Analyzing Transaction Codes with Data Requirements in Information Systems for Compliance Monitoring

A Manufacturing Case Study for Computational Auditing

Abstract. Companies, especially manufacturers, are operating in increasing demands of variability and customizability nowadays. They are faced with a multitude of complex rules and regulations that must be complied with, as well as supply chain disruptions that threaten business. Smart manufacturers take advantage of advanced information and manufacturing technologies to enable flexibility in physical processes. However, existing information systems are far from being sufficient and efficient for compliance monitoring in service-oriented manufacturing. Failure to interface well with different information systems is still a general phenomenon among manufacturers, leading to more problems such as delay of goods delivery or missing inventory. One of the main difficulties in fully integrating information systems arises from the highly flexible and volatile nature of manufacturing processes, which would need highly customizable and adaptive information systems. Moreover, improper human operations with transaction codes in information systems could hamper business processes. This research analyzes transaction codes in petri net model with data requirements investigating a manufacturing case. The results should provide insights for system integration and computational auditing.

Keywords: Auditing, Compliance, Information systems, Manufacturing.

1 Introduction

Companies usually have complex supply chains in which some parts are purchased in one region but assembled or manufactured in another region, and eventually shipped and sold to other regions. Multinational companies are compelled to comply with international trade regulations while moving goods quickly and cost effectively. This is particularly challenging due to the lack of information systems tailored to compliance management that integrates seamlessly with a company’s operational processes and Enterprise Resource Planning (ERP) environments, and at the same time covers key aspects of trade compliance and communication requirements. For instance, international companies are experiencing problems of adopting information systems to be compliant with customs during the process of custom clearance, due to the complex nature of the supply chain management across the border.

Therefore, international supply chains are facing risks concerning compliance [1], such as altering shipping documentations, inter-company manipulations, and fictitious inventory. Inventory management deals with the management of materials on a quantity and value basis, including all internal and external movement of goods in an enterprise, and the planning, entering, and documenting of these movements. The business processes within a company are executed using orders. To access these orders or running programs in information systems more rapidly each function in information systems is associated with a *transaction code*.

This research aims to devise methods for the detection and mitigation of deviations by analyzing transaction codes in information systems for manufacturers. Meanwhile, an approach with data requirements will be designed to reduce compliance problems in supply chains. In order to achieve these objectives, the main research question is:

How can incorrect inventory in ERP and compliance management systems be discovered by analyzing and detecting misusage of transaction codes?

This main objective leads to the following research steps:

Step 1: aims to discover how manufacturing and inventory management is performed and which components are typically used to assemble a product. To this end, we identify the main artifact used in the approach, i.e. transaction code diagram (further explained in Section 2.1) adopted for manufacturing and process models representing the internal organizational processes of a company.

Step 2: apply conformance checking methods to event logs in information systems to detect deviations from prescribed behaviors indicated by transaction code diagram.

Step 3: define appropriate data requirements for compliance in manufacturing and supply chains. Propose control methods to reduce misusage of transaction codes.

2 Case Study of ABC for Compliance Monitoring

ABC (anonymized) relies on international supply chains for the manufacturing of electronic equipment. The materials that are necessary for the production of the machinery are purchased both inside and outside the EU. The non-EU materials are stored in the customs warehouse, where they could undergo manufacturing operations with customs supervision; i.e. bonded materials are in the EU but have not been formally imported, hence are exempted from payment of import duties.

ABC has adopted several information systems to support manufacturing and inventory management: an ERP system, an interfacing system call TIBCO and a compliance management system called T&T. The logistical and financial records are registered in an ERP system provided by SAP. The monthly declarations for Customs cannot directly be created from SAP, due to the fact it does not maintain the customs status of the goods. Therefore, customs relevant data are processed from SAP to T&T. Based on SAP transaction codes, T&T is able to recognize the customs relevant transaction code using a correlation table. Upon receiving goods from suppliers, T&T recognizes whether the goods are free or bonded and process them differently.

According to standard ERP operations, goods movements as material flow should be recorded in terms of *movement types*, i.e. 3-digit *transaction codes* as information flow. A transaction code (e.g. t-code in SAP) consists of letters, numbers or both. Each type of material movement is given a unique *movement type*, followed after *transaction code*. The same transaction code can be used for different movement types. For example, MIGO-261 stands for ‘Goods issue for a production order’, and MIGO-262 means ‘reversal of 261’, i.e. cancellation of the production order.

In practice, some records are input manually into information systems at working sites, specifically when the production orders are being implemented. In this circumstance manual mistakes could happen. ABC regularly implements stock reconciliation procedures between information systems. However, large inventory differences underlying SAP and T&T are discovered since 2013 until 2017.

2.1 Transaction Code Diagram

To analyze transactions codes with data requirements, we derive a transaction code diagram representing the possible sequences of transaction codes that should be executed. *Movement type* is allowed per *transaction code*, and for each movement posted, a material document will be created and stored in the database. The movement type is important because it is used to control adjustment of inventories and the General Ledger account for financial purposes etc.

