



HAL
open science

Assignment of Actors to Activities at Process-Oriented Applications: A Research Agenda

Thomas Bauer, Ralf Laue

► **To cite this version:**

Thomas Bauer, Ralf Laue. Assignment of Actors to Activities at Process-Oriented Applications: A Research Agenda. 14th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM), Nov 2021, Riga, Latvia. pp.281-291, 10.1007/978-3-030-91279-6_20 . hal-04323860

HAL Id: hal-04323860

<https://inria.hal.science/hal-04323860>

Submitted on 5 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Assignment of Actors to Activities at Process-oriented Applications: A Research Agenda

Thomas Bauer¹ and Ralf Laue²

¹ Hochschule Neu-Ulm, Wileystr. 1, 89231 Neu-Ulm, Germany

² Westsächsische Hochschule Zwickau, Kornmarkt 1, 08056 Zwickau, Germany
thomas.bauer@hnu.de, ralf.laue@fh-zwickau.de

Abstract. Current process management systems (PMS) use graphical modelling languages such as BPMN for modelling the control flow. Almost no program code has to be written by business process designers for defining the possible order of activities. For another important aspect of business processes – assigning actors to activities – no such simple and standardized way of modelling exists. This causes problems at business process implementation projects in practice. We explain such aspects and shortcomings of current commercial PMS that lead to these problems. For each aspect, it is analysed whether appropriate scientific solutions exist, or whether it is an unsolved research question.

Keywords: Organisational Perspective, Actor Assignment, Organizational Model, Metamodel, Escalations, Substitutions, Research Questions.

1 Motivation

Modelling of business processes (BP) consists of the specification of several perspectives. This concerns business-oriented BP modelling (i.e. the business view, cf. ARIS) as well as BP implementation (the technical view) based on a process management system (PMS). The control-flow can be modelled graphically as a process graph and there exist standardized modelling languages for this perspective (e.g. BPMN). In the following we focus on the organizational perspective that describes which users are assigned to which activities and for which purpose. For each activity (that shall be performed by a person), an *actor assignment* defines the potential actors who are allowed to execute this activity. In addition, the behaviour in exceptional situations may be defined. For instance, an *escalation* shall be triggered in case of a delayed execution of the activity, or the long-term absence of its regular actors may be handled by a *substitution* mechanism. The organizational perspective is relevant at business-oriented BP design not only for the purpose of a complete BP documentation, but also for the training of the process participants and for process simulation; e.g. to calculate the expected workload of users. At BP implementation, an even more detailed modelling of this perspective is necessary since the PMS must be able to calculate exactly the users who shall perform a specific activity, escalation, or substitution. For this purpose, the BP engine sends a query to the organizational database that

stores the organizational model. This procedure requires that, at BP design time, rules are modelled that exactly define how to calculate the desired users.

In order to fulfil all requirements that occur in practice, it must be possible to model sophisticated rules for this purpose. Currently, however, there does not exist a standard for organizational modelling (cf. BPMN) and, as we will show, several commercial PMS do not support this appropriately. That means, it is not possible to define complex rules in an easy to use manner; e.g. graphically as possible with BPMN for the control-flow perspective. We present aspects where scientific concepts that would enable this are still missing. Thereby, the focus is on aspects that cause the problems we have identified in projects from practice and at current PMS. The reason for this decision was that defining the organizational perspective causes much effort in existing PMS. Sometimes it is not possible to define a required assignment rule at all, or the assignment must be realized by writing program code (despite BP designers normally do not possess programming skills). As we observed in real life implementation projects, in the worst case, this may result in the decision not to realize a process-aware application (PAIS) at all (but only a data and function-oriented application).

2 Basic Requirements for the Organizational Perspective

We explain relevant requirements and resulting problems that the first author has observed in real-life projects during ten years of practical experience with BP management at a large German vehicle manufacture. Then, we describe whether and how these requirements are fulfilled at some commercial PMS.

