A Meta-Model for Real-Time Fraud Detection in ERP Systems

Anna Fuchs
University of Wuerzburg
a.fuchs@uni-wuerzburg.de

Kevin Fuchs
University of Wuerzburg
kevin.fuchs@uni-wuerzburg.de

Fabian Gwinner
University of Wuerzburg
fabian.gwinner@uni-wuerzburg.de

Axel Winkelmann
University of Wuerzburg
axel.winkelmann@uni-wuerzburg.de

Abstract

Fraud is a worldwide issue affecting almost every organization once in a time. Recent studies have shown that fraudulent behavior impacts up to 5 % of a companies annual revenue. Information systems (IS) have become an integral part of every modern organization. They contain the data foundation of the entire company and thereby supporting business processes and day-to-day transactions. Although an IS usually contains control mechanisms to prevent different kinds of fraud, these mechanisms look insufficient, considering the role of IS in many fraud cases. Since many cases from different companies have shown the need for an appropriate countermeasure, we want to develop an application that efficiently detects fraud and fraudulent behavior. Therefore, we conducted a structured literature review and a qualitative survey to apply the design science research (DSR) methodology and derive requirements for a fraud detection system (FDS). As a result, we present a meta-model for a FDS for enterprise resource planning (ERP) systems. We also provide application requirements, principles, and features that define areas for further research.

1. Introduction

Fraud is a worldwide issue, the so-called ”internal fraud” is the largest and most common among the different types of fraudulent behavior, which all together impact companies in up to 5 % of their annual revenue [1]. Although various detection approaches and systems are available, helping companies to uncover occupational fraud, insider threat, and financial fraud, they are specialized in certain types of fraud and only leverage parts of an organization's data. Many mechanisms and systems do not prevent organizations from the loss caused by fraudulent behavior because manipulations are mostly not discovered in time. Today, more and more companies worldwide have adopted a central IS to run their business [2]. Most commonly are ERP systems. Hence, they integrate all data and information of a company for supporting business processes, often from purchase to pay. In ERP systems, employees carry out day-to-day transactions such as purchasing, sales, and financial transactions [3]. As a result, they have access to the most relevant information of an enterprise. Although ERP systems are beneficial for every organization, they can also lead to internal fraudulent behavior. This results in information asymmetries between the company and its employee. Hence, firms do not entirely know employees behavior regarding the handling of data in the system. In this way, fraud, as well as manipulations, may occur. Prior research has mostly focused on either financial systems or reactive detection methods. For this reason, we are aiming to develop a FDS that uncovers anomalies and prevents fraudulent behavior in the ERP system's day-to-day transactions. To build a meta-model for a FDS, we conducted a structured literature analysis and a qualitative study to generate design principles (DP). This approach was chosen to combine a scientific as well as a practical point of view. Based on this, design features (DF) can be derived. We are especially interested in actual requirements from literature and practical point of view regarding a FDS. For this reason, we formulate the following research questions (RQs):

RQ1: What are the requirements for a system that can detect fraudulent behavior in an enterprise IS?

RQ2: How does a FDS look like from a model perspective?

To answer the RQ, we structure this paper as follows: in section 2, we first provide a brief overview of the fundamental literature regarding fraud in IS and outline our applied research method. In section 3, we integrate our findings and designing principles to build a meta-model for a FDS in section 4. After that, we evaluate our model in section 5. The last chapter concludes this paper with a summary of our findings and limitations.

URI: https://hdl.handle.net/10125/71477
978-0-9981331-4-0

2. Related Work and Research Methodology

This section gives a rough overview of the fundamental literature that relates to our work. Our work contributes to the increasing system research, as we are investigating opportunities for IS. We also aim to explore additional factors that might not have been identified in previous studies. We have chosen a qualitative approach for the study to obtain confirmatory and explanatory data, following the methodological guidelines by [4]. To this end, we are adapting a design-oriented research method and adjust the development process of our meta-model following the guidelines by [5]. Figure 1 shows the applied methodology we attend for our paper. The design cycles were adopted from a theory developed by DSR [6] and implemented along the suggestions of one of DSR approaches in IS research [5]. To develop a FDS for ERP systems, as a first step, we focus in this work on the model perspective.

2.1. Literature Foundation

Organizations can face different kinds of fraud. However, in this paper, we focus on occupational fraud and therefore use the ACFE (Association of Certified Fraud Examiners) definition of fraud: "Occupational fraud is defined as the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets" [1]. Since a literature review about the current state of fraud detection in IS did not exist, we first conducted a structured literature analysis following the established methodological guidelines by [7] and [8]. Because the quality of the literature is imminent for the results, we followed the recommendation by [9]. We focused on reviewed articles published until April 2020 in outlets ranked (A+ to C) in the VHB-JOURQUAL3. We have limited our search to the database "Web of Science" and "AISeL" because they combine all relevant Journals and Conference Proceedings regarding IS research. Therefore, we used the keywords information system, enterprise system, enterprise resource planning, and erp in combination with fraud, scam, con, audit, cheat, fake, manipulation, and anomaly. After forward and backward search, we came up with a total of 70 relevant papers and structured them using a concept matrix according to [8].

