Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management
Bali, Indonesia, January 7 – 9, 2014

Real Time Process Compliance Checking using Nomenclature Approach

Murphy Choy, Ma Nang Laik and Koo Ping Shung
School of Information System
Singapore Management University
80 Stamford Road
Singapore 178902

Abstract

Process compliance has experienced rapid development in the wake of several serial scandals in both financial institutions such as RBS and Société Générale as well as non-financial institution such as Enron and Olympus. These scandals resulted in legislation enacted specifically to address these issues and prompted the businesses to develop more systematic approaches to the design of business processes which adhere to the reality as well as satisfying the compliance requirements. There are several researches into the active use of logical search approaches in checking whether the process meets the requirements. In general, most of the research focuses on the extension of the logical pattern recognition. However, most of the compliance research focuses only on the examination of business process flow during the design phase and not operation. In this paper, the authors will like to propose the novel use of Nomenclature approach to evaluate business processes as they occur. We also illustrate using a real-world case study on how this new approach also facilitate the checking of compliance in real time.

Keywords
Compliance, Process Analysis, Deontic Logic, Nomenclature, Business Process Extraction

Introduction

Modern businesses face a huge number of challenges to maintain their competitive edge. Companies must achieve the expectations set by the share-holders and remain profitable while maintaining positive customer relations. With globalization and growth of digit medias, companies are confronted with dynamic competition conditions. This resulted in large investments in information technology which is necessary to stay competitive and remain in business. To further enhance their competitiveness, many enterprises have demonstrated interest in business process management. Most businesses wished to model, automate, optimize and monitor their businesses processes to improve their customer satisfaction, increase revenue or reduce the operational inefficiency. In the recent years, there have been an increased acceptance and adoption of business process management systems.

The development has been further fuelled by the growing number of regulatory requirements imposed on business operations as a result of corporate scandals. The most prominent legislations are the Gramm-Leach-Bliley Act (US, 1999) and the Sarbanes Oxley Act (SOX) (US, 2002). While these regulations have coverage across industries, there are also industry-specific regulations such as the Basel III accord (Basel, 2011) or the European Money Laundering Regulation (UK, 2003). Demonstration of compliance with specific legal legislations and international standards requires the company to document their existing business processes as well as verifying that the processes conform to legislation. Most enterprises regard such documentation requirements and strict adherence to be a business cost, they also recognize the opportunity to identify and document their formal and informal processes to render their executions more efficiently and effectively. For huge enterprises with several divisions and thousands of different business processes, this is a major challenge both in documentation as well as process monitoring.

Most enterprises that operate in heavily regulated industries, such as financial services or health care, are controlled by a huge number of regulatory requirements that define most of their operations. As these implemented requirements needed to be enforced by a multitude of internal business and IT controls, several regulations recommend the use of industry standards, such as COBIT (Control Objectives for Information and Related Technologies) (COBIT, 2005) and ITIL (Information Technology Information Library) (ITIL, 2006), in the implementation of any enterprise IT system. These standards consist of well-defined abstract process definitions which can be tailored according to individual needs.

With increasing number of legislations and standards, companies needed a comprehensive compliance-management approach to manage the design of new processes as well as ensuring the conformance of process. The need to be able to understand the impact of new regulations for their business and processes becomes ever more important. As business processes are increasingly managed using Business Process Management Systems (BPMS), any regulatory requirements that require changes to the structure of any workflows will directly impact business process modeling.

Whenever there is new regulatory requirement, a company will need to understand the impact and react accordingly. There are three possible outcomes,

1. Adaptation or removal of business processes.
2. Creation of new business processes.
3. No change is needed.

Business processes that are automated through BPMS can implement IT processes and controls, as stipulated by ITIL or COBIT and address existing legislation requirements. However new regulatory requirements cannot be readily or easily assessed with these frameworks. For large companies with thousands of business processes deployed on the BPMS, the ability to quickly assess the compliance of existing process definitions is extremely important. In this paper, we describe a nomenclature approach that caters to static verification of business process models against a predefined set of regulatory requirements such as execution order of certain tasks. The nomenclature approach also performs historical process checking as well as real time checking. Our approach will assist the business to identify non-compliant business processes before implementation and identify non-compliance business process during operations.