Project Experiences - Actor assignments: The purpose of (regular) actor assignments is to define the users who shall perform a specific activity. In practice, many types of actor assignments are required that do not only use roles but also groups, competences, departments, etc. It must be possible to define expressions combined with Boolean operators, e.g. “role = *Software Developer* \wedge group = *Development S-Class*”. Furthermore, dependent actor assignments may refer to preceding activities or process instance data, e.g. “not same actor as at Activity X”. Ideally, a PMS should offer all required actor assignments in an easy to use manner; e.g. using graphic modelling or by combining arbitrary textual templates provided by the modelling tool.

A problem at the observed projects was that the PMS did not support all required types of actor assignments. Especially, combined expressions and dependent actor assignments were not possible or could be only realized by writing program code (e.g. JavaScript). Furthermore, there does not exist a standard for the modelling of actor assignments. Therefore, even the transformation of the business view (for instance created with ARIS) to the technical implementation causes high effort, since the actor assignments were defined with a different modelling method, using different object types, and with insufficient details at the business view.

Metamodel of the Organizational Database: The potential actors that are allowed to perform a specific activity are calculated by a query to the organizational database. Such a database needs a comprehensive metamodel to support all required types of organizational objects (i.e. roles, groups, skills, teams, etc.). Ideally, the same organi-

zational database should be used by all process-aware applications (even if they are based on different PMS) as well as by all traditional applications (the user directory).

Some metamodels of the organizational databases did not possess the required comprehensiveness (missing object types). Furthermore, no standard exists for such a metamodel. Hence, before starting with the main part of the BP implementation project, it was necessary to develop an appropriate metamodel in an external database or to extend the database of the PMS (e.g. with work-arounds). This causes delays as well as effort for the project, and represents an initial hurdle for the realization of a process-aware application. Because of the missing standardization and the absence of products providing a generally usable process-oriented organizational database, it was not possible to use the same data with different PMS. Having multiple organizational databases, however, results in high effort for their creation and maintenance.

Escalations: If an activity is not started or completed in time, an escalation is used, for instance, to inform other users or to change the set of its potential actors. For such an escalation, several parameters have to be defined; e.g. the time when it shall be triggered, the action that has to be performed, and the set of target persons. For the last aspect, complex rules may be required to define escalations as, for instance, “assign the activity to a colleague with the same role after two hours and to the supervisor after one day”. At this informal description of a rule, “colleague” stands for a different person with the same role and from the same department; i.e. a complex formal rule has to be defined in order to specify such an escalation. Since different target persons may be responsible for escalations concerning different activities, multiple such rules have to be defined; i.e. they are activity-dependent.

At the observed projects, escalations were respected very seldom. A reason may be that escalations normally were not part of business-oriented BP models; i.e. the business view created during the requirement analysis phase. Therefore, it would be necessary to capture escalations by a separate analysis; e.g. later at the BP implementation phase. If a project avoids this effort, the resulting application will not contain any escalations. Furthermore, the limited escalation functionality of the respective PMS was an additional argument not to realize escalations at all. Since escalations should be implemented by a core component of the PMS, it would be much effort to realize sophisticated escalation mechanisms at the application level (i.e. as work-around).

Substitutions: Another exceptional situation is that actors are absent for a long time. Then, substitutions may be used to transfer the activity to other users. Additionally to the information required for escalations, the BP designer has to define, for instance, whether the substitution shall be activated when one, a given quota, or all potential actors are currently absent. Furthermore, since different departments may use different substitution strategies, different target person rules may be necessary for different original actors. Since this results in high effort, an efficient modelling technique is required that allows the flexible definition of the target persons of substitutions.

Similar to escalations, at most observed projects, the business view did not contain substitution rules. But a difference was that during BP implementation it was well-known that some kind of substitution is necessary to avoid that an activity is inserted solely into worklists of users who will be absent for a long time. Furthermore, it was obvious that the substitution mechanisms offered by the PMS are insufficient to real-

ize all requirements of the project. To solve these problems, organizational work-arounds were analysed as the definition of several fixed substitutes for an actor for all activities or the definition of a manager as a potential actor who has solely the job to delegate the activity manually if the regular actors are absent. That means, additional effort became necessary to identify and realize an acceptable work-around.