For this work, we shortened the outcome to a more constructive form. The identified papers were classified into three categories: systems, methodology, and level of activity. We defined these categories in advance to find out which information systems are common, what tools and methods are already examined in the scientific research, and the time level the applications are operating.

In the analysis and synthesis of the contributions, we have defined subgroups for the categories to classify the contributions in a more differentiated way. As we want to specialize in ERP systems in further research, we have subdivided the category "Systems" again. For this purpose, we have created the categories "ERP System", "Other Systems", and "Financial Systems", as it has been shown that many papers in the literature deal with financial systems and credit card fraud.

Furthermore, we wanted to determine which methods the literature deals with to check in our qualitative study, whether these are also applied in practice. By analyzing the papers, we were able to identify six different applications, methods, and procedures: organizational, process mining, rule-based, statistical, data mining, and machine learning. "Organizational" means that no direct implementation for a system is implemented, but for example, a framework developed or a study realized [10]. The category "Process Mining" includes papers that monitor and optimize business processes [11]. In "Rule-Based" approaches, actions are triggered when defined conditions occur [12]. For example, in "Statistical" implementations, fraud is concluded based on deviations from the majority or abnormal behavior [13]. With the help of "Data Mining", large amounts of data can be efficiently analyzed and evaluated [14]. "Machine Learning" allows supervised and unsupervised algorithms to be used on existing data to make predictions [15].

Previous surveys also evaluated the level of activity and the timing of the analysis of the data [16]. The high damage potential and the long-term damage caused by manipulation, continuous evaluation, and analysis of the processes are desirable to avoid high financial losses. Therefore, we have divided the temporal criteria into "Reactive" and "Real-Time" (continuous) monitoring. In doing so, we figured out that there is no research in fraud detection applications for ERP systems operating in real-time. Table 1 shows the number of papers that have been classified into the appropriate section of the concept matrix. The literature by now focuses more on organizational fraud prevention, e.g., Segregation of Duties (SoD), or audits of the annual financial statements and not on the systemic or methodical topic of fraud detection. The literature analysis results are used in section 4 to discuss and enrich the findings of our study and conceptualize the DP.

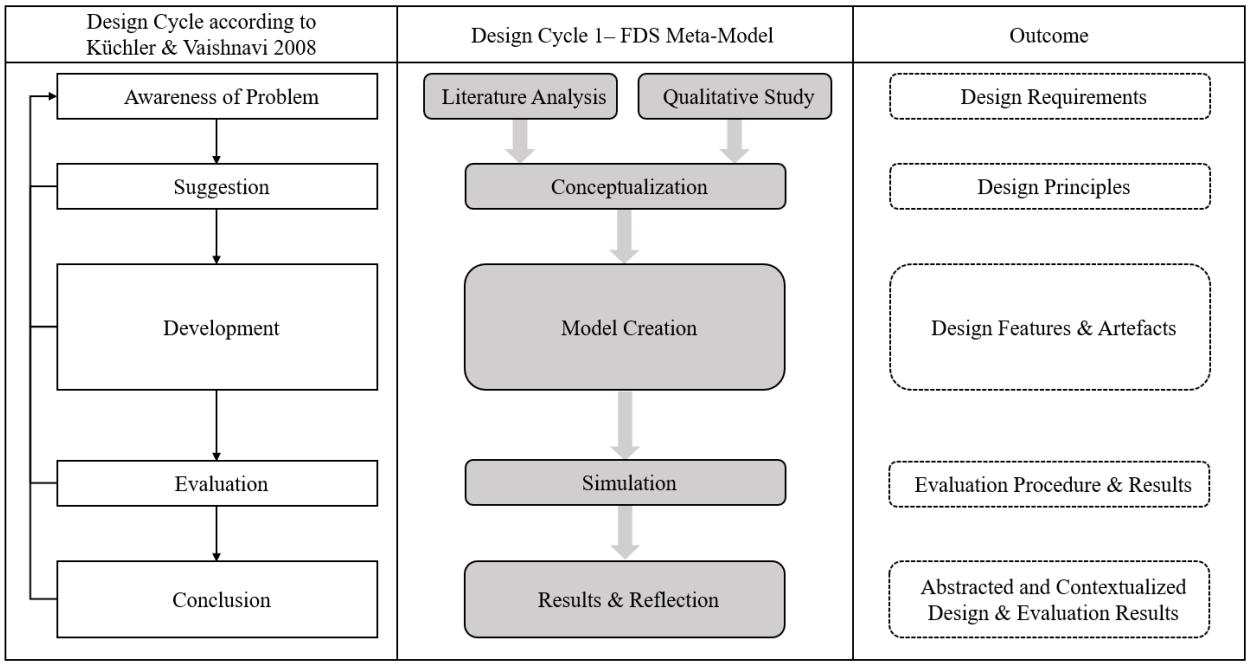

Figure 1. Research Method

Table 1. Concept Matrix

	Methods						Systems			Activity Level	
Total number of papers	Organizational / Sociological	Process Mining	Rule-Based	Statistics	Data Mining	Machine Learning	ERP Systems	Financial Systems / Credit Card	Other Systems	Reactive	Real-Time / Continuous
70	8	7	9	23	11	17	15	39	20	48	11

