

Automated Resource Assignment in BPMN Models Using RACI Matrices*

Cristina Cabanillas, Manuel Resinas, and Antonio Ruiz-Cortés

Universidad de Sevilla, Spain

{cristinacabanillas,resinas,arui}@us.es

Abstract. Organizations need to manage the responsibility of their employees with respect to all the activities that are daily carried out within them. Process-oriented organizations need to do it, in addition, in accordance to the business processes their members participate in. However, powerful mechanisms to manage responsibility in combination with business processes are missing in current modelling notations, usually limited to indicating who is in charge of undertaking the activities. RACI matrices, on the contrary, were specifically conceived to provide responsibility management information. They enable the specification of the level of responsibility each human resource has with regard to each activity carried out in a company, ranging from the performer of the work to the resource that must approve it or receive certain notifications. In this paper, we propose the use of RACI matrices together with business process models to manage human resource responsibilities in processes. Focused on a concrete type of RACI matrices, called RASCI, we introduce a novel approach to automatically generate a BPMN model with RASCI information given a BPMN model that does not handle resources, and a RASCI matrix. The resulting model is BPMN-compliant and, thus, it is ready to be executed in existing business process management systems. With this approach, the assignment of responsibilities and the management of processes can be designed separately, while being executed together.

Keywords: Responsibility management, RACI matrix, RACI-aware BPMN model, RASCI sub-process, RASCI meta model.

1 Introduction

Organizations need to manage the assignment of responsibilities to their members with respect to the activities that must be carried out within them. This means that, in order to have an action plan of the work performed by every member, not only associating functions to each member of the organization is necessary, but providing a way to organize and display these responsibility assignments is required too. This can be done by means of a Responsibility Assignment Matrix (RAM), also known as RACI matrix or Linear Responsibility

* This work has been partially supported by the European Commission (FEDER), Spanish Government under project SETI (TIN2009-07366); and projects THEOS (TIC-5906) and ISABEL (TIC-2533) funded by the Andalusian Local Government.

Chart (LRC) [1]. Such matrices provide a way to plan, organize and coordinate work, and consist of assigning different degrees of responsibility for each activity developed in the company to the members of an organization, such as who is in charge of undertaking the activity and who must be informed once the action is complete [2]. Several variants extending the functions considered in traditional RACI matrices have appeared (e.g., RASCI matrices).

Besides, process-oriented organizations need to organize and control the activities that are carried out in the company. This is typically done with business process (BP) models that represent the control flow of the activities, together with other perspectives of the process including data and resource management. With regard to resources, most BP modelling notations existing at present allow only the specification of who is in charge of performing the activities of the BP, which is short scope with respect to all the issues involved in human resource or responsibility management (i.e. with respect to RACI's expressiveness). For instance, the de-facto standard for BP modelling, Business Process Modelling Notation (BPMN) [3], has this limitation.

Thus, there is an important distance between the responsibility information that should be managed in an organization, and the one that is actually handled with current BP modelling notations. Furthermore, given the increasing interest of organizations to work with RACI matrices in combination with BPs [4], it is evident that it is necessary either the improvement of the responsibility management capabilities of current BP modelling notations, or the development of a mechanism to enrich the resource-related information contained in BP models.

In this paper, we address this problem and work in the latter direction. Specifically, we introduce a novel approach to generate a BP model with complete responsibility information (i.e. a RACI-aware BP model) from a resource-unaware BP model and a RACI matrix. Our approach tackles two main problems. On the one hand, getting BP models with all the information required to be able to execute them implies generating very accurate resource assignments for the activities of the BP. However, this cannot be done directly due to the high level at which RACI matrices and BP models are built. To overcome this issue, it is necessary to provide extra information for the RACI matrix. In particular, some information about the context in which the process is going to be executed, and some restrictions to be considered, have to be indicated. We call this extra information *binding information*. On the other hand, the control flow of the BP model must be changed according to the functions defined in RACI. To this end, we propose a collection of transformations to model the information of a modality of RACI matrix called RASCI, together with the binding information, into BPMN models. The transformations are as generic as possible and can be automated, and the resulting BP model has no intrusive information about RASCI at first sight. Indeed, RASCI information is modelled at sub-process level¹.

Notice that RASCI information could actually be modelled in BPMN with no need of our approach in an ad-hoc manner. The use of swimlanes constitute a possible mechanism to narrow this gap between RASCI and BPs. We could

¹ Please notice that in this paper we may use terms RACI and RASCI interchangeably.

use them to represent organizational roles and place the proper activities in the proper lanes to comply with the matrix. We worked on that line last year, introducing a collection of *RASCI patterns* and an extension for BPMN 2.0 [3] to allow the modelling of such patterns [5]. However, we realized that proceeding that way has several problems. On the one hand, RASCI functions for each activity are disseminated in the BP model, while they are actually part of the work carried out for a single activity. On the other hand, the resulting BPMN model may unnecessarily become very large and, consequently, difficult to read and understand, due to the increase of lanes. Also, binding information cannot be introduced with this approach based on swimlanes. Finally, keeping the information of both elements (a RASCI matrix and its associated BP model) consistent is difficult, since the modifications performed on one element should be performed on the other as well. The replication of the information thus derives in a synchronization problem. Therefore, the approach we present in this paper has the following advantages with respect to the previous work:

- The resulting BPMN model is complete from the viewpoint of responsibility management with respect to RASCI matrices. In addition, its appearance is very similar of the initial BP model.
- The output BPMN model is ready to be executed in current Business Process Management Systems (BPMSs), provided that they support the allocation of resources to tasks.
- Synchronization of the BP model and the matrix can be performed automatically by carrying out the transformations when the RASCI matrix and/or the binding information change.
- There is a decoupling of BP management and resource management at design time, but they can be automatically mixed together to be executed in combination at run time.

