Compliance checking of shipment request by utilizing process mining concepts: An evaluation of Smart Auditing Framework

Faiza Allah Bukhsh
Department of Computer Science,
University of Twente, Enschede, the Netherlands
Email: f.a.bukhsh@utwente.nl

Hans Weigand
Department of Management Sciences,
Tilburg University, Tilburg, the Netherlands
Email: h.weigand.uvt.nl

***Abstract*—Risk regulations and compliance management require business controls automation. Business processes execution yield event logs and analysis of these logs can produce valuable knowledge for organizational product and/or service improvements. In this paper we have explored a monitoring scenario of shipment request and evaluated it on the basis of Smart auditing framework. A combination of process mining techniques and business process ontologies is evaluated on simulated data in order to identify the auditing/monitoring capability of PROM plug-in's. The initial evaluation revealed that rule based audit is successful on machine-crafted data in PROM tool. Moreover, this paper also highlights lack of automated rule translation in LTL-checker (PROM plug-in) for smart audit framework's evaluations.**

***Keywords*—Compliance checking, Process mining, Information auditing**

I. Introduction

The digital universe is growing exponentially [1]. In parallel to that, development of technology give rise to the faster computers with extensive data storage capacity. Similar to most of the IT-related phenomena, the compliance of ever growing data complies with Moore's Law [2]. This growth in data and technology let organizations to think and prepare for ever growing data and its' management. Intelligent use of this "Big data" is a challenge. Not only the use but the compliance of it is even a bigger challenge.

The compliance, accuracy and correctness of big data are questions of concern however, the goal to have valuable insight from data. Organizations can analyze compliance of data by analyzing the business processes through business process audit [3]. It is known as information audit in other literature[4]. There exist two possibilities for such a compliance check, (a) Organizations do it by their own through internal audit or (b) by engaging independent audit organizations as external auditors. Compliance check by internal auditor and external auditor are overlapping and can be complemented by activities.

Compliance check done either by internal auditor or by external auditor is to ensure the quality of their implemented processes. During compliance checking of business processes, auditors relay on computerized data sources, such as event logs which are automatically in/out-putted to the information system. Event logs are the interpretation of the execution of the business processes in an organization. Use of event logs for compliance checking is supported by process mining [5], [6]. There exist numerous tools for analyzing event logs(i.e Fluxicon, Emit, ARIS PPM) [7] but we will use PROM to evaluate the smart auditing framework [8]. In the end of the paper we will be able to conclude that do we need to re-invent the wheel for Smart auditing framework's evaluation or can we use existing PROM plug-ins for smart audit purpose? Thus objective of the paper is two fold (a)evaluation of the Smart Auditing framework (b)evaluation of PROM tool for smart auditing.

In section II we start with the introduction of smart audit framework and later we have highlighted main features of the PROM tool. Section III discusses a compliance checking case followed by an evaluation in Section IV. Finally, section V discusses limitations to

0-7695-6347-3/17/$31.00 ©2017 IEEE
DOI 10.1109/FIT.2017.00049

validity, and draws conclusions, and indicates directions for future research.

II. Background

A. Smart Audit Framework

In the information systems field, we have witnessed substantive research on the topic of compliance and monitoring[9]. *Smart auditing framework* uses event-driven intelligent techniques such as process mining in order to improve efficiency and the effectiveness of audit process. However, it also takes normative auditing knowledge into account. Normative structure of REA(Resource Event Ontology) and possible business norms has been analyzed as well. Use of business norms and ontologies for compliance checking leads to the foundation of adaptive auditing system. On-line auditing is closely related to the idea of piggy-backing where auditor and auditee agree to use the same business data flow for both internal and external audit purposes. It is explained in Table I where use of *smart auditing framework* by internal and external auditor is discussed by comparing IST(as is)state of business with SOLL(to-be) state of the business (detail explanation of the framework can be seen from [8]). Usually external auditor has to perform some of the checks already done by internal auditing, and they can draw on these results, under certain conditions. So it is possible that external auditor relays on internal auditor and generate external audit reports based on internal audit reports. Table I shows modules performed by internal and external auditors. In addition to concept of piggy backing is used when three functions namely extraction(), conversion()and comparison() are called again and again as can be seen in Table I. Explanation of each function is as follows;

1) Extraction():It extracts data from a large data set. Input of this function will decide the processing steps involved for output generation. For example step by step procedure of extraction of numbers and sub-strings are different.
2) Conversion(): This function is based on casting function of programming languages (such as Java, C) in casting one type is converted into another type. For comparing two objects they need to be in comparable format. The purpose will be achieved by conversion() function by type-casting.
3) Comparison(): Comparison of two entities lead to the compliance checking. As the results of comparison, similar or dissimilar behaviours are described. IST an SOLL models will be compared by using compare(). This function is core of *smart auditing framework*.