2.2. Methodology

The study aimed to evaluate how companies handle fraud in their IS and determine the requirements for a FDS in ERP systems. We also aim to explore additional factors that might not have been identified in previous studies. We have chosen a qualitative approach for the study to obtain confirmatory and explanatory data, following the methodological guidelines by [4]. We based our study on a broad selection of companies and a representative selection of participants. The selection of experts was made by applying the following two criteria: (1) the participant had to have comprehensive knowledge about the IS, being used by the company, and (2) be in a job position that allows being responsible for security. The interview guide is semi-structured and focused on open questions regarding necessary information about the firm and the interviewee, the used systems, and actual challenges regarding fraudulent behavior [17]. For reasons of space, the interview guide is not part of the paper at hand. The guide was tested and adjusted by performing three pretests. Based on this, interviews were conducted by phone from January until March 2020 with nine experts (I1-I9). The interviews were conducted until the so-called theoretical saturation is reached so that more interviews do not gain new insights [18]. All interviews were recorded upon approval by the interviewee and fully transcribed

afterward. To gain actual insights, the raw material must be analyzed. The data interpretation includes the coding process, which was performed by using the data analysis software MAXQDA. Codes lead to keywords out of the relevant information from the interviews. We then analyzed the statements for matching synonyms and generalized the statements into the requirements R1-R11, shown in Table 2 below.

3. Conception of Design Principles and Design Features

To answer our RQs, we used a DSR approach, following the guidelines by [5] and [19]. It consists of six iterative steps, including problem identification, the definition of objectives, design and development, demonstration, evaluation, and communication. The design cycles were adopted from [6] and implemented along with the suggestions of [5]. To develop a FDS for ERP systems, as a first step, we focus in this work on the model perspective. As we want to describe what the FDS should enable users to do and how it should be built, we focus on the category of action- and materiality-oriented DP. To develop the DP, we use the requirements generated through the interviews' coding process in the analysis phase (R1-R11). The requirements "Analyze a large amount of data" and "Immediate fraud detection", for example, became the design principle DP2 "Real-time data analytics". The merging of the requirements and the insights gained through the literature analysis process results in DP1-DP6. For a better overview, we conceptualized all statements in Figure 2 and described their derivations as follows:

DP 1: The FDS must be easy to integrate into the existing system infrastructure of the company. The study has shown that different types of IS are frequently used in practice. According to the respondents, almost all relevant transactions are carried out by their ERP system. The respondents named SAP, Godesys, Microsoft, as well as in-house developments as systems in use. Since fraud can be found in all organizational divisions and processes, the solution should not only focus on a specific fraud scenario or a single business area (R6). The amount of data (R2) processed within a company's ERP system can easily reach terabytes. From the technical, financial, and organizational point of view, the FDS should be easy to integrate and directly connected to the company's existing system infrastructure.

DP 2: The FDS should be able to prevent fraud in real-time. Half of the interviewees (I1, I2, I5, I8) mentioned that their companies have fully implemented a Three Lines of Defense Model where each of the three departments has a predefined specific role in risk management, including a separated internal audit department that also has the assignment of fraud detection. The study's findings on companies' methods to detect fraudulent behavior or prevent fraud show that, except for risk management, employee training, and role or authorization management, they are all an organizational nature or used monthly or even longer-term basis. DP2 "Real-time data analytics" is based on the requirements for immediate fraud detection (R8) and the large amount of data processed within ERP systems (R2). Among the respondents, I1 described that "there are currently no automated methods, only reactive". The statement of I2 "It would be quite helpful if the employee were immediately suggested to watch out, is that correct?" underlines the advantage of analyzing the data as it is entered into the system. As manipulations and fraud can lead to immense financial and reputation losses, it is crucial to detect them immediately. From our point of view, this can be achieved by real-time data analysis. The literature review has also confirmed this DR. It has been shown that only 11 of the 70 relevant papers deal with real-time data analysis or continuous data auditing. Only four of the 11 papers include ERP systems in their analysis. For example, [12] presents an agent-based continuous audit model (ABCAM) that can process auditing in real time independently of the system. Additionally, the paper of [20] describes implementing a monitoring system for the internal audit of SAP/R3. The remaining 59 papers apply methods that require the data to be first extracted from the system to examine it on anomalies or manipulations.

This finding indicates that real-time analysis is possible but much more difficult to achieve. A more in-depth analysis of the field shows differences in the definition of real-time. If an integrated FDS needs to process the analysis between users' input and the database change, the precision and processing time is a highly relevant factor. In many of the analyzed papers, real-time is not enough substantiated by comparing the processing time between different approaches. Therefore, we conclude that the selection process for a technical approach satisfying the DP2 needs to compare the precision in finding fraudulent transactions and the false positive rate of algorithms and the processing time.