This paper is structured as follows. Section 2 introduces RACI matrices and their use with BP models. Section 3 gives details about the type of binding information required to complement RASCI information. Section 4 describes the meta model that represents the whole application scenario. Section 5 details our transformation-based approach together with some examples. A prototype of the proposal is briefly described in Section 6. Then, some related work is summarised in Section 7, and finally, conclusions drawn from this work and some future work are presented in Section 8.

2 RACI Matrices

RACI matrices constitute a mechanism to represent the assignment of responsibility of the members of an organization. In their standard modality, they are utilised to associate activities with (human) resources, typically by using the organizational roles (e.g. Project Coordinator, Sales Manager) the members of an organization play within the company or given a specific context (e.g. in a specific project or area) [1]. Figure 1 illustrates an example of RACI matrix. The

rows represent *activities* undertaken in a company, the columns of the matrix are (*human*) *resources*, and each cell contains zero or more *RACI initials* indicating the type of responsibility of *such a* resource on *such an* activity. As aforementioned, resources normally come in the form of *organizational roles*, as shown in the table. However, depending on the company, resources may be represented at different levels. For instance: (i) small companies could opt for using persons directly in each column; or (ii) at a very high level we could find RACI matrices in which each column would refer to specific organizational units. In this paper, we are using the standard way, i.e., columns represent organizational roles.

The initial in the cells are different functions, called *roles* in RACI²:

- *Responsible (R)*: person who must perform the work, responsible for the activity until the work is finished and approved by an accountable. There is typically only one person responsible for an activity.
- *Accountable - also Approver or Final Approving Authority - (A)*: person who must approve the work performed by the person responsible for an activity, and who becomes responsible for it after approval. There must be one and only one accountable for each activity.
- *Consulted - sometimes Counsel - (C)*: this role involves the people whose opinion is sought while performing the work, and with whom there is two-way communication.
- *Informed (I)*: person who is kept up-to-date about the progress of an activity and/or the results of the work, and with whom there is just one-way communication. There may be more than one informed person for an activity.

There are several variants of the original version of RACI matrices. Some are based on extending the number of RACI roles to be considered for every activity, e.g, RASCI or RACI-VS. Others give different meanings to the RACI initials. In this paper, we build on RASCI matrices because they use a function that

Table 1. RASCI matrix for the process at pool *ISA Group* of Figure 1

| | Project's PhD Student | PhD Thesis Supervisor | Project Coordinator | Project's Administrative Assistant | Research Group's Clerk |
|--------------------------------------|----------------------------------|----------------------------------|--------------------------------|---|---------------------------------------|
| Submit Paper | R/A | | | | |
| Fill Travel Authorization | R | | A/C | | |
| Sign Travel Authorization | I | | R/A | | |
| Send Travel Authorization | I | | | | R/A |
| Register at Conference | R/A | I | C/I | I | |
| Make Reservations | R/A | C | C | C/I | S |

² We will use the term *RACI role(s)* to differentiate them from organizational roles.

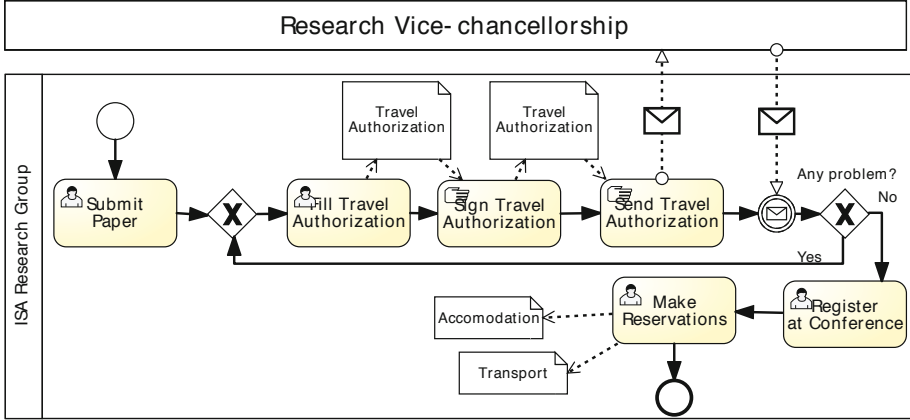


Fig. 1. Conference Travel Management Process

may be interesting specially to IT organizations, where work or tasks needed to complete an activity can usually be delegated to other people. RASCI matrices involve the aforementioned RACI roles together with RASCI role *Support*:

- *Support (S)*: people who may assist in completing an activity, i.e., the person in charge can delegate work to them. Unlike *Consulted*, who may provide input information to the activity (i.e., information helpful to perform some work), *Support* will actively contribute in the completion of the activity.

A process-oriented organization could build one RASCI matrix for each BP used in the company. The matrix would list its activities and the organizational roles that participate in them for each RASCI role. Figure 1 shows a BPMN diagram representing a *collaboration* between two BPs: one BP at pool *Research Vice-chancellorship* and another one at pool *ISA Research Group*³. It illustrates a simplified version of the procedure to manage the trip to a conference, according to the rules of the University of Seville. We are going to focus on the BP carried out at pool *ISA Research Group*. As can be seen in Table 1, the activities of the matrix are exactly the BP activities in the model.

In outline, the process works as follows. It starts with the submission of the Camera Ready version of an accepted paper by the PhD student whose paper has been accepted for publication. After that, *that* student fills in an authorization request to attend and present the paper at the conference. The coordinator of *the project that will finance the trip expenses* must sign the authorization and inform the student when it is done. The clerk of *the research group the PhD student belongs to* is in charge of delivering the form for approval. In absence of problems, *the student* must register at the conference and inform *his/her* PhD thesis's

³ We remind the reader that in BPMN a process takes place within a single pool. Diagrams with two or more pools, in which messages between the pools are exchanged, are called collaborations. All the process-oriented concepts used in this paper are taken from BPMN 2.0 [3].

supervisor, as well as *the project coordinator* and *the administrative assistant of the project*. Finally, *the PhD student* books the tickets needed, assisted by the clerk of *his/her* research group, if required.

