Association for Information Systems
AIS Electronic Library (AISeL)

Wirtschaftsinformatik Proceedings 2009 | Wirtschaftsinformatik

2009

CONTINUOUS COMPLIANCE MONITORING IN ERP SYSTEMS - A METHOD FOR IDENTIFYING SEGREGATION OF DUTIES CONFLICTS

Patrick Wolf
PricewaterhouseCoopers AG WPG, Hamburg

Nick Gehrke
PricewaterhouseCoopers AG WPG, Hamburg

CONTINUOUS COMPLIANCE MONITORING IN ERP SYSTEMS - A METHOD FOR IDENTIFYING SEGREGATION OF DUTIES CONFLICTS

Patrick Wolf, Nick Gehrke[1]

Abstract
Segregation of Duties (SOD) can be seen as one major class of control activities within a company's Internal Control framework, contributing to the reliability of financial reporting. In recent years, SOD controls in terms of user access rights have experienced a surge of attention in particular, mostly due to the growing reliance of business processes on ERP systems. This paper presents a method for automatically identifying SOD conflicts in user access rights as one component of a continuous compliance monitoring framework. The paper further demonstrates the application of the proposed method in a real world project.

1. Introduction

In recent years, the growing number of corporate scandals (e.g. Barings Bank, Enron, Worldcom, Siemens, Société Générale) has led to tighter regulatory and statutory requirements regarding a company's Internal Control over Financial Reporting (e.g. Sarbanes-Oxley-Act). According to the Committee of Sponsoring Organizations of the Treadway Commission (COSO), Internal Control can be defined as *"a process, effected by an entity's board of directors, management and other personnel, designed to provide reasonable assurance regarding the achievement of objectives in the following categories: 1. Effectiveness and efficiency of operations. 2. Reliability of financial reporting. 3. Compliance with applicable laws and regulations"* [6]. To ensure the achievement of these objectives, COSO proposes the implementation of a company-wide Internal Control framework consisting of five interrelated components: Control Environment, Risk Assessment, Control Activities, Information and Communication, and Monitoring [6]. The need for implementing a suitable and available Internal Control framework is also underlined by the aforementioned regulatory and statutory requirements [12].

Within a company's Internal Control framework, Segregation of Duties (SOD) can be classified as a major class of control activities. It contributes to the reliability of financial reporting [13] by preventing any single employee from having complete control over all phases (authorization, custody, record keeping and reconciliation) of a business transaction [15, 18], thus, avoiding a conflict of interests and preventing fraud [13, 14]. The term "fraud" is used in accordance with the

[1] PricewaterhouseCoopers AG WPG, Hamburg, Germany

International Standard on Auditing (ISA) 240 and refers to *"an intentional act by one or more individuals among management, those charged with governance, employees, or third parties, involving the use of deception to obtain an unjust or illegal advantage"* [14]. If a company's Internal Control is not working effectively due to inadequate SOD, then fraudulent activities, such as misappropriation of assets and fraudulent financial reporting, may not be prevented which, in turn, could result in a material misstatement in the financial statements of the respective company [14]. According to a survey recently published by the accounting and consulting firm PricewaterhouseCoopers (PwC), the average loss from fraud over two years per company in 2007 was US$ 2,420,700 [16].

Considering the relevance of SOD for the effectiveness of a company's Internal Control over Financial Reporting, this paper proposes a method for identifying existing SOD conflicts in the user access rights of different enterprise resource planning (ERP) systems. From our practical experience, there is a strong demand for such a monitoring method, especially due to the growing reliance of business processes on ERP systems [9] and the latest corporate scandals (e.g. the Siemens AG has lately chosen Security Weaver as its global software platform for monitoring SOD conflicts[2]). While the continuous monitoring of SOD conflicts in ERP systems combined with the continuous monitoring of system transactions and system settings is recently marketed under the term "continuous compliance monitoring" by several consulting firms, such as PwC and KPMG, and software vendors, such as ACL CCM, Approva BizRights and Security Weaver, there seems to be a lack of scientific debate regarding continuous monitoring methods. This paper tries to fill this gap and is based on the Design Science paradigm [11]. The proposed method addresses a relevant business problem (see introduction) and constitutes a viable artefact according to Hevner et al. Research rigor is achieved by applying the established method engineering approach within the construction process. The practicability of the method is then evaluated in a real world project. The research contribution of the paper can be seen in bridging the gap between the practical and theoretical debate regarding continuous compliance monitoring.

