# A Case Study on the Suitability of Process Mining to Produce Current-State RBAC Models

Maria Leitner[1], Anne Baumgrass[2], Sigrid Schefer-Wenzl[2],
Stefanie Rinderle-Ma[1], and Mark Strembeck[2]

[1] University of Vienna, Austria
Faculty of Computer Science
{maria.leitner,stefanie.rinderle-ma}@univie.ac.at
[2] Vienna University of Economics and Business (WU Vienna), Austria
Institute for Information Systems, New Media Lab
{firstname.lastname}@wu.ac.at

**Abstract.** Role-based access control (RBAC) is commonly used to implement authorization procedures in Process-aware information systems (PAIS). Process mining refers to a bundle of algorithms that typically discover process models from event log data produced during the execution of real-world processes. Beyond pure control flow mining, some techniques focus on the discovery of organizational information from event logs. However, a systematic analysis and comparison of these approaches with respect to their suitability for mining RBAC models is still missing. This paper works towards filling this gap and provides a first guidance for applying mining techniques for deriving RBAC models.

**Key words:** Process Mining, RBAC, Security in Business Processes

## 1 Introduction

Process-aware information systems (PAIS) support the execution of tasks in business processes and store so called "event log" files (e.g., [10]). In this context, process mining techniques are used to analyze and extract process-related information from event logs. In general, process mining techniques do not directly focus on the derivation of access control models. However, such models are an important means to define which subject is permitted to execute certain tasks (e.g., [3, 9]).

In recent years, role-based access control (RBAC) (e.g., [3]) has developed into a de facto standard for access control in both, research and industry. In RBAC, roles correspond to different job-positions and scopes of duty within a particular organization or information system. Access permissions are assigned to roles according to the duties this role has to accomplish, and subjects (e.g., human users) are assigned to roles. In the business process context, RBAC has been extended to consider access permissions for tasks included in a business process (e.g., [9]).

This paper investigates into the applicability of three different process mining approaches and one role derivation approach to extract RBAC information from event logs. In particular, we aim to provide an initial decision guidance on which of these approaches can be applied in a particular context and which prerequisites are necessary to retrieve proper results. For this purpose, we conducted a case study where we analyzed an event log of a real-life business process from the university context.

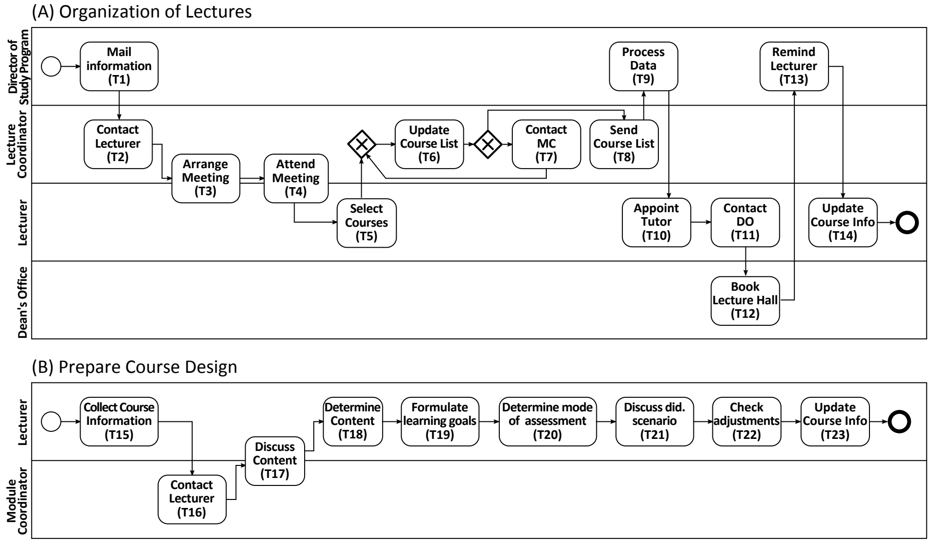
The remainder of this paper is structured as follows. Section 2 presents an overview of the four different approaches used in our case study, our running example, and the results of the four approaches. Next, these results are discussed and evaluated in Section 3. Finally, Section 4 concludes the paper.