Based on *smart auditing framework*'s internal audit process can be performed intelligently by choosing event log as an input data and REA a business norm ontology. Due to event log as a selected input, process mining has been chosen as an aggregation technique see (details can be seen from Chapter 4 [8]). For report generation external auditor usually uses almost all the same modules except adaptation. It is possible that external auditor uses the report produced by internal auditor and save time and money. In that case reliability of internal auditor's report and transparency of internal audit process is always a question of concern.

B. PROM

PROM is a process mining tool based on the process mining discipline is based on knowledge extraction during/from process execution. These processes have inbuilt hidden business norms, controls and business logic. PROM is a platform with a plug-in support for growing process-mining techniques. Not only processes but services can also be handled by PROM[6]. PROM basis are event logs and with the big data era logs are exponentially growing [10]. PROM evaluates these logs to conclude about the facts.It is platform independent and free downloadable software (www.processmining.org), Commercial version of PROM is DISCO by fluxicon (www.fluxicon.com), Perceptive Process Mining (www.perceptivesoftware.com), Celonis (www.celonis.de), and QPR ProcessAnalyzer (www.qpr.com). Detail study shows that PROM has potential to handle *Auditing* as well [11].Detailed information about the business processes are executed as event logs, which is analyzed by process mining and can produce audit trails. It is predicted by van der Aalst that use of process mining may change dramatically the role of auditor [4]. There exist more then 600 plug-ins in PROM. Different plug-ins are for different techniques such as Control-flow mining techniques, alpha algorithm, genetic mining, and multi-phase mining are developed. (For details of plug-in see [8]),

	Framework Modules	**Input**	**Output**	**Processing**	**IA***	**EA***
1a	Smart Sensors	Events	Storing data into database	Keep track of data continuity	Yes	No
1b	Effectors	Event-log	Processed Event-log	Data cleaning and maintaining	Yes	Reuse
2	I/p*DataExtraction()*	Processed Eventlog	Purpose related data	Extract purpose related data	Yes	No
3	IST	atomic/granular data	Targeted data set for compliance checking	Apply Sampling, clustering, processing techniques	Yes	Yes
4	SOLL *Data Extraction()*	Business rules, norms and business ontologies	Extracted Business norms in IF-then format and ontology as business models	Audit purpose related norms and ontologies will be extracted	Yes	No
5	Transform *Conversion()*	IST or SOLL model	IST and SOLL in comparable format	*Conversion()* Strings conversion into IF-Else rules (SOLL)and those can be applied on data set (IST)	Yes	No
6	Compliance Check *Compare()*	IST and SOLL	Compliance report with detective, preventive and corrective properties	*Compare()*: Checking rules (SOLL in machine readable format) against each instance of IST data	Yes	Yes
7	Decision Makers	Compliance Report	Action against report	Decision Making	Yes	Yes
9	Adaptation *Compare()* and *Extraction()*	Compliance report with less, more or not-violated rules and norms	More compliant and efficient	Action for gaining efficiency in the form of rules, control update, append	Yes	No

TABLE I
INPUT OUTPUT AND PROCESSING IN EACH MODULE OF SMART AUDITING FRAMEWORK(IA=INTERNAL AUDIT, EA=EXTERNAL AUDIT)

III. FROM ORDER PLACEMENT TO PRODUCT SHIPMENT: A COMPLIANCE CHECKING SCENARIO

In this section applicability of PROM for auditing purpose will be explored and we will evaluate our proposed framework on simulated data. We have considered a generalized company's scenario, which is the the outcome of detail discussion of import/export procedures with many companies. In simulated data we have considered a company which have worldwide manufacturing, sourcing, and distribution centers. The employees at central location, are responsible for coordinating activities between Company's customs, logistics, IT departments, the third party warehousing and many more. There is coordination of activities for securing all documentation and information provision and sharing. We name this company as G-company. We have focused the audit activity from customs domain's perspective where a set of rules and regulations [12] by EU customs are focused. These rules are the basis of designs/re-designs of G-company's control procedures. Customs require satisfactory system of appropriate controls. Thus in placing the security requirements for all goods entering or leaving the customs territory of the EU. One of the major EU requirement is secure transportation of goods from one location to other.

