Towards a Comprehensive Complexity Assessment of RBAC Models

Johannes Prescher[1], Sigrid Schefer-Wenzl[2], Anne Baumgrass[3], Mark Strembeck[4], and Jan Mendling[1]

[1] Institute for Information Business
Vienna University of Economics and Business (WU Vienna), Austria
[johannes.prescher|jan.mendling]@wu.ac.at
[2] Competence Center for IT-Security
Vienna University of Applied Sciences - FH Campus Wien, Austria
sigrid.schefer-wenzl@fh-campuswien.ac.at
[3] Hasso Plattner Institute for Software Systems Engineering
University of Potsdam, Potsdam, Germany
anne.baumgrass@hpi.uni-potsdam.de
[4] Institute of Information Systems, New Media Lab
Vienna University of Economics and Business (WU Vienna), Austria
mark.strembeck@wu.ac.at

Abstract. In the context of process-aware information systems, process-related RBAC models define which tasks of a business process can be performed by which subjects. Entailment constraints on tasks, such as mutual exclusion or binding constraints, are defined in such models to enforce or restrict subjects and roles to execute a particular combination of tasks. Although these constraints are an important means to assist the specification of business processes and to control the execution of workflows, they require additional checks and can make an RBAC model more difficult to understand. This paper investigates the factors that contribute to the reasoning effort required to understand a process-related RBAC model. We present a set of measures for such RBAC models. Moreover, these measures are applied to a set of different RBAC models to indicate the measures' suitability for assessing the complexity of RBAC models.

1 Introduction

In role-based access control (RBAC), roles are used to model different job positions and scopes of duty within a particular organization or within an information system. These roles are equipped with the permissions that are needed to perform their respective tasks. Human users and other active entities (subjects) are assigned to roles according to their work profile (see, e.g., [1,2]). In the context of process-aware information systems (PAIS), process-related RBAC models define which tasks of a business process can be performed by which subjects. In addition, process-related RBAC models allow for the definition of entailment constraints (such as mutual exclusion or binding constraints) on tasks (see, e.g., [3,4,5]).

However, such constraints are challenging from two perspectives. First, they require additional checks (and thereby additional computing time) when defining task-to-role, role-to-role, and role-to-subject assignments. Second, constraints make it more difficult for humans to interpret and correctly understand a corresponding RBAC model. Therefore, it is important to identify factors that help to predict the reasoning effort that is required to comprehend an RBAC model. For example, a complete and correct understanding of an RBAC model is required to define new role-to-subject assignments or to comprehend why a particular assignment is forbidden (e.g. due to a mutual exclusion constraint between two tasks).

Most existing approaches mainly focus on the computational complexity of RBAC models (see, e.g., [6,7,8]). In this paper, we complement this work and investigate the factors that contribute to the reasoning effort required to understand an RBAC model. Inspired by work in the area of software measurement, this paper provides a first step towards a set of measures for RBAC models.

The remainder of the paper is structured as follows. We collect existing approaches of related domains to measure the characteristics of RBAC models and summarise work from graph-theory, business process modelling and complexity in Section 2. In Section 3 we introduce general measures which we apply to RBAC models. We discuss algorithms which ensure the consistency of an RBAC model. In addition, we define corresponding measures that allow for the prediction of a degree of complexity. We illustrate the relevance of these measures in Section 4. Section 5 summarizes related work before Section 6 concludes the paper.

2 Background

RBAC model analysis typically focuses on two characteristics: the *size* of an RBAC model and the *structure* of the RBAC model (resulting from task-to-role, role-to-role, and role-to-subject assignments, as well as the relations between tasks resulting from mutual exclusion and binding constraints). The emphasis on these aspects is motivated by research on software and model measurement, which typically includes measures related to size, connections, and structure (see, e.g., [9,10,11,12]).

The first and most straightforward characteristic of an RBAC model is its size. The size of conceptual structures has been shown to be highly correlated with model complexity in prior research on software measurement and model metrics. For instance, process model comprehension has been found to be highly correlated with the number of elements in a process model (see, e.g, [13]). In the context of RBAC, size can be operationalized as the number of subjects, roles, and permissions/tasks in a particular RBAC model.