Commercial PMS¹ - Actor Assignments: With K2, it is possible to define an actor assignment that consists of several parts. A condition (if) is assigned to each part of the rule to specify when it shall be used. However, no dependent actor assignments are supported. Programming of an actor assignment is not possible. Instead, the only way to realize an actor assignment is to fill out the form “Recipients Rule”. With Signavio, actor assignments can be defined by filling a form as well. Additionally, two types of dependent actor assignments are offered: It is possible to define that an activity has to be assigned to the same actor as a preceding activity. Separation of duties can be modelled as well. The only possibility to define further actor assignments is to write JavaScript code. Bizagi enables the creation of complex rules (with Boolean operators) in a graphical editor and arbitrary actor assignment rules can be defined with XPath expressions. IBM Business Process Designer allows assigning an activity to a participant group, the actor of a preceding activity, or the starter of the process instance. Additionally, Routing Policies enable defining complex rules with conditions that use process variables and Boolean operators. But Routing Policies are no longer available in the succession product IBM Business Automation Workflow. Here, JavaScript code must be written to realize complex actor assignments: Team retrieval services can be realized to calculate the IDs of the appropriate users; e.g. by using functions of the product API or by calling an external service. Team filter services get the UserIDs of a participant group as input. Then, program code can be written that eliminates not intended users (e.g. wrong department, separation of duties).

Metamodel of the Organizational Database: The IBM metamodel comes with the fewest possibilities to structure organizational data since only so-called participant groups are offered. Signavio additionally enables the usage of roles. This is extended by K2 and Bizagi with competences and organizational units inclusive the possibility to define a hierarchy (e.g. team, department, center).

Escalations: Bizagi does not support any escalations. Signavio is able to send an email to a user in case of a missed deadline. In addition, K2 can transfer the activity to another user, but only a single user can be specified as target person (i.e. no group). IBM offers notifications via email and transferring an activity to other actors as well. Such escalations can be addressed to multiple persons.

Substitutions: Bizagi and Signavio do not offer substitution mechanisms. K2 enables defining different substitutes for different activities. IBM allows to select for each activity whether it shall be transferred to a substitute or not. Only one list of substitutes can be defined for each original actor, i.e. the substitutes cannot depend on the concerned activity or process data.

¹ We inspected the PMS Bizagi Studio V.11.2.3, IBM Business Process Designer V.8.0.1 and V.8.6.0 as well as the succession product IBM Business Automation Workflow V.20.0.0.2, K2 Cloud V.4.0, and Signavio Workflow Accelerator V.13.6.0 (for details see [5]).

3 Scientific State of the Art and Resulting Research Questions

In the following, scientific work is discussed that addresses these problems of the projects and the shortcomings of PMS. Thereby, the goal is to identify approaches, which have the potential to solve these topics, and not to present a complete overview on the literature concerning the organizational perspective of BP. Therefore, we will discuss literature that handles aspects causing these problems for practice. On the one hand, we want to identify which problems can be solved with existing scientific approaches. In such cases, the conclusion is that these approaches are solely not realized by current PMS technology. On the other hand, for the remaining problems, the conclusion must be that currently no appropriate scientific solution exists, and therefore, this is an unsolved research question.

Actor Assignments: A fundamental work on resources in BP are the workflow resource patterns [20]. This work describes various ways how activities are assigned to actors. It includes basic patterns such as role-based distribution of activities as well as more complex constraints such as separation of duties (4-eyes-principle) and assigning an activity to the user who has already performed a specific preceding activity.

[6] develops a language for the definition of additional constraints for actor assignments. They can be used, for instance, to realize separation of duties. [14] mentions some requirements for dependent actor assignments, e.g. that the potential actors may depend on process instance data (i.e. application data).