For evaluation purpose, the case study is limited to the process from order placement to product loading. Simulated data is randomly generated on the basis of business process diagram shown in Figure 1.

1) On PPSR an electronic file generated by the sales affiliates for Warehouse Management System.
2) Once the order has been placed, the shipment is pre-confirmed in the *warehouse management system*(WMS) to prepare the relevant documents for the shipment.
3) In parallel inventory is updated, goods are customs cleared, import duties and VAT(value added tax) are calculated and paid.
4) After loading the shipment is manually confirmed as shipped in the *warehouse management system* and at the same time is communicated by electronic transmission to the IS system of G-company,

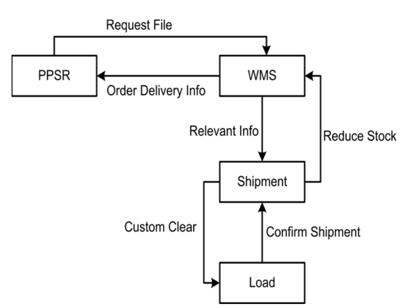

Fig. 1. Pick Pack and Ship Process (PPSR)

Code for data generation is written in Java, which constitute four methods including *Main* method. Major tasks are carried out by *'Shuffle Method'* which takes the ordered form of events, apply the simple swapping procedure and return an array which have randomized list of events. Finally, data is written on a compact file (For details see Appendix B [8]). PROM does't accept commonly used tabular format of data or excel sheet thus .XES files are generated. After successful input of data business norms application is possible which will leads us to evaluate smart auditing framework.

Generated data have hard coded built in controls for example segregation of duties: order receiving and sending cannot be done by one person. Moreover, money related issues are handled by financial department. Other tasks are generated by uniform distribution random function of Java.

For evaluation of the proposed framework we need to identify the 'audit authority', in our case study we assume that government want to check compliance of PPSR. Scope of audit is from order placement to shipment.

By considering business logic concrete business rules are formulated against simulated data. These business rules will lay down the foundation of SOLL model.

- **R1** Before sending order duties should be paid
- **R2** Before shipment request stock should be checked
- **R3** Once money is received then order should be sent
- **R4** Duties should be paid before shipment crosses the border

Following rules are generated by concrete translation of some of the REA concepts such as (i) Exchange duality, (ii) Economic reciprocity [13],

- **R5** Decrease in stock in warehouse should be equal to increase in inflow of money for G-company
- **R6** Number of orders received should be equal to number of sent and aborted orders

IV. Compliance Checking based on PROM

Process mining discover information from an event log. This discovered information is used not only for seeing business process execution but also for adaptation of already existing business processes. In-order to find the appropriate plug-in for smart auditing framework's evaluation, we have performed a detailed analysis of PROM plug-ins, shown in Table II. Following subsection discusses selected plug-in detail.

Module Name	PROM Plug-in	Tasks
Input Data	Log Analysis: Inspecting and Cleaning Log	Prepare data for processing
Data Analysis and IST Generation	Performance Sequence Diagram, Mine Petri net Using Alpha Algorithm	Mining the control flow perspective Mining case-related information
SOLL Generation and Compliance Checking	LTL checker Default Plug-in	Auditing based on normative structure

TABLE II
Correlation between smart auditing framework's modules and PROM's plug-in [8]

A. Analysis of Event Log

What is actually happening in company can be seen by inspecting the log. During inspection we have observed properties in the log for instance, complete and incomplete cases. For analysis we considered both complete and incomplete cases. Furthermore, the PROM tool provids an option of Export for saving the results of the cleaned log, so that we can avoid redoing work. Following questions are asked during analysis of data by using PROM 6.3.

1) How many cases (or process instances) are in the log?
2) How many tasks (or audit trail entries) are in the log?
3) How many originators are in the log?