The remainder of the paper is structured as follows: In chapter 2, the method engineering approach is presented as the theoretical foundation of the method development process. Based on the understanding of the five constituent elements of a method (activities, outcomes, techniques, roles and metamodel), chapter 3 focuses on describing the most important elements of our proposed method for identifying SOD conflicts in system-based user access rights (activities, outcomes and techniques). Thus, the elements metamodel and roles will not be covered within this paper. In chapter 4, the method is evaluated by applying it in a real world project. Finally, a conclusion is given and further research needs are outlined.

2. Method Elements and Method Engineering

According to Brinkkemper, the word method comes from the Greek "methodos", meaning way of investigation [4]. In the context of systems development, he defines a method as *"an approach […] based on a specific way of thinking, consisting of directions and rules, structured in a systematic way in development activities with corresponding development products"* [4]. Further definitions of methods have been given, for example, by [1], [2], [8] or [18]. A synopsis of these and other definitions can be found at [3] who derive four fundamental defining attributes of a method: goal oriented, systematically structured, principles-based, and intersubjectively repeatable.

[2] http://www.reuters.com/article/pressRelease/idUS154425+25-Feb-2008+MW20080225 (Call-Up: 22/07/2008)

Method Engineering (ME) is an approach which has emerged in the IS field in the nineteen-eighties due to the *"fundamental observation […] that no one method is equally suitable to all kinds of problem domains"* [19]. Based on this observation, ME can be understood as an engineering discipline which focuses on the systematic design, construction and adoption of methods for various purposes (e.g. information systems development) [7]. Here, a method is not conceived *"as a single intertwined and interdependent entity but as a set of disparate fragments"* [10].

Gutzwiller has analyzed several ME approaches and derived the following five general elements of a method: "activity", "role", "outcome", "technique", and "metamodel" [8]. Again, an overview of alternative definitions can be found at [3]. According to Gutzwiller, an activity is a functional unit of action which aims at creating one or more defined outcomes (e.g. a functional specification). Activities may consist of sub-activities (forming a hierarchical structure) and can be ordered in a sequence (procedure model). Techniques describe in detail how a certain outcome or a group of logically interrelated outcomes is created. The metamodel is the conceptual data model of the outcomes and visualizes their overall interrelationships. Finally, roles are aggregations of certain activities required to fulfil a certain function within the company and are normally performed by employees or organizational units [8].

Based on the aforementioned understanding, the following chapter focuses on describing the activities, outcomes and techniques of our method for identifying SOD conflicts in ERP-systems as well as the interrelationships of these elements. Neither metamodels nor role aspects will be covered in this paper. The application of the method in a real world project is demonstrated in chapter 4.

3. Activities, Outcomes and Techniques of the Method

The activities, outcomes and techniques of the proposed method are illustrated in Fig. 1.