However, note that the previous description of the process contains some nuances that are not incorporated in the RASCI matrix. In particular, specific information about the individuals that have to be assigned to the RASCI roles (e.g. the same PhD student during a single execution), or the context within which it has to be done (e.g. the project coordinator of a specific project), is missing in the matrix. The reason is that, as an organization, we aim at modelling BPs that can be applied in different areas of the company (e.g. the same BP may represent how to proceed with the application for a job, regardless of the specific department in which the job is being offered). Similarly, RASCI matrices must be as generic as possible, avoiding the replication of information due to the application of a BP to those different areas or “contexts”. This flexibility in the design of BPs and RASCI matrices is important, but means a problem when trying to automate the combination of both elements, and the generation of the corresponding resource assignments in the resulting BP model. How to solve it? In order to overcome this issue, some extra information must be provided, which we have called *binding information*.

3 Binding Information for Resource Assignment

Binding information complements the resource information provided by RASCI matrices in order to enable automated BP resource assignment. This information can be mainly of two types:

- *Organizational unit context*. Indicating only the organizational role for a RASCI role is usually insufficient, since it does not limit the context in which the BP is going to be run. Let us see it with an example. According to the RASCI matrix in Table 1, role *Project Coordinator* is responsible for activity *Sign Travel Authorization*. However, a project coordinator can sign forms only for the project(s) he/she coordinates, so *not any* project coordinator can perform this task in *any* execution of the process. Therefore, it is necessary to indicate either directly the concrete data required (e.g. name of the project we refer to in the current process instance), or *where* this information can be found, e.g. in our BP the name of the project appears in the *Travel Authorization* form filled in by the student (cf. Figure 1).
- *Additional restrictions*. Other information may be necessary in order to constrain the set of people that can be assigned certain RASCI role. For example, sometimes it is essential that two activities of the same BP be carried out not only by the same organizational role, but by the same person, i.e. Binding of Duties (BoD). In the scenario at hand the same PhD student submits the paper and fills in the travel authorization form. Other times, exactly the opposite may be necessary, i.e. Segregation of Duties (SoD), in order to avoid conflicts of interests between individuals. Restrictions concerning specific skills required to carry out a certain task may also be common.

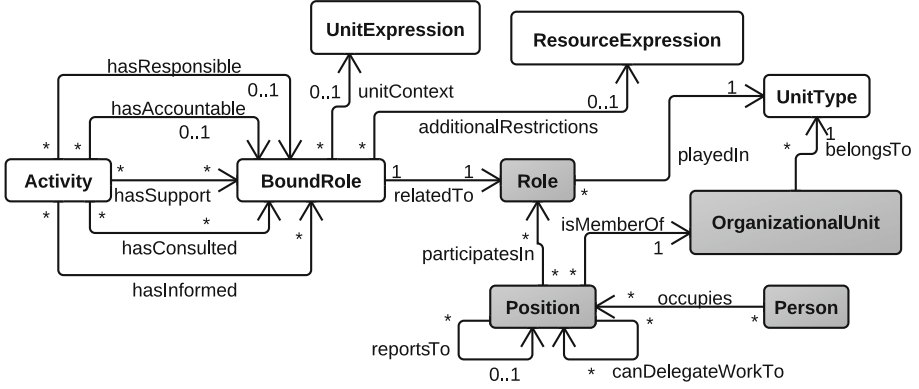


Fig. 2. RASCI meta model with binding information

It is important to let the user define all these additional restrictions that need to be taken into account for resource allocation at run time.

Notice that binding information must be given at RASCI role level, that is, for each RASCI role participating in each BP activity of the matrix.

4 RASCI Meta Model with Binding Information

Taking all the aforementioned aspects into consideration, the meta model of a RASCI matrix with binding information can be defined as shown in Figure 2.

- Class *Activity* represents the activities of the BP the RASCI matrix is associated to, and in which we aim to insert the responsibility-related information necessary to make it work according to the matrix.
- Five relations between *Activity* and *BoundRole* represent the five RASCI roles to be distributed among the members of the organization. The role is bound because it may have binding information associated. The expressions to specify the organizational unit context and any other additional restriction can be defined in classes *UnitExpression* and *ResourceExpression*, respectively.

We have added the following conditions between *Activity* and *BoundRole* in order to define some existence relations between RASCI roles. We use Object Constraint Language (OCL)⁴ to specify the following invariants:

- When there is not a resource responsible for an activity (e.g. an automatic task executed directly by the system), the other RASCI roles cannot exist, except RASCI role I. We exclude the information function (I) because there may be automatic activities in the process consisting of a notification message automatically sent by the system, but whose destination can be a resource indicated in the RASCI matrix.

⁴ <http://www.omg.org/spec/OCL/2.0/>

```

context Activity inv:
if self.hasResponsible->isEmpty()
then
self.hasAccountable->isEmpty() and
self.hasSupport->isEmpty() and
self.hasConsulted->isEmpty()
endif

```

- When RASCI role R is in, then there must be an accountable, since this role is mandatory according to RACI definition⁵ (cf. Section 2).

```

context Activity inv:
if not(self.hasResponsible->isEmpty())
then not(self.hasAccountable->isEmpty())
endif

```

- The classes in gray in the figure represent the part of the organizational meta model described by Russell et al. [6] we have relied on to build the structure of an organization. We have added class *UnitType* for the sake of understanding. In particular, each *BoundRole* is associated to a *Role* of the organizational structure of the company. However, as aforementioned, a person has a role in the context of an organizational unit (e.g. coordinator of a certain project, or research assistant in a specific research group). This relation is modelled by means of class *Position*. A position, thus, represents a collection of roles in one specific organizational unit.