DP 3: The FDS should provide state-of-the-art algorithms and techniques. The respondents stated that currently, the data from the system is often checked manually. Most of them conduct a labor-intensive review process at regular intervals. Some also work

Table 2. Results of analyzed interviews

13*		ID	I1	I2	I3	I4	I5	I6	I7	I8	I9
ID	**Requirements**	**Ind. & Employees**	Fashion <7 k	Aerospace <10 k	Engineering <10 k	IT Service < 500	Consulting < 300 k	IT Service < 100	Production < 2.5 k	Banking < 7.5k	Retail < 100
R1	Support complex fraud scenarios (e.g. group levering SoD)			X			X			X	
R2	Analyze a large amount of data		X				X			X	X
R3	Detect procedural changes & process deviation		X			X					
R4	Support of document falsification		X				X				
R5	Detection of outliers in values (e.g. invoice total, posting total, discounts)			X	X						
R6	Integration into all ERP functions / modules		X				X	X	X	X	
R7	Detection of data-theft (e.g. intellectual property) modules								X		
R8	Immediate fraud detection (e.g. real-time)			X						X	X
R9	Support of unknown scenarios in addition to pattern recognition or checksums			X		X	X			X	
R10	Learning, adaptable & intelligent logic		X				X				
R11	Integration of human experience for a better & more understandable system			X							X

together with external service providers for this purpose. According to the interviewees, this process is not, practical and human check-ups often miss anomalies or manipulations of data. Therefore, DP3 aims for techniques that are not as static as used reporting tools are and do not only checking fixed values. Different kinds of fraud and data that need to be supported by a FDS (R3-R5, R7) and the requirement (R9) to support previously unknown fraud scenarios demonstrate the need for an adaptable or intelligent logic (R10). Technically there are several methods and algorithms tested to obtain the best results for the FDS. As our literature analysis shows, different methods of fraud detection have been tested on several data sets. From rule-based and statistical algorithms to data mining techniques, e.g., clustering, classification, regression, and outlier detection, to machine learning models, like Support Vector Machines (SVM), Artificial Neural Networks (ANN), Bayesian Networks or Genetic Algorithms [13, 21, 22, 23, 24, 25]. Most of the methods we have found through our literature analysis are applied in credit card fraud detection. Many financial applications are either based on training data sets, often with synthesized credit card transactions, or the data extraction system was not explicitly mentioned. In the field of fraud detection in ERP systems, the literature review shows that only rule-based approaches and process or data mining methods were proposed or tested [12, 23, 26, 27]. It also shows that new methods, based on artificial intelligence or machine learning, are not applied in fraud detection in ERP systems.

DP 4: Combination of Process and Data Mining. In the last segment of the study, the participants were asked about methods, systems, and other measures to prevent fraudulent behavior or to detect manipulations in their ERP systems. The findings show that all companies, except I4, I6, and I9, have explicitly mentioned that they use SoD. There are two instances required to complete a task for critical processes (e.g., financial payments). I1, I5, I8 refer to documentation obligation for critical but rare processes. The qualitative survey also provided many examples of fraud that often do not consist of a single act. Since many common ERP systems are transaction-based, the execution of processes is happening using several actions (transactions) that map the individual process steps. Therefore, fraud cases and the concealment of these can extend over several process steps and