BPMN offers XPath as a method to define actor assignments. Thereby, the XPath expressions have to be defined by the BP designer. This is a similar difficulty as writing such a rule using a programming language. In addition, it is not possible to use the actor specification as a means to discuss the process among the stakeholders.

Some approaches enable additional types of actor assignments by extending standardized BP modelling languages as BPMN and UML: [12] extends the data model of BPMN in order to cover all types of actor assignments mentioned in the workflow resource patterns [20]. [23] develops a metamodel that allows to store the organizational perspective. Again, these classes extend the metamodel of BPMN. The paper, however, does not suggest concrete new object types as, for instance, roles, departments, or capabilities. [24] extends BPMN to enable the definition of additional constraints for actor assignments. The approach allows the definition of simple constraints for actor assignments (e.g. same actor as a preceding activity, separation of duties), as well as very complex rules (e.g. the same user is only allowed to perform max. 5 activities of the process instance). The constraints are defined by threshold values (i.e. numbers). For BP designers, this method is at least unusual. [22] extends UML activity diagrams with an organizational perspective. The authors suggest so-called “Business Activities” in order to enable the BP designer to define role-based access control (RBAC). [15] extends UML use case diagrams to realize actor assignments, delegations, and substitutions. This concept allows a transformation of the Platform Independent Model (PIM) into a Platform Specific Model (PSM). The latter is used by the PMS to control BP execution at runtime.

Some work addresses special types of actor assignments: [7] enhances the idea of separation of duties with advanced requirements, for instance, by respecting conflicts

between roles and users. To give an example, a family member of a requestor may not be allowed to perform the corresponding approval activity. [1] extends the organizational model with a team concept. Thus, one activity can be performed by several persons (commonly), instead by a single actor. In addition to the (normal) actors, [9] assigns additional persons to an activity for control, information, and support of its execution (cf. RACI matrix). [10] extends the language RAL in order to define dependencies between actors of activities that belong to different process instances. In [21], an actor assignment may contain soft constraints. A soft constraint shall be respected if possible (i.e. it can be ignored); e.g. if it would result in a missed deadline.

Some approaches develop methods that shall enable the simple definition of actor assignments: [3] suggests to extend the metamodel of BPMN with actor assignments. They are defined as constraints expressed in Object Constraint Language (OCL). However, this means that such constraints would have to be written in a formal and less intuitive language by the BP designers. [8] develops the language RAL to assign resources to activities. It uses organizational objects with specific types (Role, Capability, etc.). The advantage of this language is that it is powerful and easy to read (i.e. simple), but again, formal expressions must be written. [11] enhances this approach with the graphical notation RALph. It covers all resource patterns mentioned in [20]. Furthermore, it allows combining parts of rules with Boolean operators in order to define even complex actor assignments. Therefore, important types of actor assignments can be realized. While we think that RALph should be easy to understand, experiments and studies evaluating its understandability are still missing.

Conclusion: Several authors develop techniques with the goal to simplify the modelling of actor assignments. However, we found no studies that evaluate which modelling technique is best suited; e.g. graphic modelling, textual specification (cf. RAL, OCL), or the combination of pre-defined textual templates (cf. the Routing Policies of IBM). Thereby, understandability for the BP designers is a crucial aspect. It may be necessary to distinguish modelling at the business-oriented level and at the technical level (workflow implementation) since the respective BP designers typically have very different IT skills. Furthermore, the resulting modelling technique has to cover dependent actor assignments and the combination of partial expression with Boolean operators. In order to ease the usage of business-oriented modelling tools and PMS from different manufacturers, a standard should be developed for this functionality.

Several papers suggest a set of actor assignment types, some by extending BPMN or UML, and some even respect special requirements as team tasks. But there exist no studies that analyze which types of actor assignments are required in practice, typically. Such studies, however, are required as foundation to develop the set of actor assignments that is offered to the BP designer with the selected modelling technique (see above). Furthermore, no studies analyze whether the same types of actor assignments are required at different business domains; e.g. BP of hospitals, vehicle development, and marketing. Only if this applies, it makes sense for a PMS to offer a pre-implemented mechanism for the definition of actor assignments. Otherwise, using a programming language (e.g. JavaScript) cannot be avoided in most cases.