Importance of automated verification of business processes

Before compliance can be enforced, the effect of every new legislation on existing business processes needs to be identified. Our approach identifies the non-compliance processes at the relevant stages which provides a useful tool to ensure that new requirements are adapted into the processes. Hence, the contribution of our method is three folds:

1. With automated verification of business processes during design phase and operation phase, our approach increases efficiency of detection of non-compliant processes and lower the risk of the event of non-compliance. (Abrahms et. al., 2007; Giblin et. al., 2005; Ly et. al., 2011; Governatori and Rotolo, 2010)
2. Through automation of a manually tedious task, our method also reduces the cost of inspecting business process models for compliance especially during operation.
3. The real time updating of the process compliance allows management to have a better and more in-depth understanding of the problems that the business faces.

Nomenclature Approach

Nomenclature is defined as a systematic approach to the naming of items in the area of science or arts by individuals or community. It can also refer to the systematic naming of items according to taxonomy. In most scientific disciplines, there are established standards of nomenclature. Nomenclatures are used predominantly in Mathematics, Biology as well as Chemistry. The use of nomenclatures in taxonomy allows for easy tracing of the origin of the species. Nomenclatures also allow for easy identification of similar components between two chemicals or species.

In business process management, most of the business processes were mapped out in BPMS software in a logical decision flow manner such as the use of BPELs (Business Process Extraction Language) (Antoniou et. al., 2005; Ghose and Koliadis, 2007), BPCL (Business Process Compliance Language) or BPSL (Business Property Specification Language) . These models are extremely useful in understanding whether the flow is sensible and what are the components of the process. Other software uses symbolic logic such as Standard Deontic Logic (Alberti, 2004; Alberti, 2005) to assist in the checking of the process.

In this paper, we will discuss about the modification of a process flow with the addition of elements of nomenclature approaches from other area of sciences. Below we have several simple processes,

$$a_1 \Rightarrow a_2$$

$$a_1 \Rightarrow a_2 \Rightarrow a_1 \Rightarrow a_2 \Rightarrow a_1 \Rightarrow a_2$$

$$a_1 \Rightarrow a_2 \Rightarrow a_1 \Rightarrow a_2 \Rightarrow a_1 \Rightarrow a_2 \Rightarrow a_1 \Rightarrow a_2$$

where both a_1 and a_2 are two separate processes which are linked to one another and they can be repeated sequentially. In most cases, these processes will be viewed as separate processes. However, this is inappropriate as they are essentially the same process repeated multiple number of times. Under normal circumstances, the common compliance software will have to validate three separate processes. By simplifying these processes which are repeated for multiple times, we can express them in summarized form below,

$$[a_1 \Rightarrow a_2][1]$$

$$[a_1 \Rightarrow a_2][3]$$

$$[a_1 \Rightarrow a_2][4]$$

The process can be generalized in the form below which reduces the number of processes that needs to be checked to one.

$$[a_1 \Rightarrow a_2][m] \text{ Where} m \in \mathbb{Z} \text{and } m \geqslant 1$$

While we have discussed about the case for multiple events processes which are repetitive, the case for single event repetition has not been discussed. To enable one to distinguish a single event from repeated process, we will be using the notation as below.

$$\ldots a[n] \ldots$$

Algorithm for extraction and simplification

As described in the prior section, the search for repeated process and events needs to follow a simple logic to simplify the process. Below is the pseudo-code for algoithm for searching the process flow as well as simplifying the process.