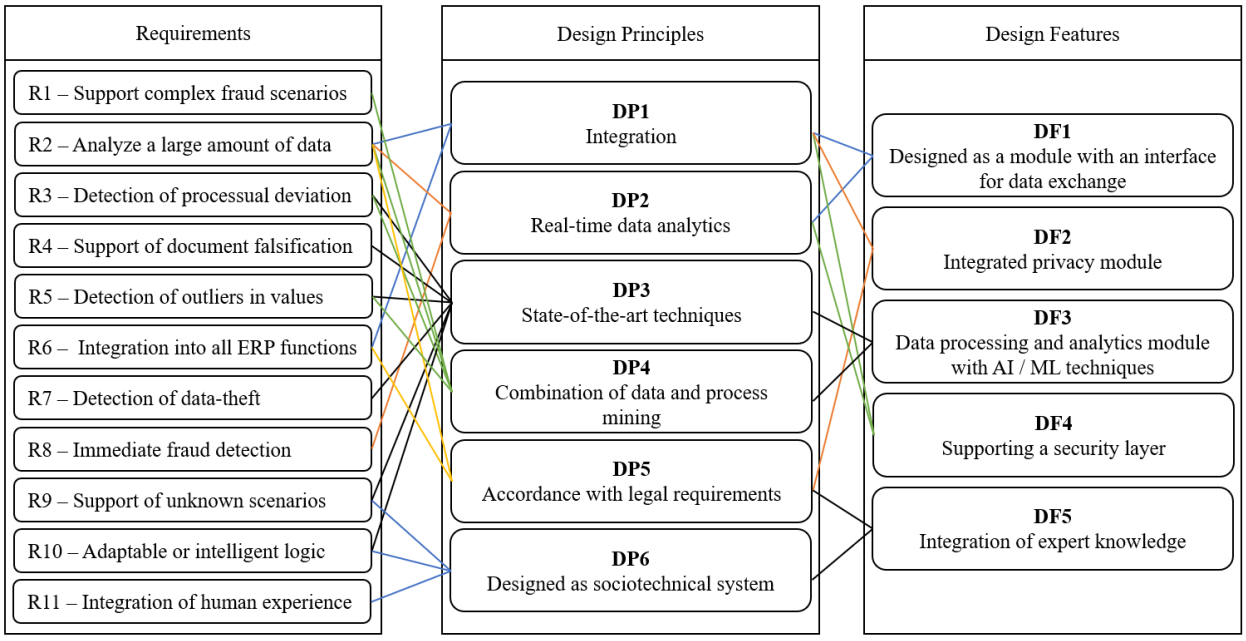

Figure 2. Results of the DSR Approach

do not always occur in only one process step or transaction. DP4 combines the requirements for an FDS that supports complex scenarios involving multiple persons and activities (R1) and needs to not only detect outliers in single values or totals (R2, R5) but also detect process deviations (R3). Supporting fraud detection over several process activities concludes that the data perspective per se is not enough. The FDS should also use the process perspective to detect anomalies in both data and process deviations. DP4 has a direct relationship to DP3 but focuses on data and less on methods or techniques to be used. The FDS can use a combination of process mining and data mining to detect outliers in a single data set and process instances. DP4 would further benefit from explainable or interpretable machine learning algorithms [28], since understanding the detection provides knowledge that can be utilized in a human-in-the-loop approach (see DP6). Although we found process-based approaches to fraud detection in the literature, for example, [23] and [11], we could not identify any publication that used a mixed methodological approach.

DP 5: The FDS should accord with legal requirements and laws. In the study, several interviewees stated that fraud could be found in all business areas and all pay groups. Therefore, it is implied that a future FDS needs to incorporate all the used functions of an ERP system (R6). Modern ERP systems manage almost all company business areas, from purchasing, sales, and administrative processes. The FDS is processing highly sensitive data, such as supplier and customer information or information about the employees itself, e.g., their health data. The processing of personal data in Germany is strictly regulated by the General Data Protection Regulation and other regulations (e.g., BetrVg), including data mining (R2) or other data processing techniques. Before processing, data must be pseudonymized or anonymized, considering every individual's interests and comply with all regulations [11, 29].

DP 6: The FDS should be designed as socio-technical system. The FDS should also support new fraud scenarios (R09) and be adaptable to them (R10). It should be clear that such a solution cannot be the last instance in the decision-making process. DP6 strongly suggests designing the FDS as a socio-technical system, where the human and computer can complement each with a human-in-the-loop approach [30, 31]. This DP is also supported by the immediate requirement always to integrate human experience (R11). In the literature analysis, we found only one paper that included the human factor in the process [6], located in the continuous auditing field.

4. Architectural Model for a FDS

Based on the six identified DP we conceptualized five DF for a FDS in ERP systems. We then developed a meta-model as an architectural framework for future FDS around these five DF. The following Figure 3

summarizes the resulting model for a FDS in ERP systems.

Since we aim to develop an adaptive and intelligent FDS, which can process large amounts of data, a simple way to extract data from the ERP system's database into the FDS is necessary. To integrate the FDS easily way into different IT landscapes and work with various ERP systems, where each has its data model, we added an ETL (extract, transform, load) module that connects the FDS with an existing ERP system. This finding leads to the first design feature, the minimization of the integration effort (DF1). ETL describes the standardized procedure for data exchange between different systems. However, it could also be used to support the initial data extraction needed for machine learning and other AI techniques to learn and create the company-specific models and neuronal networks. For creating a model, an ETL module needs to transfer a high amount of data, as a companies ERP system can store hundreds of terabytes data for a single fiscal year.

In most cases, ERP systems process highly sensitive data to address the integration aspect of DP1 and comply with the GDPR, DSGVO, and other laws in DP5. For this reason, we have added DF2 as a privacy module, which has the task to pseudonymize or anonymize all sensitive data. Therefore modern techniques for anonymizing data like k-anonymity and differential privacy and encryption can be usesd [32]. The core of the FDS is the data processing and analytics module, which combines different techniques to detect fraud or predict fraudulent behavior (DF3). The module should use techniques like data mining, process mining, artificial intelligence methods, and machine learning, which are specifically optimized for their task (e.g., detection of process variations). The data processing and analytics module combines DP3 (using state-of-the-art techniques) and DP4 (combination of data and process view on the transactions). Since enterprise systems use database tables with hundreds of columns, some unused due to the system's respective customizing, in our model, we added a pre-processing module for data cleansing, feature extraction, and generation. Regarding the algorithms, we were able to make some helpful additions to the meta-model using current literature. In contrast to credit card fraud, fraud in ERP systems is, if at all, in less than the per mill range. By comparing non-fraud and fraud transactions, the data thus is not evenly distributed at all. Special procedures (novelty / outlier detection) are required that can handle heavy unbalanced data sets [33, 34, 35]. In some IS, several transaction types with different attributes are stored in the same database table. It leads to empty columns in the table. Naturally, the normalization of the data is not given anymore. In data science, it is described in a multi-class dataset, with outliers in each class. Therefore, one-class methods are only applicable for transactions if they are classified into transaction types, subdivided, and modeled separately during pre-processing. Otherwise, neural networks or novelty-detection methods, which can work with several classes, should be used. Hence, this work does not focus on the detection and approaches to leverage. We conclude that the analytical part (DF3) should be part of further research. The security layer (DF4) is a technical term that describes the functional part of the FDS to prevent fraud by checking the input or transactional data of a user action. We could not find any scientific concepts, integrating a layer between application and database, since most enterprise systems connect these layers by reaching via a query language (e.g., SQL) from implementing the Database Management System (DBMS). In our opinion, a pure API based solution, where data is created by the user, stored in the database, and analyzed afterward, is incompatible with a real-time analysis (DP2).