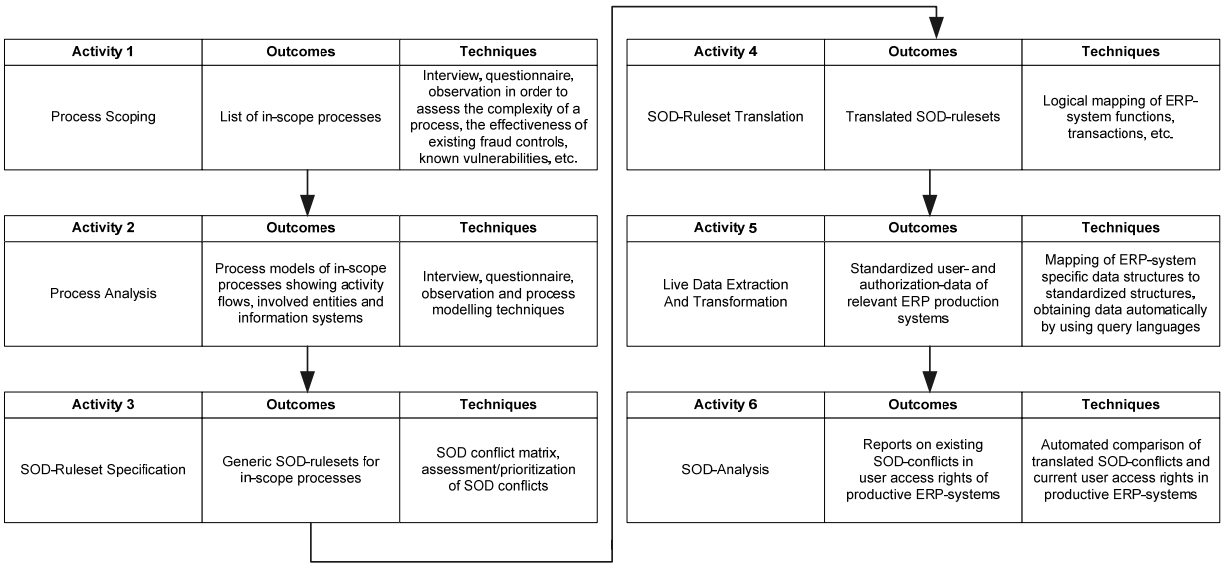

Fig. 1: Activities, outcomes and techniques of the SOD conflicts identification method

Process Scoping

The process scoping activity is the starting point of the presented method and aims at identifying those processes which will later be subject to the SOD analysis. This selection should be based on a thorough assessment of each process in terms of the probability of fraud occurrence and the expected extent of such events. Typical examples of high risk processes in most companies are the purchase-to-pay process (P2P) or the order-to-cash (O2C) process. Assessment techniques, like interviews and questionnaires, should especially take into account the complexity of a certain process, such as degree of automatization and interfaces, past fraud cases, existing opportunities to commit fraud (e.g. ineffective controls), as well as employees' incentives (e.g. existing pressure, personal fraud justification) [17].

Process Analysis

The aim of the process analysis activity is to gain a thorough understanding of the selected processes by decomposing each process in its flow of activities and by identifying the organizational entities and information systems (e.g. ERP systems) which are involved in performing these activities. Techniques that should be used for gathering the required information are interviews, questionnaires and observations. Process modelling tools (e.g. IDS Scheer ARIS, BOC ADONIS) might be used in order to facilitate the systematic creation of the process models.

SOD Ruleset Specification

The main objective of the third activity is to specify the corporate SOD ruleset which constitutes the total set of SOD conflicts considered relevant for the regarded processes. An SOD conflict is defined as a combination of exactly two process activities which hold the risk of fraud if both are performed by the same individual. SOD conflicts are relevant if both the probability of their exploitation and the expected damage resulting from such an exploitation exceed a threshold depending on a company's risk appetite [5]. This threshold ensures the manageability of the ruleset by eliminating all non-relevant conflicts. The conflicts contained in the ruleset should be formulated in a system-independent way which guarantees that the ruleset is overall applicable and not tailored to a specific system environment. This becomes especially important when dealing with more than one company (e.g. an affiliated group) and different system environments (e.g. SAP, Baan). An exemplary extract of a ruleset is given in Tab. 1.

Process	SOD Conflict	Risk
P2P	Create & Maintain Vendor Records vs. Process Vendor Invoices	A user could set up fictitious vendor accounts (or alter existing accounts inappropriately) and create fictitious invoices resulting in unauthorized payments.
P2P	Process Vendor Invoices vs. Process Outgoing Payments	A user could process payments for fictitious or invalid invoices.
P2P	Create & Maintain Vendor Records vs. Process Outgoing Payments	A user could set up fictitious vendor accounts (or alter existing accounts inappropriately) and initiate payments to these fictitious vendors.