Let us first assume the following general process

$$a_1 \Rightarrow a_2 \ldots a_{k-1} \Rightarrow a_k$$

Given the process has k events, the maximum length for repeatable substring within a longer string will be $\frac{(k-1)}{2}$ if k is an odd number and $\frac{k}{2}$when k is an even number (Lucian and Smyth, 2011), if the maximum theoretical possible length is not given by the process owner. Once the length is established, we will then initiate the search from this length to the base minimum length of 1 which is similar to the behaviour of suffix tree (Ukkonen, 1995). For each level of process search, we will also create a holder to indicate the start of the repeated process. This level of process will then set all lower levels to 0 so as to prevent multiple level simplifications. At the same time, we have to set the constraint where all the events in the process are not the same.

Translating the above information, the pseudo code form is as follow.

Let X_k be the event X at position k in a chain of events.
Let i be the number of events for a particular process.
Let j be the number of levels for a particular process (maximum length for repeatable subsring).
Let L_{kj} be the repeat flag at position k for level j in a chain of events.
Let h be the number of repetitions of a sub-process.

1. Set Max Length (j) as $\frac{(i-1)}{2}$if j is odd or $\frac{(i)}{2}$if j is even, otherwise j as given.
2. Create j levels of holder with length i.
    a. For level j till 1,
        i. For position k from 1 to i-j,
        ii. Ensure $X_k = X_{k+j}$
        iii. Repeat the check till $X_k = X_{k+j-1}$
        iv. If j > 1 then check for position k, $X_k \neq X_{k+1}$
        v. If so, $L_{k,j} = 1$ else $L_{k,j} = 0$
3. For level j till 1,
    a. For position k from 1 to n-m+1,
        i. If $L_{k,j} = 1$ and $L_{k,j-1} = 1$, then $L_{k,j-1} = 0$
4. For level j till 1,
    a. For position k from 1 to n-m,
        i. Check whether for $L_{k,j} = 1$
        ii. If yes then do the following
            1. Repeat the check on $L_{k+j*h,j}$ until $L_{k+j*h,j} = 0$ where h indicates the number of repetitions
            2. Condense the process from position k to k+j*h with the repetitions denoted as h+1.
            3. Move the search to k + j*h +1 position.
5. The condensed process is then created from the iteration process.

Let us look at an example where i = 8 and j = (8)/2 = 4.

$$P01 \Rightarrow P02 \Rightarrow P01 \Rightarrow P02 \Rightarrow P03 \Rightarrow P05 \Rightarrow P06 \Rightarrow P06$$

Let us assume that the maximum length of any single process is 2. From the algorithm, we can evaluate the process and notice that components 1, 2 and 3 are the same. This reduces the level 2 process to this.

$$(L_{2,1} = 1) \Rightarrow (L_{2,2} = 1) \Rightarrow (L_{2,3} = 0) \Rightarrow (L_{2,4} = 0) \Rightarrow (L_{2,5} = 0) \Rightarrow (L_{2,6} = 0) \Rightarrow (L_{2,7} = 0) \Rightarrow (L_{2,8} = 0)$$

Repeating the process for level 1, from the algorithm, we can evaluate the process and notice that components 7 and 8 are the same. This reduces the level 1 process to this.

$$(L_{1,1} = 0) \Rightarrow (L_{1,2} = 0) \Rightarrow (L_{1,3} = 0) \Rightarrow (L_{1,4} = 0) \Rightarrow (L_{1,5} = 0) \Rightarrow (L_{1,6} = 0) \Rightarrow (L_{1,7} = 1) \Rightarrow (L_{1,8} = 0)$$

Once we have identified the levels, we can now proceed to do the clean up.

$$(L_{2,1} = 1) \Rightarrow (L_{2,2} = 1) \Rightarrow (L_{2,3} = 0) \Rightarrow (L_{2,4} = 0) \Rightarrow (L_{2,5} = 0) \Rightarrow (L_{2,6} = 0) \Rightarrow (L_{2,7} = 0) \Rightarrow (L_{2,8} = 0)$$

$$(L_{1,1} = 0) \Rightarrow (L_{1,2} = 0) \Rightarrow (L_{1,3} = 0) \Rightarrow (L_{1,4} = 0) \Rightarrow (L_{1,5} = 0) \Rightarrow (L_{1,6} = 0) \Rightarrow (L_{1,7} = 1) \Rightarrow (L_{1,8} = 0)$$