Let us take as example *Activity Sign Travel Authorization* of our use case (cf. BP model in Figure 1 and RASCI matrix in Table 1) to exemplify the RASCI meta model shown in Figure 2. The organizational roles that participate in this activity (i.e. *Project's PhD Student* and *Project Coordinator*) fit in class *Role*. As aforementioned, every role is related to an organizational unit. In this case, it is a *project* (class *UnitType*) called *THEOS* (class *OrganizationalUnit*). There is a positional hierarchy for each organizational unit. For project THEOS it is shown in Figure 3. It has six positions, occupied by seven persons. The relation *participatesIn* of the meta model is outlined in the table attached to the figure. For the rest of organizational units in the company, a similar model is required.

The rest of classes of the RASCI meta model (i.e. *BoundRole*, *UnitExpression* and *ResourceExpression*) are specified at cell level. For RASCI roles R and A, *BoundRole* contains the assignment to role *Project Coordinator*, together with a *UnitExpression* stating that the name of the project can be found in file *Travel Authorization* (handled in the process) during execution. For RASCI role I, *BoundRole* is role *Project's PhD Student* plus a *ResourceExpression* indicating that it has to be the same person who performed activity *Submit Paper*. The language used to specify the binding information, and thus, the whole resource

⁵ The lack of A in the table is interpreted as R and A being assigned to the same role.

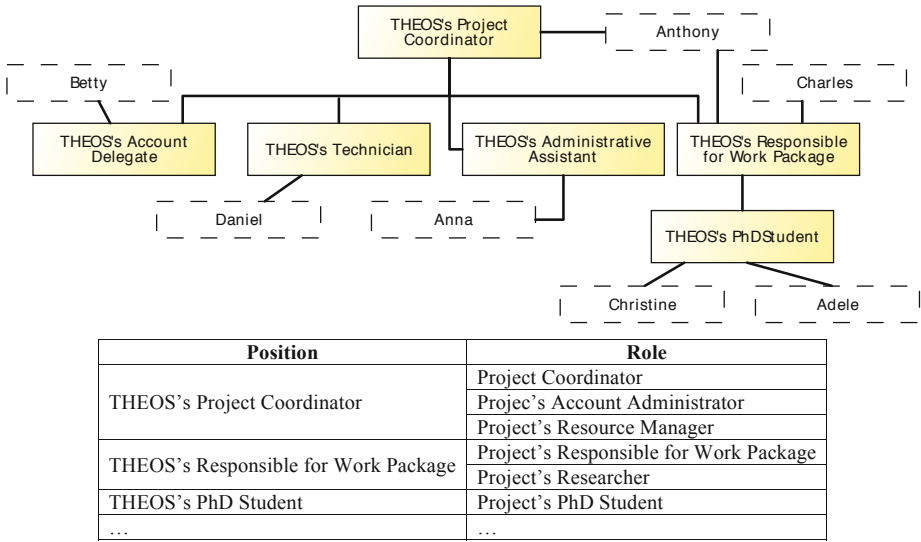


Fig. 3. Excerpt of the organizational model of ISA Group from a project perspective

assignment expression for the *BoundRole*, depends on the language supported by the BP modelling notation used. For instance, BPMN uses XPath⁶ by default to define resource assignments.

5 Using a RASCI Matrix to Specify Resource Assignments in BPMN Models

We already know all the information required to be able to automatically insert RASCI information into a BP model in order to make it compliant with the resource assignments of the matrix. As stated at the beginning of this paper, BP modelling languages existing nowadays do not provide a explicit way to model this RASCI-related information within BP models, being limited to the assignment referring to RASCI role Responsible (R) in most cases. Nevertheless, notations such as BPMN 2.0 offer extension mechanisms that may allow the introduction of any type of information into the models and, thus, we can make use of those features to add RASCI information [5].

We are working with BPMN [3] because it is the de-facto standard for process modelling, and because its extension capabilities are sufficient to enable the addition of the RASCI information we need to insert into the process models.

In the following, we introduce our approach to make a BPMN model RASCI-aware in a generic and automatable way. Furthermore, the output BP after the transformation from RASCI to BPMN is BPMN-compliant and has the required information to be executed in existing BPMSs.

⁶ <http://www.w3.org/TR/xpath/>

5.1 Resource Assignment Expressions in BPMN 2.0

Although swimlanes seem an easy and quick way to assign resources to the activities of a BPMN model, they are not a convenient form to do it. The problem basically relies on the lack of specific semantics for pools and lanes, which makes them remain an element for pure visual organization of the process elements in the model, as stated in the BPMN specification [3]. BPMN actually manages resources at activity level, using by default XPath expressions to specify resource assignments, although it permits the use of other languages. Making use of that feature, in this paper we use Resource Assignment Language (RAL) as an alternative to XPath to build resource assignment expressions.

RAL is a Domain Specific Language (DSL) specifically developed to express resource assignments in BP activities [7]. It is based on the same organizational meta model we used in Figure 2, which was defined by Russell et al. as basis to describe the so-called Workflow Resource Patterns (WRPs)[8]. These are a collection of patterns aimed at capturing the various ways in which resources can be handled in workflows (WFs). RAL expressions range from very simple assignments based on specific individuals of the company, to complex assignments containing access-control constraints (e.g. SoD) between activities, as well as compound expressions. For instance:

RAL 1: IS Anna

RAL 2: NOT (IS PERSON WHO DID ACTIVITY SubmitPaper)

RAL 3: (HAS ROLE ProjectCoordinator) OR (HAS UNIT ResearchGroup)

Therefore, RAL allows expressing role-based assignments to specify the direct assignments extracted from RASCI matrices, as well as binding information:

- *Organizational unit context.* To state that one RASCI role has to be performed by an organizational role within a *concrete* organizational unit, we could use RAL expression HAS UNIT UnitName, or expression HAS UNIT IN DATA FIELD PathToData.DataField in case the information must be retrieved from a file stored in the Information System (IS) used in the company. Notice that the specific way to access data stored in the IS depends on the implementation of the BPMS where the process is used, and so is the way to express the path to the required file.
- *Additional restrictions.* RAL offers expressions to specify many types of constraints, such as expression IS PERSON WHO DID ACTIVITY ActName to indicate BoD of a RASCI role with respect to the performer of another activity, i.e. both RASCI roles have to be allocated to the same person. Similarly, the negation of the previous expression can be used to define SoD. Expressions such as SHARES SOME ROLE WITH PersonName and HAS CAPABILITY CapabilityName allow the specification of other kinds of restrictions.