## 2 Case Study

In the context of PAIS, event logs store information that can be used to produce so called *current-state RBAC models* (see, e.g, [1, 2, 4, 6, 7]). In particular, **role derivation** approaches automatically derive a current-state RBAC model from event logs and precisely reflect how subjects performed tasks in PAIS (see, e.g., [1]). In [1], we developed a derivation component that is able to produce a current-state RBAC model. Based on the results of this derivation, we can conduct a refinement via the role engineering tool xoRET which detects and combines roles with (partially) identical permissions [8].

Furthermore, process mining approaches can also be applied to derive current-state RBAC models (see, e.g, [4, 6, 7]). The resulting RBAC models provide an abstraction of the information contained in an event log. Using ProM 5.2 [11] and its plugin for **organizational mining**, we are able to extract and represent organizational structures via organizational models (e.g., [7]). In organizational models, subjects with a similar frequency of performed tasks are grouped into organizational entities. Thereby, these models provide information on the relationship between the organizational entities and the tasks assigned to the subjects of these entities. Thus, they can be used to build a current-state RBAC model. Similar, a **role hierarchy miner** [6] is able to identify groups of subjects performing similar tasks and, in addition, to identify hierarchical structures between the groups of subjects. In this case, these hierarchical structures can be used to build a current-state RBAC model including a role hierarchy. Moreover, **staff assignment mining** aims to discover assignment rules from event log files (e.g., [4]). We apply staff assignment mining using corresponding organizational information (see Section 2.1). As a result, the staff assignment rules identify the set of subjects who are allowed to perform certain tasks based on a combination of properties (e.g., roles, organizational units, or abilities of a subject).

In summary, we use the prototypical derivation component introduced in [1] and xoRET [8] to apply role derivation, while ProM 5.2 [11] is used to apply mining techniques in this case study.


(A) Organization of Lectures

(B) Prepare Course Design

**Fig. 1.** Process *Prepare Lectures*

### 2.1 Running Example

For our case study, we selected a typical teaching process from the higher education system. The process is divided into two subprocesses. As shown in Fig. 1, the first subprocess models the organization and assignment of lectures to the faculty. The second subprocess shown in Fig. 1 contains the preparation and course design of the lectures. As illustrated in Fig. 1, the roles *Director of the Study Program* (DSP), *Lecturer* (L), *Lecture Coordinator* (LC), *Module Coordinator* (MC), and *Dean's Office* (DO) are involved in the teaching process. In the context of RBAC, each role has a set of permissions. Thus, a permission defines that a subject having a particular role is authorized to execute a specific task. For example, a *DSP* has the permission to execute the task `Mail Information (T1)` (see Fig. 1).

Based on this running example, we used CPN Tools [5] to generate an event log including 100 cases. Furthermore, this log contains 11 subjects performing these tasks. In Section 3, we use this event log to assess the suitability of selected approaches to produce RBAC models from event logs.

### 2.2 Results

Fig. 2 shows the original role model on the left hand side surrounded by a gray rectangle. The other models from Fig. 2 show the results of the role derivation, role hierarchy mining, organizational mining, and staff assignment mining approaches applied in our case study, respectively. For each approach, the differences compared to the original model are displayed in grey-shaded areas. In the following section, these outcomes serve as basis to evaluate the suitability of each approach to produce a current-state RBAC model.