The missing studies concerning the appropriate modelling technique and the required set of actor assignments may be a reason that currently there does not exist a

standard for the definition of actor assignments. Furthermore, we assume that the tool manufacturers boggle the implementation effort because of the risk that the resulting mechanisms are inadequate or not sufficient for many business domains. At the development of such a modelling method and standardization, the different requirements of the business and the technical level have to be respected. That means, the method for business modelling shall be simple to understand, but not too vague, in order to enable the direct transformation of the actor assignments into a BP implementation. For instance, even at the business level, it should be possible to define dependent actor assignments; i.e. the target must be to develop a comprehensive approach that may be used in all phases of a development project.

Metamodel of the Organizational Database: The W3C ORG Ontology deals with organizational units, membership, roles, posts, and reporting but does not include information that is fundamental for resource assignment such as competencies.

[20] presents several patterns for the organizational perspective incl. a metamodel for the organizational database: Users (human resources) have a position in an organizational unit and may belong to temporarily assigned groups. They may have assigned roles, capabilities, an organizational level, and subordinated persons. This means, the metamodel contains a variety of object types that can be used to define many types of actor assignments. [17] presents a metamodel that additionally contains competences, skills, and knowledge. In addition to users, other types of resources are respected.

The focus of [3] is the integration of actor assignments into the BPMN standard. The paper presents an organizational metamodel as well, but this contains organizational and functional roles inclusive a hierarchy only. Other object types (e.g. capabilities) are not considered. [23] extends BPMN with an organizational perspective too. For this purpose, it presents a graphical visualization for some types of organizational objects. The metamodel allows assigning arbitrary parameters to resources that can be queried at runtime to perform resource assignments. However, such an abstract approach leaves the responsibility to deal with the parameters and the rules how to use them for calculating actor assignments completely to the BP designer.

Approaches for enterprise modelling (e.g. ArchiMate, DEMO, MEMO, VDML) have the goal to enable the planning and improvement of the organization and its IT. For example, ArchiMate uses the concepts business actor and business role for defining the persons who are responsible for executing a task. However, assigning actors at runtime is out of scope; i.e. the approaches do not solve the need for an organizational database for the calculation of actor assignments used by a PMS at runtime.

Conclusion: The best situation for a BP implementation project would be that there already exists a complete and generally usable metamodel for the organizational model, since its organizational types are required to define actor assignments at the business and the technical level. Unfortunately, no such metamodel is proposed in literature. Even worse: there exist no studies that examine whether the same types of organizational objects are required at different business domains. In addition, no comprehensive metamodel is developed in literature. We assume that the types of organizational objects, proposed by the presented papers, are not sufficient (at least for some scenarios) since aspects as, for instance, worker protection cannot be modelled; i.e. cannot be respected in actor assignments.

The development of a comprehensive metamodel, however, would be a prerequisite for the standardization. Without such a standard, PMS use proprietary metamodels. This prevents the realization of a central organizational database for all process-aware applications (based on different PMS). Furthermore, we believe that a standard or concept for such a database should be based on or at least be integrable with regular user directories (e.g. LDAP, Microsoft Active Directory).

Escalations: They are mentioned in PMS literature (e.g. [12, 20]), but there does not exist any work that considers specifically the issue of how to define the target persons efficiently and flexibly. Some literature handles special aspects of escalations:

[2] suggests to react to delays not only if it is already too late. Instead, it considers expected durations for completion of activities and process instances in order to be able to react before a deadline is actually missed. Additionally, an escalation may concern several activities and process instances, e.g. if all activities of a user are delayed because of the same reason. The escalation mechanism can be selected automatically or manually and multiple escalation levels may be activated in sequence, with each level resulting in a different escalation mechanism. Several escalation mechanisms are presented. Hence, the paper describes a very powerful escalation approach that fits very well to the presented scenario of too high workload at a call center.