Second, we consider the structure resulting from the subject-to-role and task-to-role assignment relations. Structural characteristics have been investigated for both source code and graphical models (see, e.g., [9,12]). Fig. 1 shows three simple RBAC models that differ in terms of structure while having the same number of subjects and tasks. Fig. 1a) shows a model where all subjects and all tasks are assigned to the same role. Thus, every subject is allowed to execute every task. In contrast, the example from Fig. 1b) is rather restrictive. It assigns a separate role to each of the subjects, and each role owns a single task. Accordingly, subjects are only allowed to execute one specific task. Fig. 1c) depicts a model where the three tasks and subjects are assigned to two different roles. Obviously, these characteristics might have an impact on the reasoning effort related to an RBAC model, especially for real-world models which are often significantly larger than the simple examples from Fig. 1 (see, e.g., [14]).

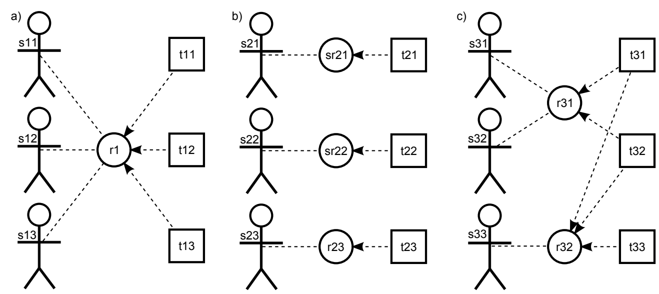

Fig. 1. Roles, subjects, and tasks

Furthermore, we consider structures resulting from role-to-role assignment relations (i.e. the role-hierarchy). Fig. 2 shows three different RBAC models. In Fig. 2a) two roles inherit permissions from other roles, in particular role *r13* inherits the tasks from *r12*, and *r12* inherits the tasks from *r11*. Role *r13* may therefore execute the tasks *t11*, *t12*, and *t13*. In contrast, Fig. 2b) depicts a flat RBAC model without role-to-role assignments (i.e. without inheritance relations between roles). However, in both models, Fig. 2a) and Fig. 2b), the three roles own the same set of permissions/tasks respectively. Such differences in the definition of a model are especially important if we have to change the corresponding model. While the assignment of a new task *t14* to role *r11* in Fig. 2a) also allows the roles *r11*-*r13* to execute this tasks, such an assignment in Fig. 2b) would lead to a significantly higher change effort as the permission/task would have to be assigned to every single role. From a visual perspective the use of role-hierarchies has two effects. On the one hand, a model may be described using fewer arcs and may therefore be more easy to comprehend (see, e.g., [13]). On the other hand, arcs representing inheritance relations carry additional semantic information. Hence, the usage of inheritance relations may also create additional reasoning effort when analyzing an RBAC model.

Another structural characteristic that we consider to assess the complexity of an RBAC model are entailment constraints. Again, we refer to research on software and model measurement that emphasizes the relevance of structure (see, e.g, [9,12]). Fig. 2c) shows a simple RBAC model with two constraints *c11* and *c12*. Depending on the type of the constraints they may affect all assignment

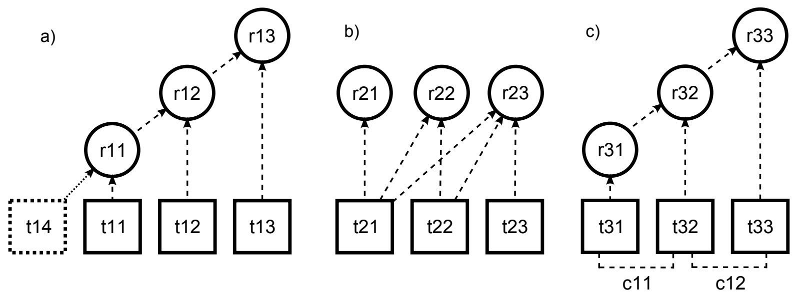

Fig. 2. Challenges in roles hierarchies and constraint relations

relations in an RBAC model (i.e. task-to-role, role-to-role, and role-to-subject assignments). In particular, the conflicts that may arise between mutual exclusion and binding constraints raise the mental effort to understand an RBAC model and to comprehend why some assignment relations must be forbidden (see, e.g., [3,15,16]). Thus, these simple examples already indicate that it is not sufficient to consider only the size and structure in a complexity assessment of RBAC models. In addition, it is necessary to understand the extent to which constraints or role-hierarchies contribute to the reasoning effort. In this manner, our work complements existing approaches presented in [6] and [17].