Tab. 1: Exemplary extract of a SOD ruleset for the P2P process

SOD Ruleset Translation

The next activity of the proposed method is based on the premise that all or most of the process activities considered in the SOD ruleset (e.g. "Create & Maintain Purchase Order") are actually performed in an ERP system, thus requiring the translation of the system-independent ruleset in the specific transaction codes, function codes, authorization objects, etc. of the respective system. Again, this activity becomes particularly important when dealing with several subsidiaries of an affiliated group which all perform basically the same processes (e.g. P2P) on heterogeneous ERP systems (e.g. SAP, Oracle Financials, Baan). In such a case, the ruleset must be consistently tailored to each ERP system. A practical example is given in paragraph 4.3.

Live Data Extraction and Transformation

After having translated the SOD ruleset, the fifth activity aims at extracting the current user access rights from the relevant database tables of the productive ERP systems and transforming that data in a standardized format for analysis purposes. In order to automate the extraction, the implementation of customized Standard Query Language (SQL)-scripts which may be triggered periodically is recommended. The first step in creating such a script consists of identifying the relevant database tables which contain the user access rights data. In an ERP system supporting purely role based access rights, the tables illustrated in Tab. 2 should be considered at minimum.

Table	Description
User Master Data	Contains all users of the system
Transaction Master Data	Contains all possible business transactions supported by the system
Role Master Data	Contains all roles defined in the system
Mapping Users To Roles	Contains all relationships of users and roles. A role can be assigned to multiple users and a user can have multiple roles (m:n).
Mapping Roles to Transactions	Contains all relationships of roles and business transactions. A role can contain multiple business transactions and a transaction can be assigned to multiple roles (m:n).

Tab. 2: Relevant user access rights tables in a role based access rights system

The automated transformation of the contents of the identified tables in a standardized format by using appropriate SQL queries can then be seen as the second step. However, in the case of ERP systems whose authorization mechanisms are not purely role based, additional data transformations must be developed. Typical issues which might be encountered are: (1) access rights are directly assigned to users without using roles or in addition to existing roles; or (2) roles do not only contain business transactions (positive list), but also entries with explicit negations of transactions (negative list). In such cases, constellations must be considered where one role grants one user access to a specific transaction and another role negates this access at the same time.

SOD Analysis

Within this activity, the standardized data extracts from the previous step are taken and analyzed in order to determine existing SOD conflicts in the user access rights assigned for the productive ERP systems. For each user and each conflict of the translated SOD ruleset, it must be checked if the respective user disposes of sufficient access rights to perform both activities of the analyzed conflict. In such a case, the result report of the SOD analysis should show the existing conflict together with additional information (e.g. user ID, conflicting access rights) to allow the remediation of the conflicting access rights. An example of a report is given in paragraph 4.4.

4. Application of the Method in a Real World Project

This chapter aims at demonstrating the application of the proposed method in a real world project. Firstly, a short overview is given describing the objectives and the general requirements of the project. Subsequently, the aforementioned activities, techniques and outcomes are described in the context of the project. Due to the sensitivity of the subject, all data have been made anonymous.

Project Overview

In the forefront of our project, the client had started to implement a continuous SOD compliance process based on the analysis tool Security Weaver (SW). Although SW is primarily focussed on SAP installations, the scope of the software was extended within the project to cover other NON-SAP systems (e.g. Exact Globe, Baan, Oracle Financials) as well. At the end, more than 60 worldwide located installations of NON-SAP systems were integrated in the automated SOD compliance process. The activities performed in order to analyze the user access rights within those systems are illustrated in Fig. 2 and described below: (1) the current user access rights are extracted from each productive NON-SAP system on a regular basis; (2) the extracted data is transformed in a common standardized format; (3) the standardized data is transferred to the central SW instance (e.g. via FTP), uploaded and analyzed; (4)/(5) the results are made accessible via web interface allowing the system owners to download the reports for the purpose of remediation.