From the clean up processes, we start from level 2 repeated process and start condensing the process.

$$[P01 \Rightarrow P02][2] \Rightarrow P03 \Rightarrow P05 \Rightarrow P06 \Rightarrow P06$$

From the clean up processes, we start from level 1 repeated process and start condensing the process.

$$[P01 \Rightarrow P02][2] \Rightarrow P03 \Rightarrow P05 \Rightarrow P06[2]$$

In the next section, we will discuss about the application of this approach to a logistic company and how it assisted them in the discovery of processes which are non-compliant.

Case Study of Logistic Company

In the case of the logistic company A, they have several delivery services with many business processes controlling the operation of the company. Because of the nature of the business, certain legislative requirements force the company formulate their existing business process around those legislation. At the same time, because of the nature of the service rendered, there are contractual agreement on the service level making it important to analyze the business processes for any anomalous behaviours.

Currently, the business process is mapped out using the BPMS software and most of the booking of services are done electronically. The operation uses electronic devices to scan the delivery process and track the movement of items. The devices capture all the process status and update them to the system. However, there are no visibility to the compliance level of the business processes. The existing BPMS does not allow the company to check their existing processes.

Using the new nomenclature approach, we have mapped out the existing processes and converted them into general processes. To account for several possible repeated processes and sub-processes, we have generated several iteration of the different processes. Using the nomenclature forms, we then analyze their existing structures. Without using the generalized form, there are almost 30,222 different processes which needed to be mapped to the BPMS systems. This is tremendous amount of work and passing them through the system individually to test the logic flow is also a computational intensive work. Using the generalized nomenclature approach, we reduce the processes to 10,272 which

is around 65% reduction in the number of processes. To test the efficiency of the algorithms, we conducted a test of the run time of the algorithms for 6 business areas and estimate the efficiency improvement from deontic to nomenclature approach.

To evaluate the runtime efficiency of the new approach, we conducted simulation run of the algorithms and compared the runtime efficiency using paired sample t-test. The simulation first virtualizes two separate compliance checking systems on two separate machines. Both systems are linked to a central database that asynchronously passes the data to the compliance checking systems. The data base is fed with real life information at regular interval which simulates the real operations of such systems. The run time is measured by the amount of time needed to process the data to check for compliance after each update. The simulation is ran 100 times which simulate 100 days of operation for both systems and the paired sample t-test used to compare the differences in the runtime. Six business areas were selected which represent 90% of all the transactions. The six business areas are same day parcel, same day mail, normal mail, express mail, bulk parcel and letter deliveries.

The paired difference test is a type of location test used when comparing two sets of measurements to assess whether their population means differ. A paired difference test uses additional information about the sample that is not present in an ordinary unpaired testing situation, either to increase the statistical power, or to reduce the effects of confounders. Specific methods for carrying out paired difference tests are, for normally distributed differences the paired t-test where the population standard deviation of difference is not known.

Let R_i be the run time for deontic approach run i.
Let W_i be the run time for nomenclature approach run i.
Let μ_0 be the population mean to be tested against.
Let $\bar{X}_D$ be the sample difference mean.
Let S_D is the sample standard deviation of the sample
Let n be the sample size.