The reasons why we use RAL instead of XPath are:

- Even though XPath is a standard, it is barely supported by current BPMSs, which usually implement resource assignments in an ad-hoc fashion.

- XPath is not conceived specifically for resources, so it is difficult to use it to define resource assignment expressions.
- Derived from the previous point, XPath does not allow to express some binding information such as SoD constraints or skill-based restrictions. We refer the reader to [7] for further information about how to use RAL with BPMN 2.0.

Besides, RAL's formal semantics based on Description Logics (DLs) [9] provides it with powerful analysis capabilities, which could be useful in case we subsequently wanted to extract and analyse RASCI information included in the resulting BP model [10]. This analysis consists of answering questions related to how resources are being managed in a BP. For instance, in a RACI scenario, we could analyse the BPMN model resulting from our approach to check: (i) whether there can be any person responsible (or accountable) for all the activities of the process after allocation; or (ii) all the activities in which some person may participate somehow (with any RASCI role); among others. The design-time analysis of RAL expressions and its implementation within BPMN was described in [10]. Run-time analysis has already been developed and is being tested as part of a Business Process Management System (BPMS) called Activiti⁷.

5.2 Generation of RASCI-Aware BPMN Models

In this section we introduce a collection of transformations that can be used to include information coming from a RASCI matrix (plus binding data) in a resource-unaware BPMN model. The BP activities that appear in the RASCI matrix are changed into a sub-process with the name of the activity. All the RASCI information will be contained in the sub-process. We will sometimes refer to such a sub-process as *RASCI sub-process*. Within it, for each RASCI role it is necessary to indicate:

- The control flow elements required, with the *name convention pattern* we will use to make the transformation as automatic and generic as possible.
- The proper resource assignment expression associated to each new task. This expression comes from class *BoundRole* of the RASCI meta model (cf. Figure 2). Using RAL, the expression will be (i) (HAS ROLE Role IN Unit_InUnitExpression) AND (ResourceExpression), if there is a *Unit-Expression*; (ii) (HAS ROLE Role) AND (ResourceExpression), otherwise.

We assume that there is only one person responsible and one accountable for each activity (cf. meta model in Figure 2). Furthermore, the approval action (RASCI role A) takes place after the completion of the work developed for the activity, and only then the notification action (RASCI role I) can be performed. We could opt for a different order of RASCI roles or, even, for allowing them at different phases of the activity life cycle (for instance, to inform also before the start of the task or during execution). However, in the latter case, changes should be made in the RASCI matrix, and it is out of the scope of this paper.

⁷ <http://activiti.org/>

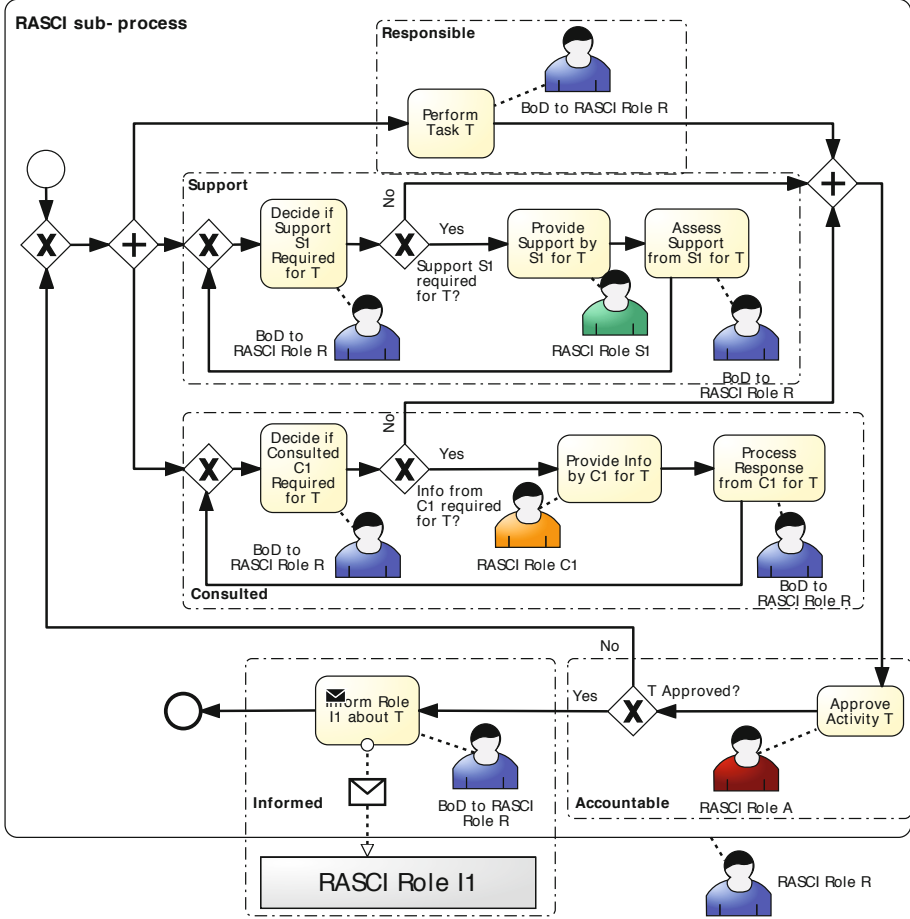


Fig. 4. Overview of a RASCI sub-process

The overview of a RASCI sub-process is depicted in Figure 4. BPMN groups define the process fragments related to RASCI roles. In case a RASCI role does not participate in the activity, the corresponding process fragment will be omitted in the sub-process. If, on the contrary, there are several roles performing a RASCI role, the associated process fragment will be added for everyone of them. We will use activity *Register at Conference* of our RASCI matrix (c.f. Table 1) to explain the transformations. Figure 5 shows the RASCI sub-process for it.