3 Assessment of RBAC Models' comprehensibility

- introduce general measures which we apply to RBAC models
- discuss algorithms which ensure the consistency of an RBAC model
- define corresponding measures that allow for the prediction of a degree of complexity

In this section we examine the sources of complexity for RBAC models. We summarise general measures, discuss algorithms which ensure the consistency of RBAC models and define corresponding measures to predict a degree of complexity. The term *complexity* may, depending on the context in which it is used, translate to a variety of meanings (see, e.g., [18,19]). For the purposes of this paper, and in accordance with [6], we define the complexity of RBAC models as the accumulated effort of an RBAC model's creation and maintenance. In particular, we define measures to compare the design and maintenance of several RBAC model configurations.

3.1 Definition of Measures

The complexity of RBAC models is composed of different characteristics which influence its complexity, i.e. size, structure, and model semantics. Therefore, all of these characteristics have to be considered to represent a models complexity. Consequently, we distinguish between four different measures: *counting measures* measuring size, *ratio measures* relating components to each other, *structural measures*, and *semantic measures*, each of which provided in Table 1, where $|\cdot|$ denotes the size of a set.

We assume that there is a positive correlation between the size of an RBAC model and the cognitive effort to understand it (see Table 1). First, we count the elements of an RBAC model, i.e. roles, subjects, tasks, entailment constraints, role-hierarchies as well as role-to-subject and task-to-role assignments. An RBAC model is essentially a graph in which the nodes (subjects, roles, or tasks) are connected via arcs (assignment relations or constraints). Therefore, we also include two measures that count the *number of arcs* and the *number of nodes* (see Table 1).

In addition to these counting measures, we define different ratios in the second part of Table 1. Assignment relations are used to measure the degree of linkage provided by an RBAC model ($\frac{|rsa|}{|R|}$, $\frac{|tra|}{|R|}$, see Table 1). Our hypothesis is that a stronger linkage leads to a higher complexity. Furthermore, the use of role-hierarchies relates the amount of role-to-role assignment relations (rh) to the amount of roles, i.e. connectivity between roles. Again, our hypothesis is that the comprehension decreases if that ratio is higher (positive correlation). The *tree ratio* measures the average number of relations of an RBAC element with other elements. The *constrained tasks ratio* relates the amount of tasks which are associated with at least one constraint to the number of all tasks. Our assumption for both, the tree ratio and the constrained tasks ratio, is that a model is more complex the more connected its components are. We therefore assume a positive correlation of these two measures with model complexity.

Note that the measures introduced above hardly reflect the relations between separate RBAC elements. We assume that the more relations a model includes, the more effort is required for its understanding. The role-to-role assignments form a directed acyclic graph (the role-hierarchy). In contrast, entailment constraints between tasks result in an undirected graph. Therefore, we can apply established algorithms (e.g., shortest path) from graph theory in order to derive indicators for complexity. The structural measures are subdivided into two categories to measure the structure resulting from role-to-role assignment relations and the structure resulting from entailment constraints between tasks. Adapted from graph theory, we define the maximum *distance* or *depth* within a role-hierarchy as the maximum shortest path between any two roles in the same hierarchy. Introducing this measure, we assess different designs of role-hierarchies. The *number of separate components* is the amount of independent graphs within a role-hierarchy. We assume a high connectivity (resp. less separate components) indicates a higher complexity.

In the second category of structural measures (see Table 1) we assess the complexity resulting from the entailment constraints in an RBAC model. First, the *number of constrained tasks* describes the accumulated value of all constrained tasks. It is supposed to have a positive correlation with complexity. Accordingly, the *number of unconstrained tasks* is defined as accumulated value of all unconstrained tasks. The *average number of related tasks (rt)* measure derives the