The t-test will test the null hypothesis (H_0) which assumes that population mean is equal to a specified value μ_0. The alternate hypothesis (H_1) assumes that population mean is not equal to a specified value μ_0.

$$H_0: \bar{X}_D = \mu_0$$
$$H_1: \bar{X}_D \neq \mu_0$$

Let X_i be the run time difference between deontic and nomenclature approach for run i.

$$X_i = R_i - W_i$$

Below is the calculation for $\bar{X}_D$.

$$\bar{X}_D = \frac{\sum_{i=1}^{n}(X_i)}{n}$$

Below is the calculation for S_D.

$$S_D = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X}_D)^2}{n}}$$

Below is the paired samples t-test statistics.

$$t = \frac{\bar{X}_D - \mu_0}{\frac{S_D}{\sqrt{(n)}}}$$

If the test statistics is significant and $\bar{X}_D$ is greater than 0, then nomenclature approach is superior. Otherwise the test statistics is significant and $\bar{X}_D$ is lesser than 0, then deontic approach is superior. Below are the various test results.

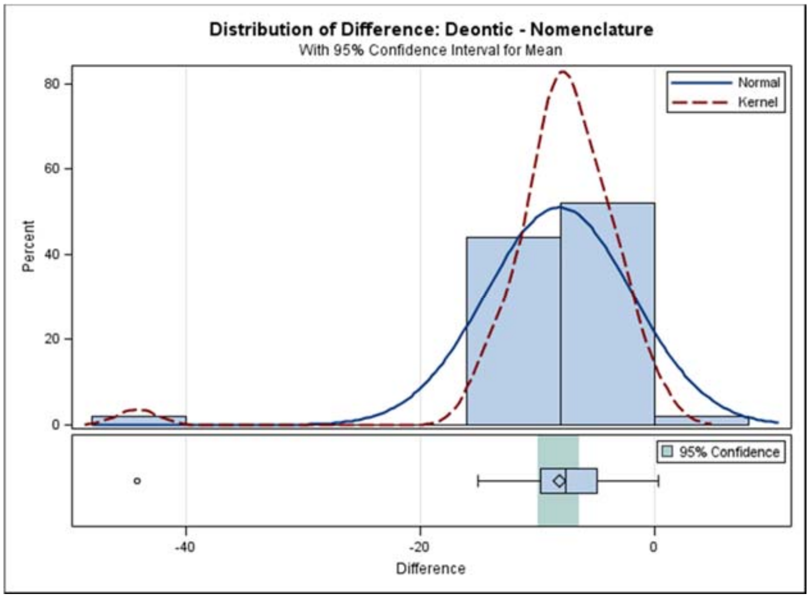

Figure 1: T-Test Results between Deontic and Nomenclature approaches for same day delivery

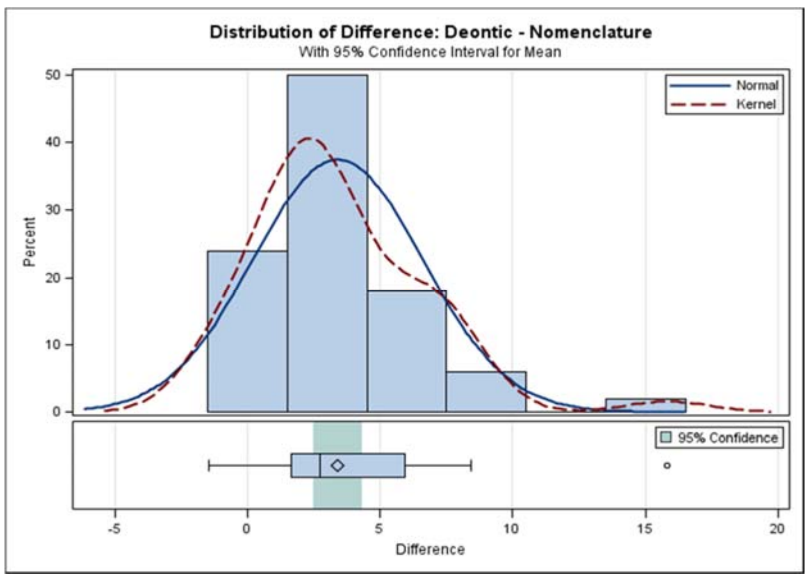

Figure 2: T-Test Results between Deontic and Nomenclature approaches for same day mail