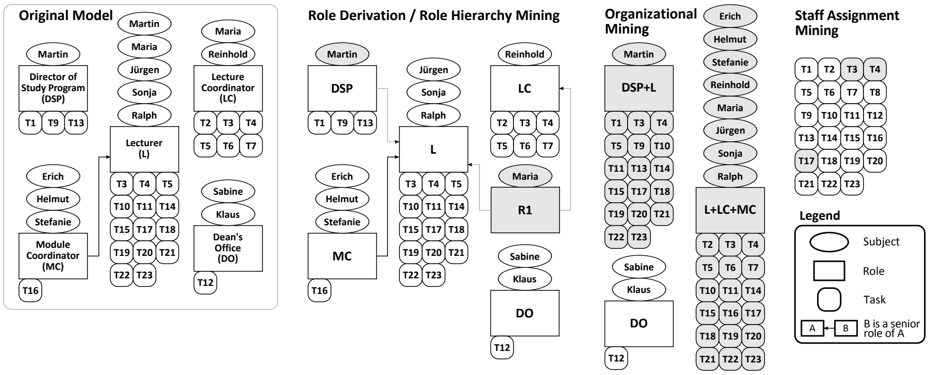


**Fig. 2.** Overview of Results: Role Models

## 3 Evaluation and Discussion

In this section, we compare the results from the four approaches with the original model (see Fig. 2). In fact, we investigate if all techniques can identify the same roles compared to the running example, reveal differences or similarities, and examine how the results of the approaches are suitable to generate RBAC models.

For the first step in our evaluation, we identified three essential issues within the results that we discuss in the following:

- **Discovering original roles:** In our case study, most of the roles were identified by all approaches. In addition, role derivation and role hierarchy mining provide a role hierarchy that covers all task-to-role assignments using less role-to-subject assignments than in the original model. Furthermore, these two approaches generated an additional role *R1* which is the accumulation of the roles $L$ and $LC$. This role is assigned to its tasks via inheritance relations in the role hierarchy. As a customization towards the original model, the role *R1* may be removed and the related subject can be assigned to the other two roles.
- **Unidentified roles:** Mostly, we were able to obtain all roles that are required to perform the teaching process of our case study (see Section 2.1). Yet, domain knowledge is required to define reasonable threshold levels for organizational mining. Depending on these threshold values certain roles may not be identified ($L$, $LC$, and $MC$). Other approaches (e.g., [12]) propose that in systems with existing organizational and role models, those roles which were unidentified by mining techniques can potentially be eliminated from the model; this can apply to roles which are scarcely used and do not provide enough administrative benefits.
- **Frequency of executions:** Results may vary if the techniques consider the frequency of task executions. For example, role hierarchy mining techniques can be applied considering different frequencies of executions. In contrast, role derivation excludes the frequency and also establishes roles with rarely used task sets. However, these roles can be customized and further evaluated.

**Table 1.** *Quantitative Measures of Results*

| | Original Model | Role Derivation | Role Hierarchy | Staff Assignment | Organizational Mining |
|---|---|---|---|---|---|
| Characteristics | | | | | |
| Roles | 10 (5 relevant) | 6 | 6 | - | 3 |
| Organizational Units | 11 (2 relevant) | - | - | - | - |
| Role-to-Subject Assignments | 13 | 11 | 11 | - | 11 |
| Task-to-Role Assignments | 25 | 25 | 25 | 23 | 36 |
| Comparison to Original Model | | | | | |
| Roles exactly identified | - | 5 | 5 | - | 1 |
| Role-to-Subject Assignments | - | 11 | 11 | - | 11 |
| Task-to-Role Assignments* | - | 25 | 25 | 20** | 1 |
| Accuracy (acc) | - | 100% | 100% | - | 20% |
| Coverage (cov) | - | 100% | 100% | 80% | 4% |

* covered by discovered (exactly identified) roles

** staff assignment rules matched by (exactly identified) task-to-role assignments

In a next step, we compare the results from the techniques used in this paper. Table 1 documents the discovered roles, organizational units, role-to-subject assignments, and task-to-role assignments for each technique. Furthermore, it shows that role derivation and role hierarchy mining were able to identify most of the original roles. For all techniques, there is a similar number of role-to-subject and task-to-role assignment relations.