Table 1. Measurement system

Measure	Formal Definition	Hypothesis
Counting measures		
Number of elements, assignments, and relations	$\lvert R\rvert, \lvert S\rvert, \lvert T_T\rvert, \lvert sme\rvert, \lvert dme\rvert, \lvert rb\rvert, \lvert sb\rvert, \lvert rh\rvert, \lvert rsa\rvert, \lvert tra\rvert$	⇑
Number of arcs	$\lvert Arcs\rvert = \lvert sme\rvert + \lvert dme\rvert + \lvert rb\rvert + \lvert sb\rvert + \lvert rh\rvert + \lvert rsa\rvert + \lvert tra\rvert$	⇑
Number of nodes	$\lvert Nodes\rvert = \lvert R\rvert + \lvert S\rvert + \lvert T_T\rvert$	⇑
Ratio measures		
Rsa-role ratio	$\frac{\lvert rsa\rvert}{\lvert R\rvert}$	⇑
Tra-role ratio	$\frac{\lvert tra\rvert}{\lvert R\rvert}$	⇑
Use of hierarchies	$\frac{\lvert rh\rvert}{\lvert R\rvert}$	⇑
Tree ratio	$\frac{\lvert Arcs\rvert}{\lvert Nodes\rvert}$	⇑
Constrained tasks ratio	$\frac{\lvert T_c\rvert}{\lvert T_T\rvert}$	⇑
Structural measures		
Structural measures for roles		
Max. distance	max. shortest path from one role to another	⇑
Number of separate components (roles)	number of independent role-graphs	⇓
Structural measures for constraints		
Number of constrained tasks	$\lvert T_c\rvert = \lvert T_{sb} \cup T_{rb} \cup T_{sme} \cup T_{dme}\rvert$	⇑
Number of unconstrained tasks	$\lvert T_{uc}\rvert = \lvert T_T \setminus T_c\rvert$.	⇓
Average number of related tasks (rt)	average number of tasks related to one task either through role-hierarchy or constraints	⇑
Constraint impact	average number of tasks related to one task only through transitive constraints	⇑
Number of separate components (constraints)	number of task-groups connected with constraints	⇓
Semantic measures		
Cumulative reasoning effort	see Section 3.2	⇑

⇑ Positive correlation with complexity
⇓ Negative correlation with complexity

average number of tasks that are related to at least one other task. A relation between two tasks exists if both tasks are connected either directly or transitively via a constraint or the inheritance relations in a role-hierarchy (see Fig. 2). A large number of related tasks may indicate an increased reasoning effort for adding new RBAC elements because the consistency checks for an RBAC model are more complex for an increasing number of model elements (see, e.g., [15,16]). In addition, a set of constraints may create a chain of related tasks. We therefore introduce the *constraint impact* measure which represents the average number of transitively related tasks via constraints (see Table 1). Furthermore, similar to the measures for the role-hierarchy, we calculate the *number of separate components* for graphs that connect tasks via entailment constraints.

3.2 Semantic Complexity

Even though different relations are syntactically similar, their semantics are diverse and so is their influence on the complexity and comprehensibility of an RBAC model. We therefore extend our set of measures to consider such semantics of RBAC model elements. In particular, our semantic measures are derived from the generic algorithms presented in [15], and corresponding implementation experiences of several open source projects (see, e.g., [4,20,21,22]).

In the following, we discuss these algorithms with respect to their computational complexity.

Table 2 depicts the effort needed if an entailment constraint or an assignment relation is added to an existing RBAC model. Each connection has two properties that we consider for its semantic: *a)* required number of checks and *b)* its transitive dependency. Property *a)* counts the number of checks required to add a specific constraint or relation to an RBAC model, whereas *b)* represents the transitive dependency calculation within each algorithm (see [15]). The assumption we make here is that transitive checks influence the calculation depending on the size of the model. In this way, the definition of a new task-to-role assignment tra only requires three checks but contains four transitive dependencies. In contrast, the definition of a new dme constraint also requires three checks but only one check for transitive dependency (see [15]). The size of the RBAC model therefore influences the calculation effort of adding new constraints or assignment relations.