These questions are answered by clicking on the tab Summary or by calling the analysis plug-in i.e Log Summary. We have found that 501 cases consisting of 3970 events where 8 originators are involved as can be seen from Figure 2.

Individual cases can also be analyzed by clicking the tab Inspector (cf. third left tab in Figure 2). View options of log (i.e Dashboard, inspector and summary) provide an abstract view of the data. As it is mentioned earlier that we have simulated data so here we can see classes, events, cases and many more which mimics the G-company's real data. In our case inspection and preprocessing of data provided an insight of the (simulated) business processes. Next step is how rule compliance is being handled by PROM?

B. Auditing Based on Smart Auditing Framework

Smart auditing framework discusses the use of some norms and standards for compliance checking(for details see II). From the description of G-company example, multiple rules need to be complied. For compliance check of rules discussed in section III two possible

Fig. 2. Log summary

scenario exist, (a) by using *LTL Checker default plug-in* where each condition is described by giving input to the LTL parameter A and B (b) by using *LTL Checker plug-in* where LTL rules file is generated for auditing.

In LTL we can identify which pre-defined formula is possible to use where we translated rule **R2** that is checking stock before placing the shipment request by using LTL checker default . The resulting screen shows that log have divided into two parts: (a) one with the cases that satisfy the property (or formula) and (b)another with the cases that do not satisfy the property. In the following we have considered rules **R1, R3, and R4** (see Section III).

Result of the LTL checker default shows that from satisfying rule of "check stock then send order" have 0.43 coverage, which shows that business process are not followed in an order as described by business norms (see Figure 3).

We can further mine the results by looking into each case and detail of the event. Each event has detail description about itself as how it is being performed, what was the time? What was the order and much more. Moreover, incorrect process instances can be checked by LTL plug-in again. An example output is shown in the Figure 3 where we have one satisfying and 3 non satisfying rules. We further explore non-satisfying cases.

From REA economic event's description we can see increment or decrement in the value of economic resources e.g sales of goods: which is continuous occurrence of an economic event. Based on rule **R1**:*before sending order duties should be paid* economic duality can be identified between between products in hand(i.e order sending) and money in-hand. Thus, Before shipment crosses the border and reached its destination we have to pay some money in the form of excise. Representation of excise in REA is discussed in [14] where complex scenarios of excise payment are explored which in return increase the efficiency of auditing (details can be see in Chapter 4 [8]).

We have observed that duality between increase in sales and reduction in money is not possible to represent in LTL default checker. We have partially checked the duality by using the business rule "excise is not paid before shipment crosses the border " see Figure 3.

Audit report states that mostly *excise is not being paid* but from our documentation study we have found that there are two possible ways for excise payment, (i) excise is paid when shipment crosses the border (ii) excise payment is managed by customs broker. Customs brokers pay weekly/monthly. In such cases it is not a violation it is only that customs broker's norms are overlooked in the norm list. The audit report will be provided not only to decision makers but also to the adaptation module for future use. The adaptation module performs necessary checks and adds the *customs broker's norms* in the business norms list. Based on this report decision makers (can be of company itself or customs authorities) can see which norm is less violated, mostly violated or not violated at all and take appropriate actions. In addition, the company reports its corrective actions to customs. The customs system checks whether these are in balance, and generates an alert only when this is not the case.

V. Conclusion

In this paper, we have targeted two tasks one is the evaluation of proposed framework and other is to see the possible use of PROM for auditing and compliance checking.

For answering audit related questions we have used a specialized plug-in known as LTL checker default and audited G-company's simulated data. For compliance checking there exist many plug-ins but most of them