We represent transaction code diagrams as Petri nets. A Petri net is a tuple (P,T,F,m_i,m_f) where P is a set of places, T is a set of transitions representing transaction codes, $F \subseteq (P \times T) \cup (T \times P)$ is the flow relation connecting places and transitions, m_i is the initial marking and m_f is the final marking. Petri nets provide a formal semantics that can be exploited for automated analysis.

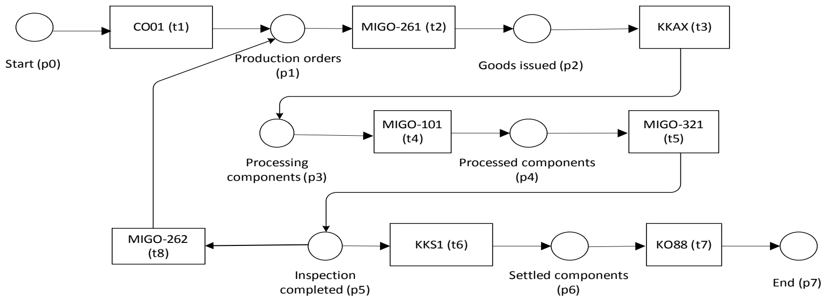

Fig. 1. An example of transaction code diagram regarding production process

Fig. 1 presents an example of the transaction code diagram in the form of Petri net regarding the production process. The normal sequence of movement types should follow a certain pattern as the process model, as the black arrows shown in Fig. 1. For example, movement type 261 initiates a “Goods issue for a production order”. When some components are cancelled, the components should always be reversed by 262 even though there would be sophisticated manufacturing processes after 261. By the end of the production phase any remaining unused components are returned to the inventory and a 262 against this order should be posted. Below is the detailed description with the transactions codes used:

1. Create production order - CO01
2. Issue goods for production order - MIGO-261
3. WIP (Work in Process) - KKAX (or KKAO for mass production)
4. Good receipts(GR) from production to Inventory - MIGO-101

5. Transfer posting stock in quality inspection - MIGO-321
6. Variance calculation - KKS1 (or KKS2 for the whole plant)
7. Settlement - KO88 (or CO88 for the whole plant)

2.2 Conformance Checking

Process mining is a promising means to systematically analyze data recorded by information systems like ERP [2]. It provides auditors a new and more comprehensive way for understanding the state of the control environment than the procedures that they rely on today. Process mining aims to extract knowledge from event logs. Event logs can be extracted from the contemporary information systems [3]. Information systems capture activities happening in the "real world". To perform process mining on data, the digital traces captured in the information systems must be extracted and transformed into *event logs*. We assume the existence of an event log where each event refers to a *case*, an *activity*, and a point in time, i.e. timestamp. An *event log* can be seen as a collection of *cases*. A *case* can be seen as a trace/sequence of *events*.

Conformance checking aims at the detection of inconsistencies between a process model and its corresponding execution log [4]. Logs represent observed execution sequences of activities from the normative process model. In the desirable case, logs completely comply with the behavior defined by the process model and are called valid execution sequences. In practice, observed execution sequences often deviate from predefined behavior.

As Fig. 1 shows, MIGO-262 is used when a production order needs partial quantity reversal. However, in this case the fault of using MIGO-101 happens. Some employees confuse the use of code 262 and 101, or they cannot distinguish whether these materials are received for cancellation reversal or production order. Instead of typing correct 262 as a reversal movement into systems to close the production order, they use 101 to receive goods instead. This fault may cause unnecessary duty repeatedly levied on the same goods, because 101 acts as an intermediate message may continue another duty levy process to production again automatically.

We analyzed ABC's 88538 records of monthly declarations to customs in 2012 and 41706 signals from compliance management system. In addition, data of two production orders in 2016 is used for conformance checking. Further analysis of the root causes of inventory differences can be categorized as three main reasons: Deviation from the standard operating procedure by employees as agreed in the workflow instructions; Misusage of the movement types in ERP by which items are incorrectly assigned in the returning stock; Constraints in the ERP system, i.e. infeasibility to customize for every new product as well as the flexible manufacturing process.

2.3 Data Requirements and Controls for Compliance Monitoring

The feasible control method is to enrich data requirements in ERP. A production order should not only define which material is to be processed, at which location, at what time and how much work is required. It also defines which resources are to be used and how the order costs are to be settled, especially regarding compliance status.

In the ABC case the compliance status refers to customs status of each material items. As soon as a planned order or other request is generated from material requirements planning, the information is passed on to shop floor control. The order-relevant data including compliance status is also added to ensure complete order processing.

The alternative solution is to place Internet of Things(IoT) / RFID sensors at each transition in the Petri net model for fault mitigation as Fig. 2 illustrates. The deployment shows a situation in which we can differentiate between 'easy' and 'difficult' cases. WoPeD (Workflow Petri Net Designer) is an open-source software developed for modelling, simulating and analyzing processes described by Petri nets. A built-in resource editor in WoPed allows for each process the definition of a resource model, i.e. roles, groups, and objects. Additionally, sensors in the transitions can have resource class assignments. Based on the capacity requirement per transition, we can calculate the capacity requirement of each resource class.