The transitive dependencies are checked via nested loops (see [15]) and introduce an exponential complexity. Therefore, based on Table 2, we extract the following formula which represents the cumulative reasoning effort RE for a given process-related RBAC model PRM:

$$RE(PRM) = 6sme2^5 + 3dme2 + 4rb2^2 + 7sb2^2 + 4rh2^6 + 3tra2^4 + rsa2^3.$$

The different components of the formula were defined via the same pattern. For example, the first component $6sme2^5$ multiplies the number of sme constraints in an RBAC model by 6. Here, the multiplier "6" results from the number of checks that need to be performed when adding a new static mutual exclusion constraint

Table 2. Weight of RBAC algorithms from [15]

constraint/relation	number of checks	transitive dependency
static mutual exclusion	6	5
dynamic mutual exclusion	3	1
role binding	4	2
subject binding	7	2
role to role	4	6
task to role	3	4
role to subject	1	3

(see Table 2). Next, the result (6*sme*) is multiplied by 2^5. This factor (2^5) introduces a weight to reflect the exponential complexity of the corresponding consistency checks. The exponent "5" results from the number of transitive dependencies (checked via nested loops) that need to be checked when defining a new *sme* constraint. Finally, the base "2" (in 2^5) was chosen because it produces a realistic (semantic) weight for the different elements (in our experiments, other values resulted in a significant distortion and did not yield meaningful weights).

4 Case Study

In this section, we present the results of a case study where we applied the complexity measures on a set of RBAC models. In particular, we generated 10 models each including 8 roles, 15 task types, 15 subjects, and a similar number of task-to-role and role-to-subject assignment relations. Thus, the models mostly differ with respect to the number role-to-role assignment relations and entailment constraints.

Table 3 shows the results of the case study. Although the number of roles, task types, and subjects was constant, we see considerable differences in the corresponding complexity measures. For example, model *m11* has the lowest number of arcs but the highest tra-role ratio (average number of *tra* relations for each role). In case of a high tra-role ratio, we must consider an increasing number of consistency checks for the definition of new entailment constraints or assignments relations (see also [15]).

Although model *m22* includes *dme* constraints, the corresponding estimated reasoning effort resulting from the models semantic complexity is only slightly higher than the effort for model *m12* does (see semantic measure 3380 vs. 3456). This is due to the fact that the *dme* constraints only require a very low amount of checks and include a small number of transitive dependencies (see Table 2). Moreover, note that, role-hierarchies may reduce the amount of *tra* relations (see also [23]). When we compare *m21* and *m22* (which contain a similar amount of *tra* relations) the result of the semantic complexity measure supports our assumption of the higher reasoning effort for model *m22* because of its additional role-hierarchy that requires more cross-checks to make sure that the consistency

Table 3. Measurement results

	m11	m12	m21	m22	m31	m32	m41	m42	m51	m52
Constant: $\|R\| = 8$, $\|T_T\| = 15$, $\|S\| = 15$, $\|Nodes\| = 38$										
Counting measures										
$\|Arcs\|$	86	87	89	95	101	102	97	103	92	99
$\|rsa\|$	44	44	42	45	57	55	48	49	50	44
$\|tra\|$	42	38	37	35	34	32	39	39	32	40
$\|rh\|$	0	5	0	5	0	5	0	5	0	5
$\|sme\|$	0	0	0	0	10	10	0	0	0	0
$\|dme\|$	0	0	10	10	0	0	0	0	0	0
$\|rb\|$	0	0	0	0	0	0	10	10	0	0
$\|sb\|$	0	0	0	0	0	0	0	0	10	10
Ratios										
rsa-role-ratio	5.50	5.50	5.25	5.63	7.13	6.88	6.0	6.13	6.25	5.5
tra-role-ratio	5.25	4.75	4.63	4.38	4.25	4.00	4.88	4.88	4.0	5.0
use of hierarchies	0.0	0.63	0.0	0.63	0.0	0.63	0.0	0.63	0.0	0.63
tree-ratio	2.26	2.08	2.34	2.50	2.66	2.68	2.55	2.71	2.42	2.61
constrained tasks ratio	0.0	0.0	0.8	0.87	0.73	0.67	0.73	0.67	0.73	0.67
Structural measures on roles										
max. role-distance	0	2	0	3	0	4	0	3	0	2
Number of separate components	8	3	8	3	8	3	8	4	8	3
Structural measures on constraints										
Number of constrained tasks	0	0	12	13	11	10	11	10	11	10
Number of unconstrained tasks	15	15	3	2	4	5	4	5	4	5
Average number of related tasks	9.6	10.4	9.53	10.93	11.73	10.47	8.93	11.33	7.4	9.93
Average constraint impact:	0.0	0.0	0.4	0.4	0.6	0.53	0.0	0.0	0.0	0.0
Number of separate components (constraints)	15	15	6	5	5	6	7	6	6	7
Semantic measures										
RE	2368	3456	2172	3380	4008	5176	2416	3704	2216	3832

requirements of the RBAC model are not violated. The results of the case study suggest that a combination of several measures is required to assess the complexity of an RBAC model. As a preliminary result, we found that the semantic complexity measure is a useful indicator for the reasoning effort that is required to understand an RBAC model. In our future work, we will conduct further case studies to assess the correlation between the different measures.