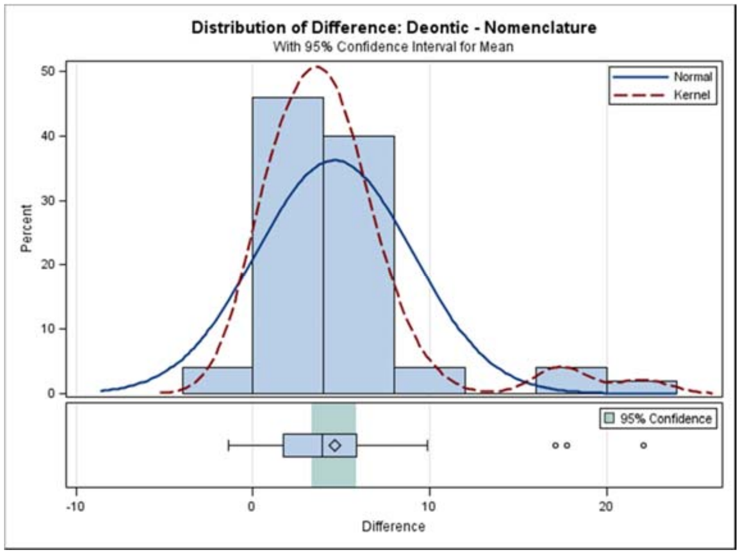

Figure 3: T-Test Results between Deontic and Nomenclature approaches for normal mail

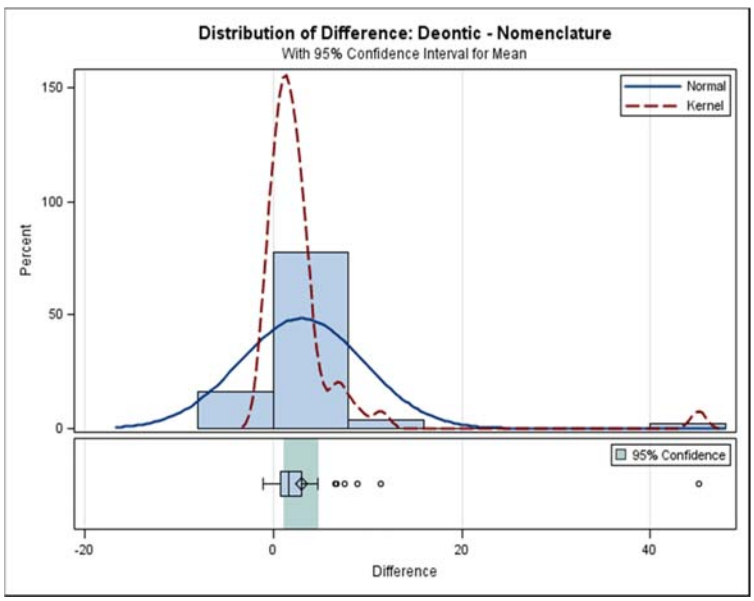

Figure 4: T-Test Results between Deontic and Nomenclature approaches for express mail