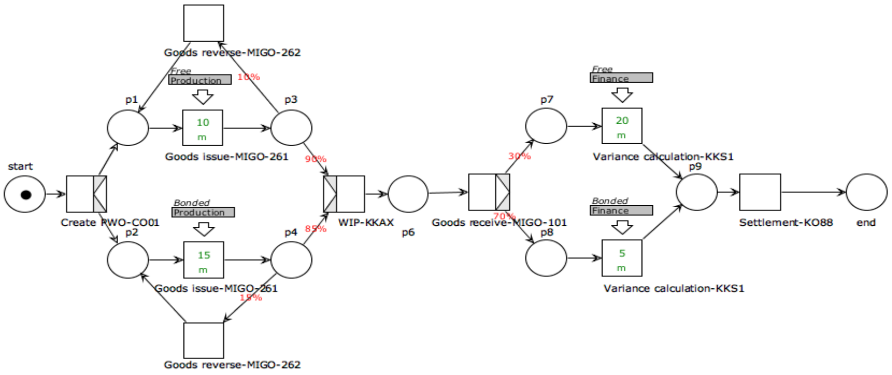

Fig. 2. An example of sensor deployment in Petri net model for fault mitigation

Given an expected supply of components and a number of assumptions about their processing, we can use simulation and/or the queueing theory to determine the capacity requirement during a particular period. Two different algorithms are implemented, both based on stochastic distributions of arrival rate, task service times and XOR branching probabilities [5]: Firstly, a quantitative simulator engine for the random construction of execution traces, allowing to derive quantitative properties like average waiting or completion time. Secondly, a capacity planner in order to calculate the optimal number of resource objects for each resource class.

Suppose the sensor in goods issue for an ‘easy’ case of free components takes an average of 10 minutes, whereas for a ‘difficult’ case of bonded components (extra information needs to be registered) it takes an average of 15 minutes. The sensor in variance calculation for an easy case of bonded goods (no need to pay duties) takes an average of 5 minutes, whereas for a difficult case of free components it takes an average of 20 minutes. On average, 70% cases are classified as bonded goods, 30% as free goods. Suppose four resource classes are in this case: Bonded, Free, Production and Finance. A resource belongs to either Production or Finance, but not to both.

It is assumed that the time taken to perform those activities which require no resources is negligible. For the others, the average processing time in minutes is shown.

For example, the sensor in activity ‘Bonded’ ‘Goods issue’ takes an average of 15 minutes. In general, 85% of the bonded goods issue cases have been moved to WIP and 15% are reversed. If we know the number λ of new components arrive each day, then we can calculate the capacity requirement of sensors for each activity.

Because there are always fluctuations in the supply of components and the processing times, it is not always possible to make full use of the capacity available. It is therefore not sensible to assume that the resources will be utilized to their full capacity. To illustrate this, let us examine a process with 80% capacity. When λ=50, WePed gives the results that the minimum number of sensors for Finance is 1.11, Production 3.74, Bonded 2.71, Free 2.15. We can calculate the results under other conditions once the actual situation and operational data in ABC is available.

3 Conclusion

This research aims to support businesses like ABC for compliance monitoring in manufacturing. Through tracing abnormal movement sequence patterns, deviations can be distinguished. Ideally the results of these research should be applied in ERP and compliance management systems as an active fault detection and mitigation module. Once the system detects abnormal code input, it can generate alerts and a composite recommendation of internal controls immediately and directly to the relevant managers. If these deviations can be found in the early period, it is more likely and easier to timely correct them in order to eliminate or minimize their negative impacts. To this end, we analyze transaction codes based on data analytics using process mining and propose data requirements that allow:

1. identifying different deviation scenarios related to trade and compliance;
2. checking whether processes are in place and functioning as intended;
3. assessing the impact of transaction codes misusage in compliance monitoring;
4. providing recommendations about internal controls and risk mitigation mechanisms depending on different types of misusages and deviations.

References

1. Spira, L., Page, M.: Risk management: The reinvention of internal control and the changing role of internal audit. Accounting, Auditing & Accountability Journal. 16 (4), 640-661 (2003).
2. Jans, M., Alles, M., Vasarhelyi, M.: The case for process mining in auditing: Sources of value added and areas of application. International Journal of Accounting Information Systems. 14 (1), 1-20 (2013).
3. van der Aalst, W. M., Reijers, H. A., Weijters, A. J., van Dongen, B. F., De Medeiros, A. A., Song, M., et al.: Business process mining: An industrial application. Information Systems , 32 (5), 713-732 (2007).
4. Rozinat, A., van der Aalst, W.: Conformance checking of processes based on monitoring real behavior. Information Systems. 33 (1), 64-95 (2008).
   W. M. P. van der Aalst, K M. van Hee: Workflow Management: Models, Methods, and Systems MIT Press (2002).