As we could not find any suggestions for a possible concept, we added two conceptual ideas on how the security layer could be integrated into an ERP system. We have to state that these are only ideas that need to be conceptualized and analyzed further. If part of the application, the security layer could, for example, be constructed like other batch processing mechanisms. In some ERP systems, where the sheer number of transactions is difficult to handle, the database tables' changes are first collected in a buffer table. They will be saved to the corresponding database table(s) batch-wise, with a minimal time delay. A posting mechanism changes the data only if the needed database tables are not locked. The previously collected changes are only deleted from the buffer if changes are in the database. Otherwise, the entries remain for the next try. Such mechanisms exist in DBMS to avoid deadlocks but adopted in enterprise applications (e.g., for goods movement posts) to solve problems, especially for multi-server or distributed computing. If part of the database, an already existing possible entry point for such a solution approach could be an additional layer on top of the database, analogous to SAP in the HANA database added core data service views[1]. With these, it is possible to execute custom code between querying and delivering data to and from the database. Hence, an additional layer's conception is not trivial and needs to be coordinated with the company developing the database. We see the conception and implementation of the security layer as a task for the ERP system's software

[1]https://cap.cloud.sap/docs/cds/

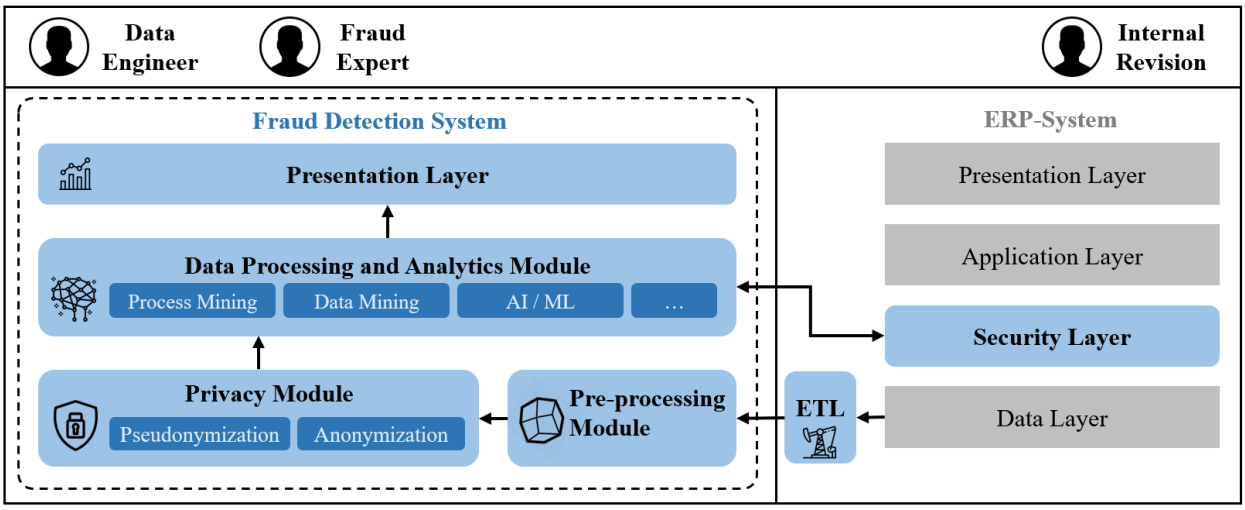

Figure 3. Meta-Model for the proposed FDS

manufacturer and not for companies using the system.

In order to design the FDS as a socio-technical system (DP6) that can handle human and technology interactions, we use expert knowledge for different purposes (DF5). Data engineers should always monitor the generated AI models and trained neuronal networks to avoid failures and bias in the fraud detection process. In system changes (e.g., data model, or processual changes), the initially created machine learning models and artificial intelligence networks need to be updated continuously. For supervised fraud detection approaches, experts should identify and monitor the sample data and sample size, as these factors are directly relevant to such a technique. An impartial person or department should also examine disputed cases. As large organizations usually have an internal audit department, the task should either be located or coordinated from that team of employees. Smaller companies, on the other hand, must make use of direct superiors to clear uncertain cases.

5. Evaluation

We are aware that evaluating the proposed model is very important in the DSR process. We have decided to conduct our model to a theoretical evaluation. Our evaluation should by no means represent an endpoint. As described in the DSR process, we consider the model's evaluation as necessary to move on to the second design cycle. We want to collect other ideas, criticisms, and problems for the improvement of the system. In doing so, for this first design cycle, we have chosen a simulation-based evaluation. The scenario for evaluating our model is based on the workshops we held during our research at a German Tier One automotive supplier. As a serial manufacturer with over 1.500 employees, the company focuses on the production of drive-related components. The company uses SAP SE's latest ERP system, SAP S/4HANA®, to support its processes and business. The organization has been a fraud victim on several occasions, each with different scenarios in various departments.