Further, we examine the roles which were exactly identified, the role-to-subject assignments, the tasks of subjects covered by the exactly identified roles, the accuracy (acc), and the coverage (cov) of each technique. Therefore, we adapted the quantitative measures for accuracy and role coverage from [12]:

$$\text{acc} = \frac{\text{no. of roles identified exactly}}{\text{no. of roles in original model}}$$

$$\text{cov} = \frac{\text{no. task-to-role assignments covered by discovered roles}}{\text{no. task-to-role assignments in original model}}$$

In our case study, Table 1 shows that role derivation and role hierarchy mining have the highest accuracy and coverage of all tested techniques for our running example. Hence, the two methods are the most suitable techniques and their results can be used as basis to build RBAC models. With a few additional customizations, these two models can be tailored to the original model. In case role and organizational models exist, staff assignment mining is the most suitable technique to establish task-to-role assignment relations. Furthermore, we revealed that domain knowledge is essential to generate suitable roles via organizational mining. Without the knowledge and definition of thresholds it was difficult to obtain roles and assignment relations similar to the original model.

## 4 Conclusion

In this paper, we evaluated four approaches, namely role derivation, role hierarchy mining, organizational mining, and staff assignment mining, in order to obtain access control information from event logs. We applied these four techniques in a case study on a typical teaching process from the higher education system. First, we compared the models derived via the four techniques to the original model. Most of these techniques were able to identify the roles and tasks from the original model. Then, we evaluated the results with respect to similarities and differences and further examined if the results are suitable candidates for RBAC models. In future work, we will examine mining techniques for deriving RBAC models using more enhanced processes and corresponding event logs to determinate if we can obtain similar results.

**Acknowledgements** The authors cordially thank Alexander Brandl for modeling the use case in CPN tools.

## References

1. Baumgrass, A.: Deriving Current-State RBAC Models from Event Logs. In: Proc. of the 6th International Conference on Availability, Reliability and Security (ARES). IEEE Computer Society (2011)
2. Baumgrass, A., Schefer-Wenzl, S., Strembeck, M.: Deriving Process-Related RBAC Models from Process Execution Histories. In: Proc. of the 2012 IEEE 36th Int. Conference on Computer Software and Applications Workshops. IEEE Computer Society (2012)
3. Ferraiolo, D.F., Kuhn, R.D., Chandramouli, R.: Role-Based Access Control, Second Edition. Artech House, Inc., Norwood, MA, USA (2007)
4. Ly, L., Rinderle, S., Dadam, P., Reichert, M.: Mining Staff Assignment Rules from Event-Based Data. In: Bussler, C., Haller, A. (eds.) Business Process Management Workshops. Lecture Notes in Computer Science, vol. 3812, pp. 177–190. Springer (2006)
5. de Medeiros, A.K.A., Günther, C.W.: Process Mining: Using CPN Tools to Create Test Logs for Mining Algorithms. In: Proc. of the 6th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools (2005)
6. de Medeiros, A., van den Brand, P., van der Aalst, W., Weijters, T., Gaaloul, W., Pedrinaci, C.: Semantic Process Mining Tool - Final Implementation, Deliverable 6.11, Project IST 026850 SUPER (Sep 2008)
7. Song, M., van der Aalst, W.M.: Towards comprehensive support for organizational mining. Decision Support Systems 46(1) (2008)
8. Strembeck, M.: A Role Engineering Tool for Role-Based Access Control. In: Proc. of the 3rd Symposium on Requirements Engineering for Information Security (SREIS) (Aug 2005)
9. Strembeck, M., Mendling, J.: Modeling process-related RBAC models with extended UML activity models. Information and Software Technology 53(5) (2011)
10. van der Aalst, W.M.P.: Process Mining – Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
11. van der Aalst, W.M.P., Dongen, B.F.v., C. Günther, A.R., Verbeek, H.M.W., Weijters, A.J.M.M.: ProM: The Process Mining Toolkit. In: Proc. of the BPM 2009 Demonstration Track. vol. 489. CEUR-WS.org (Sep 2009)
12. Zhang, D., Ramamohanarao, K., Ebringer, T., Yann, T.: Permission Set Mining: Discovering Practical and Useful Roles. In: Proc. of the 2008 Annual Computer Security Applications Conference. IEEE Computer Society (2008)