5 Related Work

In the RBAC context, a variety of different views on complexity exist, including computational complexity [24], size and structure [17,25], performance measures for evaluating access control requests [26], or with respect to maintenance effort [6]. In [24], Colantonio et al. examine the complexity of role mining algorithms, i.e. the effort for the computation of an RBAC model. Furthermore, El Kateb et al. argue that the increasing number of RBAC model elements can cause performance issues and propose an approach to split RBAC models [26]. In contrast, Jaeger [6] defines the complexity of RBAC models as the cumulative effort to create and maintain an RBAC model. Similar to Jaeger, we define the complexity of an RBAC model as the effort required to understand this model.

Furthermore, our work is related to previous contributions from the area of role mining. Role mining aims to derive RBAC models from permissions that exist in the software systems of an organization [27]. In this way, role mining may also help to optimize and refactor an existing RBAC model to simplify access control administration [17,25,28,29]. For example, Li et al. [17] argue that the complexity increases with an increasing number of elements and relations in an RBAC model, while Vaidya et al. [25] argue that the complexity is influenced by the number of roles that are needed to represent the permissions of a user. In summary, the measures used in role mining for discovering an RBAC model mainly focus on the size and structure of an RBAC model. Our work complements this stream of research with a set of measures that can be used to improve RBAC model refactoring with respect to reasoning effort.

6 Conclusions

In this paper, we presented a set of measures for assessing the complexity of RBAC models. Our measures aim to capture the size, the structure, and the semantics of different RBAC model elements. We evaluated our measures in a case study that included 10 generated RBAC models with a constant number of roles, tasks, and subjects, and a variable number of entailment constraints and role-to-role assignment relations. Our approach complements existing approaches, and the results of our case study indicate that our measures can describe the characteristics of an RBAC model in a detailed way.

In our future work, we will elaborate the suitability of the complexity measures for a larger set of RBAC models with a higher amount of elements and relations, as well as a broader variety of combinations of elements and relations. This will involve synthetically generated models as well as real-world RBAC models. Furthermore, we aim to validate the measures in experiments and apply them in (automatic) refactoring of RBAC models.