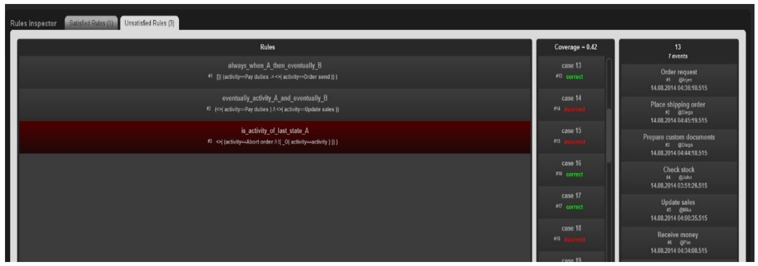

Fig. 3. Screen-shot of the few log patterns from Log

require a log and a (process) model as input [15]. Most companies are based on process-aware information system (PAIS) that require process models [9]. However, these process models may be incomplete because (a) at deployment stage one could not think of all possible scenarios, (b) due to dynamic nature of business: the way employees work may change but the prescribed process models remain the same. Thus, in the *smart auditing framework* we have proposed the use of ontological concept to populate business norms list and process model. In smart auditing framework we have proposed the use of intelligent techniques. Use of these techniques improve efficiency by (i) demanding fewer resources,(ii) implementing controls. Thus, we achieve added value not only in the form of reliability of data by using ontological concepts and process mining techniques. Moreover, we conclude that smart auditing framework is not only efficient and effective but it has capacity to adapt over time. We have seen how PROM provides an insight into the data and we have obtained answers to some basic questions about data as highlights in each section.

There exist many audit software's which we will use for further evaluation of *smart auditing framework's* but initially we have experimented with LTL checker default(PROM), where auditing can be performed by using ontological concepts. LTL checker default is partially inline with the smart auditing framework we can check event log against business rules but aggregated business rules are not possible to represent REA ontological concepts.

References

[1] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers, "Big data: The next frontier for innovation, competition, and productivity," 2011.

[2] R. R. Schaller, "Moore's law: past, present and future," *IEEE spectrum*, vol. 34, no. 6, pp. 52–59, 1997.

[3] F. A. Bukhsh and H. Weigand, "Smart auditing–innovating compliance checking in customs control," in *IEEE* 15^{th} *Conference on Business Informatics (CBI)*. IEEE, 2013, pp. 131–138.

[4] W. Aalst, A. Adriansyah, A. K. A. Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. Brand, R. Brandtjen, J. Buijs *et al.*, "Process mining manifesto," in *Business process management workshops*. Springer, 2012, pp. 169–194.

[5] A. P. Kurniati, G. P. Kusuma, and G. A. A. Wisudiawan, "Designing application to support process audit using process mining," *Journal of Theoretical and Applied Information Technology*, vol. 80, no. 3, p. 473, 2015.

[6] W. v. Aalst, *Process mining: discovery, conformance and enhancement of business processes*. Springer, 2011.

[7] N. Agarwal and L. Singh, "Process mining tools: A comparative analysis and review," *Advances in Computer Science and Information Technology (ACSIT)*, vol. 1, no. 2, pp. 26–29, 2014.

[8] F. A. Bukhsh, "Smart auditing: Innovative compliance checking in custom's control," Tilburg University, PhD Thesis, 2015.

[9] G. Grambow, R. Oberhauser, and M. Reichert, *Advances in Intelligent Process-Aware Information Systems: Concepts, Methods, and Technologies*. Springer, 2017, vol. 123.

[10] A. Abbasi, S. Sarker, and R. H. Chiang, "Big data research in information systems: Toward an inclusive research agenda." *Journal of the Association for Information Systems*, vol. 17, no. 2, 2016.

[11] W. v. Aalst, K. v. Hee, J. v. Werf, and M. Verdonk, "Auditing 2.0: Using process mining to support tomorrow's auditor," *Computer*, vol. 43, no. 3, pp. 90–93, 2010.

[12] European Commision, "Regulation (Ec) no 450/2008 of The European parliament and of the council of 23 april 2008 laying down the community customs code (modernised customs code)." *Official Journal of the European Union*, vol. 51, pp. L145/1–L145/64, 2008.

[13] W. McCarthy, G. Gal, and G. Geerts, "Semantic specification and automated enforcement of internal control procedures within accounting systems," in 10^{th} *World Continuous Auditing Conference*, 2005.

[14] F. A. Bukhsh and H. Weigand, "e-government controls in service-oriented auditing perspective: Beyond single window," in *Workshop on IT Innovations Enabling Seamless and Secure Supply Chains, In conjunction with the* 10^{th} *International Electronic Government Conference 2011 (EGOV-2011)*, S. Overbeek, Y.-H. Tan, and G. Zomer, Eds., 2011.

[15] A. A. Kalenkova, W. M. van der Aalst, I. A. Lomazova, and V. A. Rubin, "Process mining using bpmn: relating event logs and process models," in *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*. ACM, 2016, pp. 123–123.