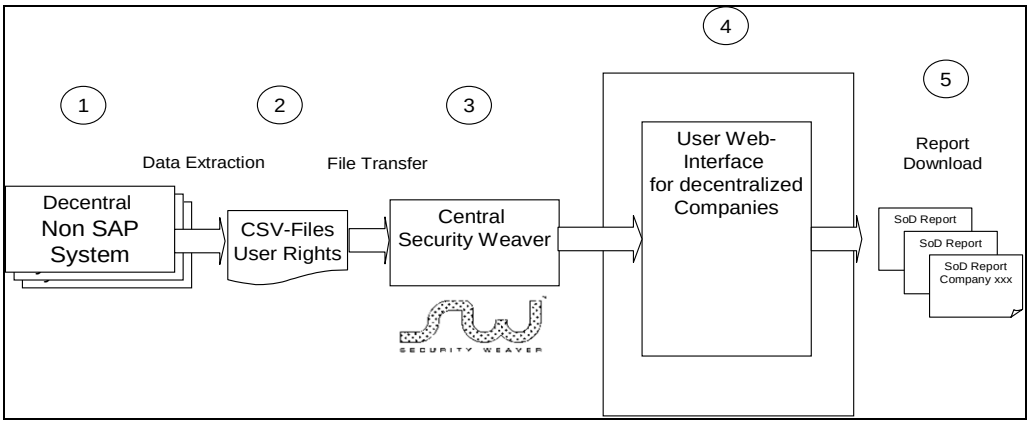

Fig. 2: Activities within an exemplary continuous SOD compliance process

In the context of the overall SOD project, the main objective of our sub-project was to initially analyse the user access rights established in the productive NON-SAP systems of several selected subsidiaries with regard to existing SOD conflicts. The scope of the analysis was further limited to cover only P2P-related conflicts as well as some conflicts relating to the finance and the system administration process.

SOD Ruleset Specification

The SOD ruleset for the P2P process had already been defined by our client and, thus, marked the initial point of our sub-project. To obtain the ruleset, the client first identified all relevant activities performed within his P2P processes and then created an SOD matrix based on these activities to derive the potential conflicts. Tab. 3 shows an excerpt of the SOD matrix for the P2P process. The crosses indicate existing conflicts between the activities.

	CMVR	PVI	POP	MBA	PIP
CMVR	-	X	X		
PVI	X	-	X		
POP	X	X	-	X	
MBA			X	-	X
PIP				X	-

Tab. 3: Exemplary SOD matrix for the P2P process[3]

SOD Ruleset Translation

The system-specific translation of the P2P ruleset is demonstrated on the example of the NON-SAP system Exact Globe. Within Globe, the table "pwfunc" contains all transactions supported by the system. Tab. 4 shows an extract of this table.

transactionID	exename	Transactiontitle
650888	EBUDGETALLOCATION	Budgets
650892	BALANCELIST	Receivables history
650896	BALANCELIST	Payables history
650925	ENTRYREPORT	To be processed
652065	EFENTRY	Invoices
652066	EFENTRY	To create credit notes

Tab. 4: Extract of transactions supported by Exact Globe

To translate a specific conflict of the SOD ruleset, it is necessary to identify all transactionIDs contained in the table "pwfunc" which logically belong to one of the two activities that make up a conflict and assign those transactionIDs to the respective activity. It is immediately obvious that this activity requires a profound knowledge of the regarded ERP system and the transactions it supports. Tab. 5 illustrates the process activity "Process Vendor Invoices" tailored to Exact Globe.