References

1. Sandhu, R., Coyne, E., Feinstein, H., Youman, C.: Role-Based Access Control Models. IEEE Computer **29**(2) (February 1996)
2. Strembeck, M.: Scenario-Driven Role Engineering. IEEE Security & Privacy **8**(1) (January/February 2010)
3. Tan, K., Crampton, J., Gunter, C.A.: The Consistency of Task-Based Authorization Constraints in Workflow Systems. In: Proc. of the 17th IEEE Workshop on Computer Security Foundations (CSFW), IEEE Computer Society (June 2004)
4. Strembeck, M., Mendling, J.: Modeling process-related RBAC models with extended UML activity models. Information and Software Technology **53**(5) (2011)
5. Wainer, J., Barthelmess, P., Kumar, A.: W-RBAC - A Workflow Security Model Incorporating Controlled Overriding of Constraints. International Journal of Cooperative Information Systems (IJCIS) **12**(4) (December 2003)
6. Jaeger, T.: Managing access control complexity using metrics. In: Proc. of the 6th ACM Symposium on Access Control Models and Technologies (SACMAT). (2001)
7. Dranger, S., Sloan, R., Solworth, J.: The Complexity of Discretionary Access Control. In: Advances in Information and Computer Security. Volume 4266 of LNCS. Springer (2006)
8. Hu, V.C., Kuhn, D.R., Ferraiolo, D.F.: The Computational Complexity of Enforceability Validation for Generic Access Control Rules. In: IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), IEEE Computer Society (2006)
9. Chidamber, S., Kemerer, C.: A metrics suite for object oriented design. IEEE Transaction on Software Engineering **20**(6) (1994) 476–493
10. Fenton, N., Kitchenham, B.: Validating software measures. Softw. Test., Verif. Reliab. **1**(2) (1991) 27–42
11. Henry, S., Kafura, D.: Software Structure Metrics Based On Information-Flow. IEEE Transactions On Software Engineering **7**(5) (1981)
12. Mendling, J., Strembeck, M., Recker, J.: Factors of process model comprehension - Findings from a series of experiments. Decision Support Systems **53**(1) (2012)
13. Reijers, H.A., Mendling, J.: A Study Into the Factors That Influence the Understandability of Business Process Models. IEEE Transactions on Systems, Man, and Cybernetics, Part A **41**(3) (2011)
14. Schaad, A., Moffett, J., Jacob, J.: The role-based access control system of a European bank: a case study and discussion. In: Proc. of the 6th ACM symposium on Access control models and technologies (SACMAT). (2001)
15. Strembeck, M., Mendling, J.: Generic Algorithms for Consistency Checking of Mutual-Exclusion and Binding Constraints in a Business Process Context. In: Proc. of the 18th International Conference on Cooperative Information Systems (CoopIS), LNCS, Vol. 6426, Springer. (2010)
16. Schefer, S., Strembeck, M., Mendling, J., Baumgrass, A.: Detecting and Resolving Conflicts of Mutual-Exclusion and Binding Constraints in a Business Process Context. In: Proc. of the 19th International Conference on Cooperative Information Systems (CoopIS), LNCS, Vol. 7044, Springer. (2011)

17. Li, N., Li, T., Molloy, I., Wang, Q., Bertino, E., Calo, S., Lobo, J.: Role Mining for Engineering and Optimizing Role Based Access Control Systems. Technical Report CERIAS Tech Report 2007-60, Center for Education and Research in Information Assurance and Security (2007)
18. Ramalingam, G., Reps, T.: On the computational complexity of dynamic graph problems. Theoretical Computer Science **158**(1-2) (1996) 233–277
19. Jukna, S.: On graph complexity. Combinatorics, Probability and Computing **15**(06) (2006)
20. Mark Strembeck: Business Activity Library and Runtime Engine. Available at: `http://wi.wu.ac.at/home/mark/BusinessActivities/library.html` (2013)
21. Strembeck, M.: xoRBAC: a policy decision point and policy repository for RBAC policies. Available at: `http://wi.wu.ac.at/home/mark/xoRBAC/index.html` (2013)
22. Strembeck, M.: xoRET: a software tool for RBAC policy engineering. Available at: `http://wi.wu.ac.at/home/mark/xoRET/index.html` (2013)
23. Guo, Q., Vaidya, J., Atluri, V.: The Role Hierarchy Mining Problem: Discovery of Optimal Role Hierarchies. In: Proc. of the 2008 Annual Computer Security Applications Conference (ACSAC), IEEE Computer Society (2008)
24. Colantonio, A., Pietro, R.D., Ocello, A., Verde, N.V.: Taming role mining complexity in RBAC. Computers & Security **29**(5) (2010)
25. Vaidya, J., Atluri, V., Guo, Q.: The role mining problem: A formal perspective. ACM Trans. Inf. Syst. Secur. **13** (July 2010)
26. El Kateb, D., Mouelhi, T., Le Traon, Y., Hwang, J., Xie, T.: Refactoring access control policies for performance improvement. In: Proc. of the 3rd ACM/SPEC International Conference on Performance Engineering (ICPE). (2012)
27. Kuhlmann, M., Shohat, D., Schimpf, G.: Role Mining - Revealing Business Roles for Security Administration using Data Mining Technology. In: Proc. of the 7th ACM Symposium on Access Control Models and Technologies (SACMAT). (2003)
28. Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S., Lobo, J.: Mining Roles with Semantic Meanings. In: Proc. of the 14th ACM Symposium on Access Control Models and Technologies (SACMAT). (2008)
29. Molloy, I., Li, N., Li, T., Mao, Z., Wang, Q., Lobo, J.: Evaluating role mining algorithms. In: Proc. of the 14th ACM Symposium on Access Control Models and Technologies (SACMAT). (2009)