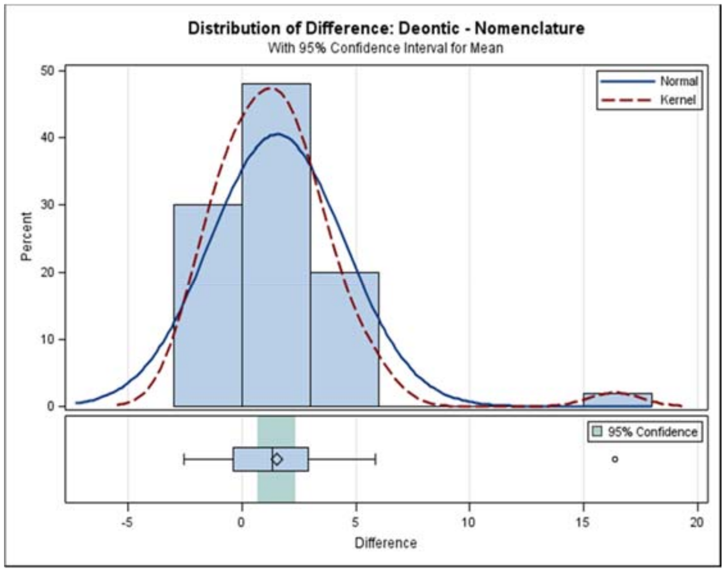

Figure 5: T-Test Results between Deontic and Nomenclature approaches for bulk parcel

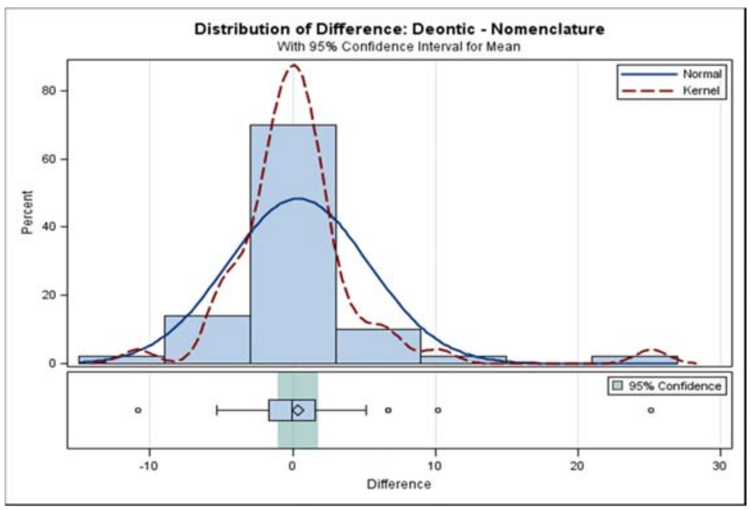

Figure 6: T-Test Results between Deontic and Nomenclature approaches for letter deliveries

From the results, four of six processes demonstrated that the new nomenclature approach is more efficient than the classical deontic approach. For the first case where nomenclature approach failed to outperform, the processes were in general very short and thus there are no significant amount of compression work.

Given the nature of the approach, the analysis can be scheduled to run for regular time intervals and the outputs compared to a predetermined list of possible processes for compliance checks. At the end of the daily operation, a report can be generated to calculate the compliancy rate.

Conclusion

The nomenclature approach offers improvement to the compliancy checking in the following area.

1. Innovative approach
2. Ease of logical interpretation
3. Detection of non-compliancy
4. Real time detection is possible
5. Improve productivity
6. Ensure compliance

The approach also allow flexibility to users who wants to further adapt it for their company usage. The ease of applying the compliancy check in operation also enables the management to have a tighter rein on compliancy as well as near real time report any compliancy failures. However, there are still some weaknesses in the existing approach. The approach currently do not deal with multiple level nested process as these processes are extremely rare. Reseting the process or transfer from one process to another cannot be dealt with using this approach. Future research direction requires more in depth improvement to the heuristic to identify and address these issues.

References

A. Ghose and G. Koliadis. Auditing business process compliance. ICSOC, LNCS 4749, pages 169–180, 2007. Basel III, Basel Committee on Banking Supervision (2011)

C. Abrams, J. von Kanel, S. Muller, B. Pfitzmann, and S. Ruschka-Taylor, ‘‘Optimized Enterprise Risk Management,’’ IBM Systems Journal 46, No. 2, 219–234 (2007).

C. Giblin, A. Y. Liu, S. Muller, B. Pfitzmann, and X. Zhou, ‘‘Regulations Expressed as Logical Models (REALM),’’ Proceedings of the 18th Annual Conference on Legal Knowledge and Information Systems, Brussels, Belgium (2005), pp. 37–48.