Examples of fraud cases mentioned are in the purchasing sector, e.g., price-agreements, bonus payments, or the warehouse sector, e.g., in the stock-tacking process. We have conducted several workshops on fraud detection with senior executives and the management team when we set up the following scenario for our evaluation. In the purchasing department, a buyer agent and his supervisor teamed up to fake outgoing payments to a supplier to enrich oneself. Payment in two installments was agreed with the supplier. Therefore, the purchasing employee doubles the purchasing prices and changes after the first-rate of the supplier's account details to a third party they both know. With the first install payment, the real supplier was fully paid, and with the second rate, the employees enrich themselves. The given scenario is not easy to execute because there are several security mechanisms. For example, due to SoD, critical master data changes must always be approved from a second instance.

For this reason, the employees teamed up with their supervisor to confirm the changes in the account details. The payment needs to be inconspicuous, which is why they choose a third person as a recipient for the payment and not themselves. After we had set up the companies' scenario, it was discussed and played through with the

respective affected people in the departments. We then discussed how a system based on the proposed model should proceed with IT and management's responsible members. The following process describes our model in the scenario-based evaluation. Before the FDS can be used, it must be adapted to the existing ERP system. After installation, the first step is to transfer existing data from the database in the ERP system's data layer to the FDS by using the ETL module. Subsequently, initial data needs to be cleared to remove useless content, such as empty and inconsistent database table columns, and generalize it. The privacy module anonymizes and pseudonymizes all sensitive data. In the data processing and analytics module, the adaption and training of the FDS are taken place.

Along with data mining (e.g., clustering) and process mining, the FDS uses state-of-the-art techniques to train machine learning models on the data and processes regular to the company. During the initial training phase, the FDS needs to be monitored by a professional data engineer. In addition to that, a fraud expert should be available to assist with supervised or semi-supervised learning techniques. Outliers and anomalies detected by the FDS should be addressed to at least more than one-third-party for a more secure handling procedure. To detect fraud in the given case, the FDS, for example, could use data mining techniques on the purchasing prices concerning the bought goods or services to detect outliers in the prices. As an alternative, a deep learning algorithm could be trained on the processes and detect the process deviation in the purchasing process due to the change of the payment data or when the supervisor needs to check the changes, which was probably shorter than usual. There are several other ways to detect the given fraud case based on the options such a FDS delivers.

6. Conclusion

In our work, we developed a model for a real-time FDS in ERP systems to support companies by preventing fraudulent behavior and data manipulations and thus avoid financial losses and reputational damage. To answer RQ1, we present a dual approach consisting of a systematic literature review and a qualitative study about fraud in IS to derive requirements for a future FDS for ERP systems. Based on this, a meta-model containing the evaluated DF is proposed to answer RQ2. The literature review shows that the field of fraud detection applications in IS research receives hardly any attention. Many of the various fraud or outlier detection methods have been investigated in other contexts, like financial or credit card transactions. Also, methods and procedures to detect fraud have been tested from a theoretical perspective but not been implemented into applications or systems in practice. By finding this white spot in the current IS literature, we have evaluated our literature review results through insights from a practical point of view. Based on a qualitative study with experts from various industries and company sizes, we were able to identify the current state of fraud detection in practice. By combining theory and practice, we derived requirements for a FDS in IS (RQ1). We integrate the findings to develop a meta-model that includes the DF we have identified through our DSR approach (RQ2).

However, approaches of this kind have a variety of well-known limitations that should be considered. First, the underlying literature review is mainly in the representative design, since we have limited our analysis to the A+ to C ranked IS Journals in the VHB-JOURQUAL3. This limitation leads to a limited number of publications that were used for the analysis. Second, our qualitative study is limited to only nine participants in Germany. This limitation can be addressed in the future by conducting further surveys in other countries. A more extensive study with more participants is necessary to provide the validity and reliability of our findings. Third, the resulting meta-model is preliminary and only provides an initial conceptualization of the FDS. In future studies, we seek to implement, evaluate, and refine the prototype in a real scenario. Further research steps must incorporate more companies and involve testing the developed FDS using a real data set from an organization's ERP system.

7. Acknowledgements

This study is based upon work funded by the German Federal Ministry of Education and Research (BMBF) within the program "ICT 2020 – Research for Innovations", funding number (01IS18045A) and implemented by the German Aerospace Center (DLR).

References

[1] ACFE, "Report to the Nations 2018 Global study on occupational fraud and abuse," *Report To the Nations*, 2018.

[2] Statistisches Bundesamt, "Nutzung von informations- und kommunikationstechnologien in unternehmen," vol. 49, pp. 1–64, 2017.

[3] R. Thome and A. Winkelmann, *Grundzüge der Wirtschaftsinformatik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015.

[4] A. Bryman, E. Bell, and B. Harley, *Business Research methods*. Oxford university press, 2018.

[5] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[6] B. Kuechler and V. Vaishnavi, "On theory development in design science research: anatomy of a research project," *European Journal of Information Systems*, vol. 17, no. 5, pp. 489–504, 2008.

[7] J. vom Brocke, A. Simons, B. Niehaves, K. Riemer, R. Plattfaut, and A. Cleven, "Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process," in *17th European Conference on Information Systems*, vol. 9, pp. 2206–2217, 2009.

[8] J. Webster and R. T. Watson, "Analyzing the Past to Prepare for the Future: Writing a Literature Review," *MIS Quarterly*, vol. 26, no. 2, pp. xiii – xxiii, 2002.

[9] J. Rowley and F. Slack, "Conducting a literature review," *Management Research News*, vol. 27, no. 6, pp. 31–39, 2004.