[18] improves the management of escalations with the goal to minimize their number. An algorithm adjusts deadlines of activities in order to compensate already occurred delays and, therefore, to avoid escalations at succeeding activities. Costs that are caused by not avoidable escalations are minimized by predicting whether an escalation will occur despite these adjustments. In this case, the escalation is triggered as early as possible (in the BP) since costs for early escalations are typically lower.

Conclusion: [2] and [18] present very advanced concepts that are not in our focus since they do not directly solve the mentioned problems of PMS in practice. The other papers mention the necessity of escalations but do not develop solution concepts. This is acceptable for many aspects of escalations that are quite simple as, for instance, the selection of the escalation mechanism (e.g. email vs. worklist entry) or the definition of the escalation deadline (e.g. measured from the availability or reservation time of the activity till its start or completion). The definition of the target persons of an escalation, however, requires a more sophisticated concept: They shall not only depend on the concerned activity and the process context, but also on the original actor of the activity (e.g. at an escalation to the supervisor). Furthermore, even different rules for different original actors may be required (e.g. escalation to the team leader for actors who belong to department X and to the department leader otherwise). It would result in too much effort for the BP designer, however, to define separate rules for each relevant combination of concerned activity and original actor. Therefore, a concept has to be developed that significantly differs from normal actor assignments.

Substitutions: [4] presents several requirements: It shall be possible to model rules that depend on the absent person and the concerned activity. It must be definable, whether the substitution shall be activated when one or all potential actors are currently absent, or when they have actively demanded to be substituted. Furthermore, multi-level substitutions may be desired (if substitutes are absent too) and activities may be revoked from a substitute when the original actor returns. Additionally, for all these

cases, algorithms are presented that calculate the resulting substitutes. The work, however, does not contain a concept that enables “efficient” modelling of such substitution rules. That means, similar as at escalations, high effort may result for the BP designers to define many separate substitution rules.

Other papers do not offer solutions for this issue as well: The resource patterns [20], consider substitutions not as a separate topic, but as a delegation that is performed by the PMS. [16] describes a simple substitution mechanism: An (eventually restricted) access to the worklist of the substituted person is granted. The substitutes can solely be defined with roles. [13] only mentions that substitutions are necessary. However, no detailed requirements and no solutions are presented. [19] describes a metamodel for the organizational perspective that contains substitutions as well. The corresponding substitution rules, however, cannot depend on the concerned activity. Despite the necessity of context-related substitutions is mentioned, no concept for the modelling of such substitution rules is presented.

Conclusion: Compared to escalations, there exist more papers that handle substitutions. Additional requirements are presented, as for instance multi-level substitutions or revoking a substitution. As well, corresponding solution concepts were published. But again, there remains the unsolved topic, how to define the target persons efficiently: Different substitution rules are required for different activities and original actors. In addition, multi-level substitutions may require further or modified substitution rules. Therefore, there remains the unsolved research question, how to avoid the definition of many different rules in order to reduce the effort for the BP designer; i.e. how to re-use and adapt such rules (e.g. for different departments with slightly different substitution strategies).

Resulting Research Questions: The organizational perspective is essential to model a BP comprehensively and to enable a PMS to control the BP execution. Unfortunately, there occur several problems at the implementation of BP in practice. The following unsolved topics (i.e. research questions) still exist: (i) Researching which requirements for resource assignment exist in different business domains; i.e. which types of actor assignments and organizational objects are required. (ii) Based on these results: Developing a comprehensive modelling method and establishing a standard for actor assignment; i.e. a uniform organizational metamodel and easy to use actor assignments for all business domains. (iii) Creating formalisms and algorithms for the identified requirements, especially an efficient method for defining the set of target persons for escalations and substitutions that considers the non-trivial requirements as well. (iv) Evaluating and comparing the expressiveness, understandability, and usability of the formalisms resulting from (ii) and (iii).