Responsible (R). This is the only RASCI role whose resource assignment expression is associated to the RASCI sub-process itself, i.e. for activity *Register at Conference*, the new sub-process has the following RAL expression: (HAS ROLE ProjectsPhDStudent) AND (IS PERSON WHO DID ACTIVITY SubmitPaper).

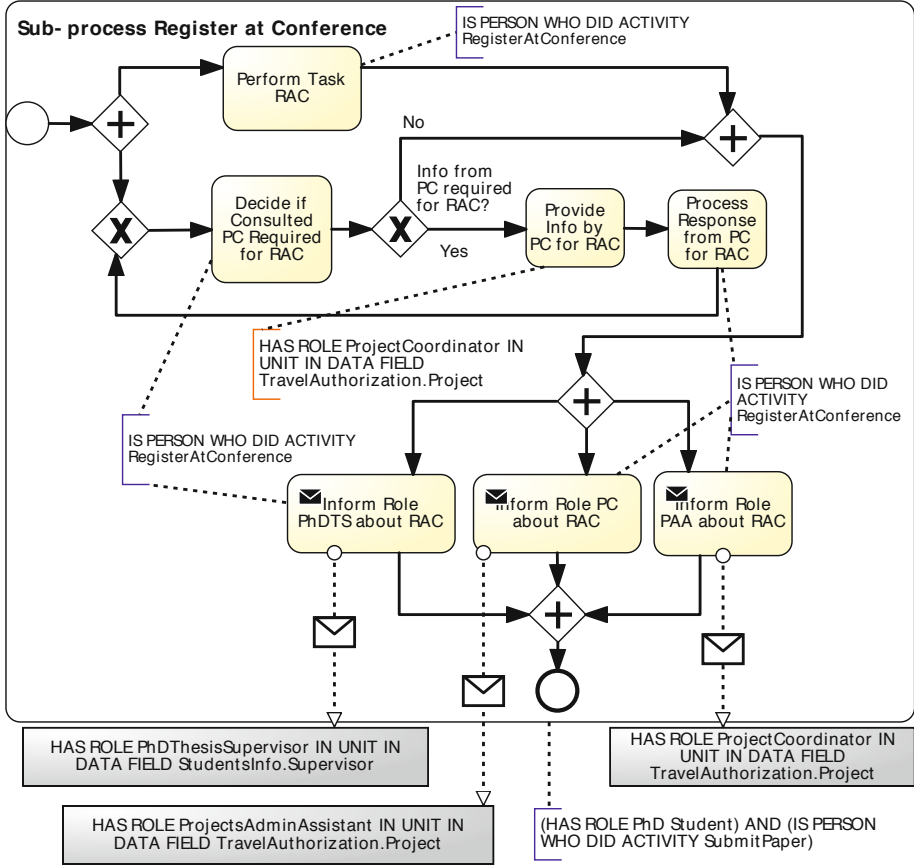


Fig. 5. RASCI sub-process for activity *Register at Conference* of the BP in Figure 1. Acronyms have been given for activities and roles for the sake of visualization.

Nonetheless, task *Perform Task ActivityName* is introduced in the RASCI sub-process to represent the actual work to be completed for the activity. This task is directly assigned to the performer of the sub-process, i.e. the RAL expression for task *Perform Task Register at Conference* is **IS PERSON WHO DID ACTIVITY RegisterAtConference**. This allows every element within the subprocess to make reference to the performer of the activity being sure that there is already an allocated performer. Note that if the activity itself were a sub-process, then *Perform Task Register at Conference* would be *that* sub-process.

Accountable (A). To model this RASCI role we insert a new task into the RASCI sub-process named *Approve Activity ActivityName*, in charge of approving the work developed in the activity at hand. Moreover, we have to add the control flow required to go back to the beginning of the sub-process in case the activity was not approved, by means of an *XOR join* gateway. The assignment expression of RASCI role A will be assigned to the new task.

This process fragment can be omitted only if R and A are assigned to the same organizational role in the RASCI matrix, and the binding information for A consists of a BoD with respect to R (i.e. BoD between A and R). For size reasons in Figure 5, we assume this is satisfied in our example activity. Note that if the previous condition is met and no other RASCI role participates in the activity, then the whole RASCI sub-process can be omitted and, thus, the result of the transformation is the same initial activity with the resource assignment expression corresponding to R.

Support (S). Inserting this RASCI role is not as straightforward for several reasons: (i) support is not mandatory. It is a decision of the person in charge of the task whether support is required to complete the work; (ii) it is said nowhere that support cannot be requested more than once to the same organizational role associated to RASCI role S in the matrix; (iii) it is inherent to term support that the work performed by the person “external” to the task must be evaluated by the resource in charge of the task (R) in order to decide whether the goal of the support has been achieved and/or whether more support is required; and (iv) it is not evident at which moment in task execution, support can be requested. In this sense, we have to make some decisions. So, we propose the control flow structure depicted in Figure 4 for RASCI role S, basically composed of tasks *Decide if Support Role Required for ActivityName*, *Provide Support by Role for ActivityName* and *Assess Support from Role for ActivityName*, and a couple of XOR gateways. The task targeted at providing the required support is assigned to the organizational role performing RASCI role S in the matrix with the appropriate resource assignment expression. The rest of new tasks belong to the person that is responsible for the activity. Thus, the resource assignment expression for these tasks is a BoD constraint in RAL: `IS PERSON WHO DID ACTIVITY ActivityName`. According to our RASCI matrix, activity *Register at Conference* does not need support.

