

# 6 Conclusion

As the importance of governance, risk and compliance grows for various industries, there is an evident need to provide supporting tools and methods to enable organizations seeking corporate social responsibility to achieve their objectives. The challenges that reside in this topic warrant systematic approaches that motivate and empower business users to achieve a high degree of compliance with regulations, standards, and corporate policies.

The contribution of this paper has been two fold — firstly to present an overall methodology for utilizing business process platforms as a vehicle for compliance, and secondly to provide a method to quantitatively measure the distance of control rules from existing work practice through analysis of affected process models and their underlying execution behaviour. Thereby empowering process owners with the capability to make informed decisions when dealing with compliance obligations.

The reliance of the proposed method on FCL ensures that compliance rules have a representation which is evidently conceptually faithful to the complexities inherent in the normative nature of compliance requirements, thus ensuring a rich environment for expressing these requirements. The advanced logical formalism provided by FCL overcomes a number of limitations in previous work such as the ability to deal with reparations, and ability to reason with norm changes (Governatori et al., 2006). At the same time, the proposed method provides a computationally efficient means of analysing the relationship between the process models and the entire spectrum of compliance rules. Although previous approaches have provided methods for specific aspects of compliance rules (e.g. (Goedertier et al., 2006), (Liu, Muller, & Xu, 2007)), the work presented in this paper is covers a much broader scope of consideration, while maintaining the efficiency of the associated computations (i.e. generation of execution sequences and calculation of compliance degree).

The work has many interesting extensions in terms of advanced analysis techniques for business process models with respect to the control objectives. In our future work, we hope to develop a more precise taxonomy for interpretation of the sub-ideal compliance degree, and subsequently utilize it for providing concrete error diagnostics, that is providing a means of understanding what

needs to be done in order to achieve (a higher degree of) compliance. This is indeed a challenging task given the diverse nature of sub-ideal semantics. However, targeting this aspect will create a more holistic approach to compliance management, by not only providing preventative and detective techniques, but also corrective recommendations.

# References


van der Aalst, W. M. P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., & Weijters, A.J.M.M. (2003). Workflow Mining: A Survey of Issues and Approaches. *Data & Knowledge Engineering*, 47, 237–267.

van der Aalst, W. M. P., Alves de Medeiros, A. K., Weijters, A. J. M. M. (2006). Process Equivalence: Comparing Two Process Models Based on Observed Behavior. In *Proceedings of the 4th International Conference on Business Process Management*, pp. 129–144, Vienna, Austria 2007. Springer-Verlag.

van Dongen, B.F., de Medeiros, A.K.A., Verbeek, H.M.W., Weijters, A.J.M.M., & van der Aalst, W.M.P. (2005). The ProM Framework: A New Era in Process Mining Tool Support. In *Proceedings of 26th International Conference Applications and Theory of Petri Nets*, pp 444–454, Miami, USA, 2005. Springer-Verlag.

zur Muehlen, M., & Rosemann, M. (2005). Integrating Risks in Business Process Models. In *Proceedings of 16th Australasian Conference on Information Systems*. Sydney, Australia.

Agrawal, R., Johnson, C., Kiernan, J., & Leymann, F. (2006). Taming Compliance with Sarbanes-Oxley Internal Controls Using Database Technology. In *Proceedings of the 22nd International Conference on data Engineering*, 2006, Atlanta, GA, USA. IEEE Computer Society.

BPM Forum (2006). CEE: The Future. Building the Compliance Enabled Enterprise. Report produced by Global Fluency in partnership with: AXS-One, *Chief Executive Magazine and IT Compliance Institute*.

Desai, N., Mallya, A.U., Chopra, A.K., & Singh, M.P. (2005). Interaction Protocols as Design Abstractions for Business Processes. IEEE *Transaction on Software Engineering* 31(12), 1015–1027.

Giblin, C., Muller, S., Pfitzmann, B. (2006). *From regulatory policies to event monitoring rules: Towards model driven compliance automation*. IBM Research Report. Zurich Research Laboratory.

Goedertier, S., & Vanthienen, J. (2006). Designing Compliant Business Processes with Obligations and Permissions. In Eder, J., & Dustdar, S. et al. (Eds.) *Proceedings of Workshop on Business Process Design*, pp. 5–14, Vienna, Austria 2006. LNCS 4103 Springer-Verlag.

Governatori G. & Milosevic, Z. (2006) A Formal Analysis of a Business Contract Language. *International Journal of Cooperative Information Systems* 15(4), 659–685.

Governatori G. Milosevic, Z, & Sadiq, S. (2006). Compliance checking between business processes and business contracts. In *Proceedings of the 10th IEEE Conference on Enterprise Distributed Object Computing*, Hong Kong.

Hagerty, J. (2006). *SOX Spending for 2006*. AMR Research, Boston USA.

Kuster, J., Ryndina, K., & Gall, H., (2007). Generation of Business Process Models for Object Life Cycle. In *Proceedings of the 5th International Conference on Business Process Management*, pp. 165–180, Brisbane, Australia. Springer-Verlag.

Liu, Y., Muller, S., & Xu, K. (2007). A static compliance checking framework for business process models. *IBM Systems Journal*, 46(2), 2007.

Lu, R., Sadiq, S. (2006). Managing Process Variants as an Information Resource. In *Proceedings of International Conference on Business Process Management*, Vienna, Austria, 2006. Springer-Verlag.

Padmanabhan, V., Governatori, G., Sadiq, S., Colomb, R., & Rotolo, A. (2006). Process Modeling: The Deontic Way. In M. Stumptner, S. Hartmann and Y. Kiyoki, editors, *Australia-Pacific Conference on Conceptual Modeling*, pp. 75–84, CRPIT 53.

Sadiq, W. (2002). *On Verification Issues in Conceptual Modeling of Workflow Processes*. PhD Thesis, The University of Queensland.

Sadiq, S., Governatori, G., & Naimiri, K. (2007) Modeling Control Objectives for Business Process Compliance. In *Proceedings of the 5th International Conference on Business Process Management*, pp. 149–164, Brisbane, Australia 2007. Springer-Verlag.

Sadiq, W., Orlowska, M. (2000). Analyzing Process Models using Graph Reduction Techniques. *Information Systems*, 25(2), 117–134.