References

1. Aalst WM van der, Kumar A (2001) A Reference Model for Team-Enabled Workflow Management Systems. *Data & Knowledge Engineering*:335–363
2. Aalst WM van der, Rosemann M, Dumas M (2007) Deadline-based Escalation in Process-Aware Information Systems. *Decision Support Systems*:492–511

3. Awad A, Grosskopf A, Meyer A et al. (2009) Enabling Resource Assignment Constraints in BPMN, Hasso Plattner Institute, Potsdam
4. Bauer T (2009) Substitution Rules for Task Actors in Process-oriented Applications. *Datenbank-Spektrum* 9:40–51 (in German)
5. Bauer T, Laue R (2020) State of Application-related Research and Technology for the Organizational Perspective of Business Processes. *Proc. ZuGPM*:605-619 (in German)
6. Bertino E, Ferrari E, Atluri V (1999) The Specification and Enforcement of Authorization Constraints in WfMS. *ACM Transactions on Information and System Security* 2:65–104
7. Botha RA, Eloff J (2001) Separation of Duties for Access Control Enforcement in Workflow Environments. *IBM Systems Journal*:666–682
8. Cabanillas C, Resinas M, Ruiz-Cortés A (2011) RAL: A High-Level User-Oriented Resource Assignment Language for Business Processes. *Proc. BPM*:50–61
9. Cabanillas C, Resinas M, Ruiz-Cortés A (2012) Automated Resource Assignment in BPMN Models Using RACI Matrices. *Proc. CoopIS*:56–73
10. Cabanillas C, Resinas M, Ruiz-Cortés A (2012) Designing Business Processes with History-Aware Resource Assignments. *Proc. BPM*:101–112
11. Cabanillas C, Knuplesch D, Resinas M et al. (2015) RALph: A Graphical Notation for Resource Assignments in Business Processes. *Proc. CAiSE*:53–68
12. Großkopf A (2008) An Extended Resource Information Layer for BPMN, Hasso-Plattner-Institute for IT Systems Engineering, Potsdam
13. Hochmüller E, Dobrovnik M (2005) Flexibility Issues in Workflow Management Systems. *Proc. Business Process Modeling, Development and Support* 5
14. Künzle V, Reichert M (2009) Integrating Users in Object-aware Process Management Systems: Issues and Challenges. *Proc. BPM*:29–41
15. Link S, Hoyer P, Schuster T et al. (2008) Model-Driven Development of Human Tasks for Workflows. *Proc. 3rd Int. Conf. on Software Engineering Advances*:329–335
16. Mühlen M zur (2004) Organizational Management in Workflow Applications – Issues and Perspectives. *Information Technology and Management Journal* 5:271–291
17. Oberweis A, Schuster T (2010) A Meta-model based Approach to the Description of Resources and Skills. *Proc. Americas Conf. on Information Systems*
18. Panagos E, Rabinovich M (1998) Reducing Escalation-Related Costs in WFMSs. *Workflow Management Systems and Interoperability*:107–128
19. Rosemann M, Mühlen M zur (1998) Modelling of the Organizational Structure in Workflow Management Systems. *EMISA-Forum*:78–86 (in German)
20. Russell N, Aalst W van der, Hofstede A ter et al. (2005) Workflow Resource Patterns: Identification, Representation and Tool Support. *Proc. CAiSE*:216–232
21. Stefansen C, Rajamani S, Seshan P (2008) SOFTALLOC: A Work Allocation Language with Soft Constraints. *Proc. IEEE Int. Conf. on Web Services*:441–448
22. Strembeck M, Mendling J (2011) Modeling Process-related RBAC Models with Extended UML Activity Models. *Information and Software Technology* 53:456–483
23. Stroppi LJR, Chiotti O, Villarreal PD (2015) Defining the Resource Perspective in the Development of PAIS. *Information and Software Technology* 59:86–108
24. Wolter C, Schaad A (2007) Modeling of Task-based Authorization Constraints in BPMN. *Proc. BPM*:64–79