Consulted (C). The translation of this RASCI role into BPMN language is very similar to RASCI role S. As depicted in the figure, the structure introduced in the sub-process is the same, as well as the considerations to be made. The only difference between the application of the two RASCI roles is in the name of the tasks involved, and in the semantics of tasks *Provide Support by Role for ActivityName* and *Provide Info by Role for ActivityName*. The person in charge of the latter is not so much compromised with the global activity, as his/her involvement is limited to providing certain information. For activity *Register at Conference*, the RAL expression associated to the task performed by the project coordinator, could be `HAS ROLE ProjectCoordinator IN UNIT IN DATA FIELD TravelAuthorization.Project`. For the rest of tasks of this process fragment, the associated RAL expression is `IS PERSON WHO DID ACTIVITY RegisterAtConference` (cf. Figure 5).

Informed (I). This RASCI role has a very special difference with respect to the others. The organizational role indicated in the matrix is the target person of

the notification action, not the performer like in the rest of cases. The problem is that in BPMN we do not have a way to specify the resource “affected” by a task. However, there is a mechanism based on message interchange to communicate information to people working on other processes. The key point here is whether the informed person can be considered an external participant or not. In case that role does not participate in any other activity of the BP, it is undoubtedly somebody external to the process. Thus, we could use messages to send the notification. Otherwise, that person may have his own assigned tasks in the process, and we do not have a way to notify something to that person without interrupting his work flow. We believe that, given the RACI definition for I, it is reasonable to consider it an external participant of the process under any circumstances, due to the absence of collaboration from his part. Independently of his responsibilities with respect to other BP activities, for *that* activity in question he is a target, not an executor. Therefore, we will introduce task *Inform Role about ActivityName* to represent an activity that sends a message to a collapsed pool representing RASCI role I (cf. Figure 4). This task is assigned to the person in charge of the main activity, like in RASCI roles S and C. For activity *Register at Conference*, there are three informed people. The control flow and the RAL expressions for them are directly shown in Figure 5.

Following these rules we can convert the initial BP model into another one with the information necessary to implement RASCI in our organization. Furthermore, the resulting model is very clean from the visualization perspective, in the sense that it is very similar to the initial one. So, the overall understandability of the initial process is maintained. The real complexity related to the RASCI information is found only when the RASCI sub-processes are opened.

6 Implementation

An overview of the prototype we have developed for the approach is shown in Figure 6. RACI2BPMN module receives the BPMN description of a resource-unaware BP model as an XML file, and the representation of the associated RASCI matrix together with the required binding information in a JSON file. Applying the transformations explained in Section 5.2, the tool automatically generates a BPMN model in which the previous tasks are now collapsed sub-processes containing the RASCI-related information, i.e. the required control flow and the proper assignment expressions. As depicted in the figure, the appearance of the resulting model is very similar to the initial process. We have used RAL to write the resource assignment expressions because of its advantages with respect to other options such as XPath (cf. Section 5.1). Nonetheless, a requirement for the design of the prototype has been to allow the easy modification of the language used for the assignment expressions.

The BPMN generated can be manipulated in any BP modelling tool, such as Oryx [11], and it can be executed in BPMSs such as Activiti, in which we have previously included the required functionality to process RAL expressions.

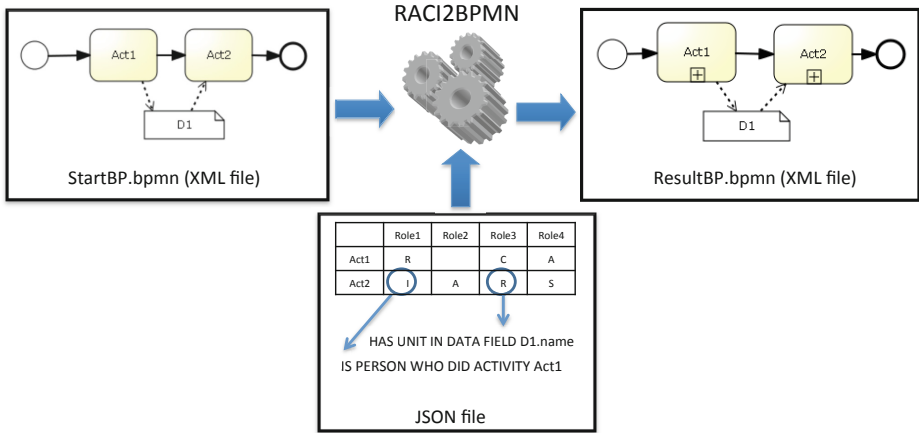


Fig. 6. Overview of the RACI2BPMN prototype

Notice that a pre-processing of the resulting BPMN model could be performed before launching the process in order to make sure that the task labels in the RASCI sub-processes conform to the name convention patterns established.

Further information about the prototype and examples can be found at www.isa.us.es/cristal.

7 Related Work

Human resource management in BPs is an appealing challenge that is recently catching much attention in academy. However, existing approaches are mainly focused on indicating who must perform the BP activities and/or on introducing the so-called access-control constraints (e.g. SoD, BoD) in WFs and BP models [7, 12–15], but they leave aside the rest of responsibility functions involved in RACI matrices.

Last year, we gave a step forward in the merger of RASCI matrices and BPMN, introducing a collection of *RASCI patterns*, and an extension for BPMN 2.0 [3] to allow the modelling of such patterns [5]. Unlike our current approach, that work was purely centered at design time and the assignment of organizational roles to RASCI roles in the BPMN model was done by means of the swimlanes. Now that RAL has a well-defined semantics, we believe it is a good alternative to perform the resource assignments at activity level, as proposed by BPMN 2.0. To the best of our knowledge, there are not other proposals so far pursuing the same goal we are chasing [5]. Indeed, as derived from recent studies, the handling of RACI matrices may become a very tough task [16].