[10] A. Harrison, B. Mennecke, and W. Dilla, "An empirical study of a two-sided model of fraudulent exchange," 2012.

[11] S. Hoyer, H. Zakhariya, T. Sandner, and M. H. Breitner, "Fraud prediction and the human factor: An approach to include human behavior in an automated fraud audit," *Proceedings of the Annual Hawaii International Conference on System Sciences*, pp. 2382–2391, 2012.

[12] C. L.-Y. Chou, T. Du, and V. S. Lai, "Continuous auditing with a multi-agent system," *Decision Support Systems*, vol. 42, no. 4, pp. 2274–2292, 2007.

[13] H. D. Kuna, R. García-Martinez, and F. R. Villatoro, "Outlier detection in audit logs for application systems," *Information Systems*, vol. 44, pp. 22–33, 2014.

[14] P. K. Chan, W. Fan, A. L. Prodromidis, and S. J. Stolfo, "Distributed Data Mining in Credit Card Fraud Detection," *IEEE Intelligent Systems and Their Applications*, vol. 14, no. 6, pp. 67–74, 1999.

[15] D. B. Prakash, "ATM Card Fraud Detection System using Machine Learning Techniques," *International Journal for Research in Applied Science and Engineering Technology*, vol. 6, no. 4, pp. 5124–5129, 2018.

[16] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.

[17] G. Paré, "Investigating Information Systems with Positivist Case Research," *Communications of the Association for Information Systems*, vol. 13, 2004.

[18] J. M. Corbin and A. Strauss, "Grounded theory research: Procedures, canons, and evaluative criteria," *Qualitative Sociology*, vol. 13, no. 1, pp. 3–21, 1990.

[19] L. Chandra, S. Seidel, and S. Gregor, "Prescriptive knowledge in IS research: Conceptualizing design principles in terms of materiality, action, and boundary conditions," *Proceedings of the Annual Hawaii International Conference on System Sciences*, pp. 4039–4048, 2015.

[20] M. Alles, G. Brennan, A. Kogan, and M. A. Vasarhelyi, "Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at Siemens," *International Journal of Accounting Information Systems*, vol. 7, no. 2, pp. 137–161, 2006.

[21] M. J. Kim and T. S. Kim, "A neural classifier with fraud density map for effective credit card fraud detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2412, pp. 378–383, 2002.

[22] P. Kanhere and H. K. Khanuja, "A methodology for outlier detection in audit logs for financial transactions," *Proceedings - 1st International Conference on Computing, Communication, Control and Automation, ICCUBEA 2015*, pp. 837–840, 2015.

[23] M. Jans, M. Alles, and M. Vasarhelyi, "Process Mining of Event Logs in Auditing: A Field Study of Procurement at a Global Bank," *The Accounting Review*, vol. 89, no. 5, pp. 1751–1773, 2012.

[24] F. Benmakrouha, C. Hespel, and E. Monnier, "An algorithm for rule selection on fuzzy rule-based systems applied to the treatment of diabetics and detection of fraud in electronic payment," *2010 IEEE World Congress on Computational Intelligence, WCCI 2010*, 2010.

[25] Q. Deng and G. Mei, "Combining self-organizing map and k-means clustering for detecting fraudulent financial statements," *2009 IEEE International Conference on Granular Computing, GRC 2009*, pp. 126–131, 2009.

[26] S. I. Chang, D. C. Yen, I. C. Chang, and D. Jan, "Internal control framework for a compliant ERP system," *Information and Management*, vol. 51, no. 2, pp. 187–205, 2014.

[27] S. M. Huang, D. C. Yen, Y. C. Hung, Y. J. Zhou, and J. S. Hua, "A business process gap detecting mechanism between information system process flow and internal control flow," *Decision Support Systems*, vol. 47, no. 4, pp. 436–454, 2009.

[28] I. Sample, "Computer says no: why making ais fair, accountable and transparent is crucial," *The Guardian*, vol. 5, pp. 1–15, 2017.

[29] L. Strous, "Audit of information systems: The need for cooperation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1521, pp. 264–274, 1998.

[30] W. Karwowski, *International Encyclopedia of Ergonomics and Human Factors-3 Volume Set*. CRC Press, 2006.

[31] L. Edwards and M. Veale, "Slave to the algorithm: Why a right to an explanation is probably not the remedy you are looking for," *Duke L. & Tech. Rev.*, vol. 16, p. 18, 2017.

[32] I. Mironov, O. Pandey, O. Reingold, and S. Vadhan, "Computational differential privacy," in *Annual International Cryptology Conference*, pp. 126–142, Springer, 2009.

[33] M. Schreyer, T. Sattarov, D. Borth, A. Dengel, and B. Reimer, "Detection of anomalies in large scale accounting data using deep autoencoder networks," *arXiv preprint arXiv:1709.05254*, 2017.

[34] M. Schreyer, T. Sattarov, C. Schulze, B. Reimer, and D. Borth, "Detection of accounting anomalies in the latent space using adversarial autoencoder neural networks," *arXiv preprint arXiv:1908.00734*, 2019.

[35] A. Pumsirirat and L. Yan, "Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine," *International Journal of advanced computer science and applications*, vol. 9, no. 1, pp. 18–25, 2018.