Process activity	TransactionID	GroupID	Transaction Title
Process Vendor Invoices	G0000000018	08	Exchange rates
Process Vendor Invoices	G0000000054	NO	Transactions
Process Vendor Invoices	G0000000062	NO	Financial entries
Process Vendor Invoices	G0000000060	04	Enter
Process Vendor Invoices	G0000000068	NO	Make recurring purchase entries
Process Vendor Invoices	G0000000063	NO	Make recurring general journal entries
Process Vendor Invoices	G0000000121	07	Invoices & Bank/Cash
Process Vendor Invoices	G0000000102	08	Invoices & Bank/Cash
Process Vendor Invoices	G0000000119	04	Process
Process Vendor Invoices	G65047	NO	Purchase
Process Vendor Invoices	G65076	NO	General journal
Process Vendor Invoices	G65303	07	Exchange rates

Tab. 5: Exemplary translation of the activity "Process Vendor Invoices"

It should be noted that a user needs only one of the transactionIDs marked with a "NO" in the GroupID column of the table above to perform the activity "Process Vendor Invoices" within Exact Globe ("or"-conjunction). On the other hand, values other than "NO" in the GroupID column point out that a user must possess all transactionIDs belonging to the same group in order to be able to

[3] Legend of abbreviations: Create & Maintain Vendor Records (CMVR), Process Vendor Invoices (PVI), Process Outgoing Payments (POP), Maintain Bank Accounts (MBA), Process Incoming Payments (PIP)

perform the activity "Process Vendor Invoices". In the case of group "04", only entering AND processing invoices is deemed critical. Users with access to only one of these transactions cannot perform the activity "Process Vendor Invoices" on their own.

Live Data Extraction and SOD Analysis

For obtaining the user access rights data from the productive NON-SAP systems, customized SQL scripts were created and implemented. The data transfer was performed using standardized csv-files. The access rights data contained in the files was then uploaded into the central SW instance. Subsequently, the analysis of the user access rights data in terms of existing SOD conflicts was performed automatically in SW by applying the appropriate system-specific SOD ruleset to the uploaded data. A simplified exemplary result report showing one identified SOD conflict ("Maintain Vendor Records vs. Process Vendor Invoices") for the user "4711" is depicted in Tab. 6. The report also indicates the transactionIDs causing the conflict, the roles of the user containing the conflicting transactionIDs and the respective process activities to which the transactionIDs have been mapped. Based on the report, the existing conflict could be solved, for example, by revoking the user's access to transactionID 650476.

Company	User	Conflict	Process Activity	Role	Transaction-ID
100	4711	Maintain Vendor Records vs. Process Vendor Invoices	Maintain Vendor Master Records	Sales invoices23456	650124
100	4711	Maintain Vendor Records vs. Process Vendor Invoices	Maintain Vendor Master Records	00_PURCHASE AGENT	650120
100	4711	Maintain Vendor Records vs. Process Vendor Invoices	Process Vendor Invoices	00_PURCHASE AGENT	650476

Tab. 6: Simplified report of identified SOD conflicts

At higher organizational levels, more aggregated reports are commonly demanded (e.g. existing SOD conflicts over all entities). Fig. 3 shows a scatter plot which we used within our project for reporting the initial status of SOD conflicts in several selected entities (C001 to C007) and the status after a first remediation (visualized via arrows) to the client's management.

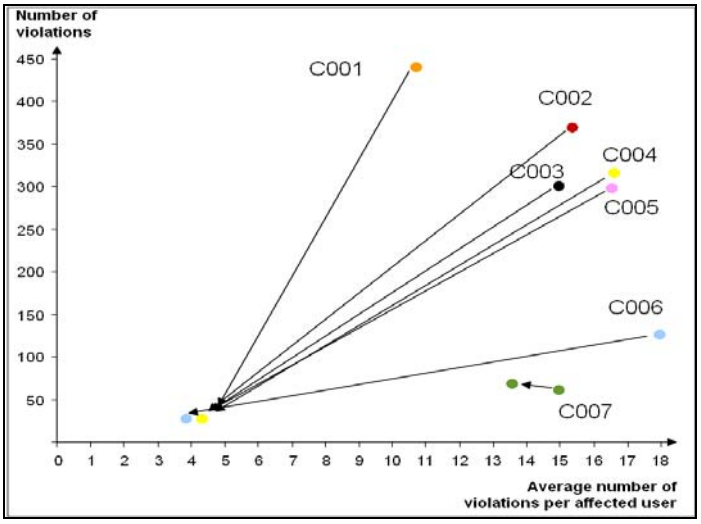

Fig. 3: SOD report graph showing the number of conflicts per entity (C001 to C007)

While the scatter plot in Fig. 3 shows only the total number of SOD conflicts for each entity, the graph in Fig. 4 gives more detailed information on the frequency of each single conflict. On the x-axis of the graph, the analyzed conflict IDs are shown (e.g. AP01 to AP15), whereas the y-axis indicates the percentage of ERP system users within the regarded entities (C001 to C007) having the respective conflict. A broken line occurs if a conflict is not applicable to the considered system.