In the market, on the contrary, it is noticeable the increasing interest of companies and software developers in the use of RACI responsibility management with BPs. Academic Signavio⁸ has recently added capabilities to indicate what

⁸ <http://www.signavio.com/en/academic.html>

RACI role each resource assignment associated to a task refers to, and to generate the RACI matrix automatically from the BPMN model. However, the solution is ad-hoc, it is “just” a matter of design (without any semantics), and the generation of the RACI matrix is something trivial having all the information already in the model. The same is allowed in and applies to iGrafx for Enterprise Modeling⁹. Other tools such as YAWL [17], WS-HumanTask [18] and BPEL4People [19] do not provide support for RACI, being their improvements on resource management mainly oriented to the introduction of capabilities for task delegation, re-allocation, team work, and the like (see the definition of the WRP for further information about resource management in WFs [8]).

8 Conclusions and Future Work

From this work we can conclude that it is possible to join RASCI information with BP models, and so be able to perform *complete* responsibility management in our organization. Furthermore, it is possible to do so by generating a BP model containing all the required information about resources (i.e. resource assignment expressions including the binding information necessary) to be executed by a process engine (i.e. a BPMS) with no need for any changes. In this sense, with our approach we outperform the scope of previous work focused on the use of BP swimlanes, which actually do not contribute in the generation of an executable BP due to their lack of semantics, and the limited flexibility as for expressing resource assignments. Our solution gets to keep the resulting BPMN as “clean” as possible from the viewpoint of the initial process, since all the RASCI information is at sub-process level. Besides, by decoupling responsibility management and BP management, the problem of how to maintain the consistency between the information in RASCI matrices and in BP models is already solved. Any change regarding resources can be automatically applied to the BP model by performing the transformation rules we introduced in this paper. This, thus, avoids synchronization problems.

The approach presented in this paper could be specially useful to those organizations that are not using BPs yet but want to move to a process-oriented style, and use RACI to manage responsibility. Our transformations could help them build RACI-aware BP model directly by re-using the information they have in the RACI matrices.

The analysis of RASCI matrices regarding resource management, as well as the resource assignment expressions resulting from the transformation, is part of our future work in this line, for which we already have some results.

References

1. Smith, M.: Role And Responsibility Charting (RACI). In: Project Management Forum (PMForum), p. 5 (2005)
2. Conchúir, D.O.: Human Resource Management Processes. In: Overview of the PMBOK Guide, pp. 129–145. Springer, Heidelberg (2011)

⁹ <http://www.igrafx.com/products/process4em/>

3. BPMN 2.0, recommendation, OMG (2011)
4. ARIS, RACI. ARIS Community's Website (2012), <http://www.ariscommunity.com/raci>
5. Cabanillas, C., Resinas, M., Ruiz-Cortés, A.: Mixing RASCI Matrices and BPMN Together for Responsibility Management. In: VII Jornadas en Ciencia e Ingeniería de Servicios (JCIS 2011), vol. 1, pp. 167–180 (2011)
6. Russell, N., ter Hofstede, A., Edmond, D., van der Aalst, W.: Workflow Resource Patterns. Tech. rep., BETA Working Paper Series, WP 127, Eindhoven University of Technology, Eindhoven (2004)
7. Cabanillas, C., Resinas, M., Ruiz-Cortés, A.: RAL: A High-Level User-Oriented Resource Assignment Language for Business Processes. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 50–61. Springer, Heidelberg (2012)
8. Russell, N., van der Aalst, W.M.P., ter Hofstede, A.H.M., Edmond, D.: Workflow Resource Patterns: Identification, Representation and Tool Support. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 216–232. Springer, Heidelberg (2005)
9. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: The Description Logics Handbook: Theory, Implementations, and Applications. Cambridge University Press (2003)
10. Cabanillas, C., Resinas, M., Ruiz-Cortés, A.: Defining and Analysing Resource Assignments in Business Processes with RAL. In: Kappel, G., Maamar, Z., Motahari-Nezhad, H.R. (eds.) ICSOC 2011. LNCS, vol. 7084, pp. 477–486. Springer, Heidelberg (2011)
11. Decker, G., Overdick, H., Weske, M.: Oryx – An Open Modeling Platform for the BPM Community. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 382–385. Springer, Heidelberg (2008)
12. Bertino, E., Ferrari, E., Atluri, V.: The specification and enforcement of authorization constraints in workflow management systems. *ACM Trans. Inf. Syst. Secur.* 2, 65–104 (1999)
13. Awad, A., Grosskopf, A., Meyer, A., Weske, M.: Enabling Resource Assignment Constraints in BPMN. Tech. rep., BPT (2009)
14. Wolter, C., Miseldine, P., Meinel, C.: Verification of Business Process Entailment Constraints Using SPIN. In: Massacci, F., Redwine Jr., S.T., Zannone, N. (eds.) ESSoS 2009. LNCS, vol. 5429, pp. 1–15. Springer, Heidelberg (2009)
15. Strembeck, M., Mendling, J.: Modeling process-related RBAC models with extended UML activity models. *Inf. Softw. Technol.* 53, 456–483 (2011)
16. Bronkhorst, J.: RACI matrices - how difficult can it be? HP's Website (June 2010), <http://h30507.www3.hp.com/t5/ITILigent-Service-Management/RACI-matrices-how-difficult-can-it-be/ba-p/41138>
17. Adams, M.: YAWL v2.3-User Manual. Tech. rep., The YAWL Foundation (2012)
18. Web Services-Human Task (WS-HumanTask) v1.1. Tech. rep., OASIS (2010)
19. WS-BPEL Extension for People (BPEL4People). Tech. rep., OASIS (2009)