Control Objectives for Information and Related Technology (COBIT), Version 4.0 IT Governance Institute (2005), http://www.itgi.org.

F. Leymann and D. Roller, Production Workflow: Concepts and Techniques, Prentice Hall PTR, Upper Saddle River, NJ (2000).

G. Antoniou, D. Billington, G. Governatori, and M. J. Maher. Representation results for defeasible logic. ACM Trans. on Computational Logic, 2:255–287, 2001.

Governatori, Guido, and Antonino Rotolo. "A conceptually rich model of business process compliance." *Proceedings of the Seventh Asia-Pacific Conference on Conceptual Modelling-Volume 110*. Australian Computer Society, Inc., 2010.

Gramm-Leach-Bliley Act of 1999 (GLBA), Public Law 106- 102 (113 Statute 1338), United States Senate Committee on Banking, Housing, and Urban Affairs (1999).

IT Infrastructure Library (ITIL), Office of Government Commerce (2006), http://www.itil.co.uk.

Ilie, Lucian, and William F. Smyth. "Minimum unique substrings and maximum repeats." *Fundamenta Informaticae* 110.1 (2011): 183-195.

Ly, Linh Thao, et al. "Monitoring business process compliance using compliance rule graphs." *On the Move to Meaningful Internet Systems: OTM 2011*. Springer Berlin Heidelberg, 2011. 82-99.

M. Havey, Essential Business Process Modeling, O’Reilly & Associates, Sebastopol, CA (2005).

M. Alberti, D. Daolio, P. Torroni, M. Gavanelli, E. Lamma, and P. Mello. Specification and Verification of Agent Interaction Protocols in a Logic-based System. SAC’04, pages 72–78, 2004.

M. Alberti, M. Gavanelli, E. Lamma, P. Mello, P. Torroni, and G. Sartor. Mapping of Deontic Operators to Abductive Expectations. NORMAS, pages 126–136, 2005.

Sarbanes-Oxley Act of 2002, Public Law 107-204 (116 Statute 745), United States Senate and House of Representatives in Congress (2002).

The Money Laundering Regulations, Statutory Instrument 2003 No. 3075, Act of Parliament, http://www.opsi.gov.uk/si/si2003/20033075.htm.

Ukkonen, Esko. "On-line construction of suffix trees." *Algorithmica* 14.3 (1995): 249-260.

Biography

Murphy Choy is an instructor in Singapore Management University. He has extensive experience in the area of risk management and has developed credit risk models for several banks in various parts of the world. Murphy has also developed solutions and models for other industries such as logistics and retail. He is completing his doctorate and his research interest is in the area of Risk Management Operation management and Text Mining.

Nang Laik is a director of Master of IT in Business-Analytics (MITB-A) at School of Information Systems, Singapore Management University (SMU). She holds a PhD from Imperial College, London where her research focused on operations research (OR) in the area of optimization of resource. She teaches undergraduate core modules and master level courses in SMU. She has been teaching Business Process Modelling course and Computer as an Analysis Tool and Business Analytics Practicum. Her research expertise lies in the simulation and modelling of large scale real-world problems and the development of computationally efficient algorithms to enable sound and intelligent decision making in the organization. Nang Laik is an expert in transportation and logistics industry, she serves as a consultant for one of the best airports - Changi Airport Group to use data and decision Analytics to generate insights, make better decision and improve the business efficiency and productivity.

Koo Ping Shung is an experienced practitioner of analytics spanning numerous industries and business functional areas such as Marketing Management, Risk Management, Strategic Management and Human Resource Management. He specializes in the business adoption and implementation of analytics and is acknowledged by fellow practitioners as well-read in the field of business analytics. He is a trainer for SAS, a business analytic solution provider, in SAS-related subject matters. He has excellent command of the SAS programming language with deep understanding of the functionality of the base SAS software. His career have taken him to two Singapore Universities, National Institute of Education and Singapore Management University and two banks in Singapore, DBS Bank and OCBC Bank.