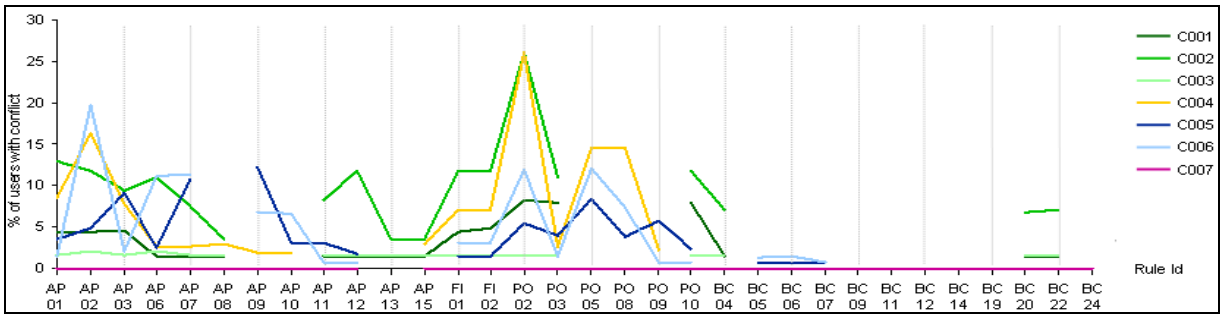

Fig. 4: SOD report graph showing the frequency of each conflict per entity

5. Conclusion

This paper proposed a method for identifying existing SOD conflicts in the user access rights of different ERP systems. The method consists of six interrelated activities whose practicability was evaluated in a real world project. From this evaluation, some "lessons learned" can be derived:

1. The creation of a complete, accurate and consistent SOD ruleset without redundancies is highly important as it constitutes the basis for all subsequent activities. Thus, an independent quality assurance assessment of the ruleset should be performed prior to any implementation activities.
2. The system-specific translation of the ruleset demands highly experienced system users to ensure the accuracy and completeness of the mapping. Again, an independent quality assurance assessment has proved to be important.
3. In the case of ERP systems whose authorization mechanisms are not purely role-based, additional support tools for transforming the data into the required format had to be developed. These additional efforts should be considered when planning similar projects.
4. The redesign of user access rights in order to ensure compliance with the SOD principle may result in a substantial change of organizational work routines, e.g. if employees are not allowed to perform certain activities in the ERP systems anymore due to existing SOD violations. These required changes should be taken into account when starting an SOD project.

The presented method constitutes a first step towards a comprehensive continuous compliance monitoring framework that addresses the particularities of complex ERP systems. Further research should focus on developing sound methods that can be used to automatically monitor system transactions and system settings. Due to the fact that issues with segregation of duties usually stem from a complex interaction of technical and non-technical activities that are usually not properly reflected in the master data of an ERP system alone, further research should also concentrate on the complex interplay of these types of activities and the media breaks that typically occur between them.

6. References

[1] BALZERT, H.: Lehrbuch der Software-Technik, Spektrum Akademischer Verlag GmbH: Heidelberg/Berlin 1998.

[2] BECKER, J. / KNACKSTEDT, R. / HOLTEN, R. / HANSMANN, H. / NEUMANN, S.: Konstruktion von Methodiken, in: Becker, J. / Grob, H. L. / Klein, St. / Kuchen, H. / Müller-Funk, U. / Vossen, G. (Eds.), Arbeitsbericht des Instituts für Wirtschaftsinformatik, No. 77, Universität Münster: Münster 2001.

[3] BRAUN, C. / WORMANN, F. / HAFNER, M. / WINTER, R.: Method Construction A Core Approach to Organizational Engineering, in: Haddad, H. M. / Liebrock, L. M. / Omicini, A. / Wainwright, R. L. (Eds.), The 20th ACM Symposium on Applied Computing (SAC2005): Santa Fe, New Mexico USA, 2005, pp. 1295-1299.

[4] BRINKKEMPER, S.: Method engineering: engineering of information systems development methods and tools, in: Information and Software Technology, No. 38, 1996, p. 275-280.

[5] COMMITTEE OF THE SPONSORING ORGANIZATIONS OF THE TREADWAY COMMISSION: Enterprise Risk Management - Integrated Framework - Executive Summary, 2004.

[6] COMMITTEE OF THE SPONSORING ORGANIZATIONS OF THE TREADWAY COMMISSION: Internal Control - Integrated Framework, o.J., Internet: http://www.coso.org/IC-IntegratedFramework-summary.htm (Call-Up: 22/07/2008)

[7] GOEKEN, M.: Entwicklung von Data Warehouse Systemen: Anforderungsmanagement, Modellierung, Implementierung, Vieweg + Teubner: Wiesbaden 2006.

[8] GUTZWILLER, T.: Das CC RIM-Referenzmodell für den Entwurf von betrieblichen, transaktionsorientierten Informationssystemen, Hochschule St. Gallen für Wirtschafts-, Rechts- und Sozialwissenschaften, Dissertation 1994.

[9] HENDRAWIRAWAN, D. / TANRIVERDI, H. / ZETTERLUND, C. / HAKAM, H. / KIM, H. H. / PAIK, H.: ERP Security and Segregation of Duties Audit: A Framework for Building an Automated Solution, in: Information Systems Control Journal, Vol. 2, 2007, pp. 1-4.

[10] HENDERSON-SELLERS, B. / GONZALEZ-PEREZ, C. / SEROUR, M.K. / FIRESMITH, D. G.: Method engineering and COTS evaluation, in: ACM SIGSOFT Software Engineering Notes, Vol. 30, No. 4, 2005.

[11] HEVNER, A. R. / MARCH, S. T. / PARK, J. / RAM, S.: Design Science in Information Systems Research, in: MIS Quarterly, Vol. 28, No 1, 2004, pp. 75-105.

[12] HÜLSBERG, F. / FELLER, S. / PARSOW, C.: Anti-Fraud-Management - Reduzierung von Haftungsrisiken und Vermögensschädigungen, in: ZRFG, Vol. 5, 2007, pp. 204-208.

[13] INSTITUT DER WIRTSCHAFTSPRÜFER IN DEUTSCHLAND E.V.: IDW-Prüfungsstandard: Das interne Kontrollsystem im Rahmen der Abschlussprüfung (IDW PS 260), in: WPg, 2001, pp. 821ff.

[14] INTERNATIONAL FEDERATION OF ACCOUNTANTS: International Standard on Auditing 240 (Revised) - The Auditor's Responsibility to Consider Fraud in an Audit of Financial Statements, 2004.

[15] LUECK, W.: Elemente eines Risiko-Managementsystems, in: Der Betrieb, Vol. 01/02, 1998, pp. 8-14.

[16] PRICEWATERHOUSECOOPERS: Economic crime: people, culture and controls - The 4th biennal Global Economic Crime Survey, 2007.

[17] RAMOS, M.: Auditors' Responsibility for Fraud Detection, in: Journal of Accountancy, Vol. 195, No. 1, 2003, pp. 28-36.

[18] SCHEER, A.-W.: EDV-orientierte Betriebswirtschaftslehre: Grundlagen für ein effizientes Informationsmanagement, Springer, Berlin 1990.

[19] TER HOFSTEDE, A.H.M. / VERHOEF, T.F.: On the feasibility of situational method engineering, in: Information Systems, Vol. 22, No. 6, 1997, pp. 401-422.