Generation of stakeholder-specific BPMN models

Diogo Cardoso

University of Lisbon, Lisbon, Portugal
Av. Rovisco Pais, 1049-001 Lisboa
diogocardoso@tecnico.ulisboa.pt

Abstract. A business process model always has a dominant perspective in the detriment of others, motivating the need of different stakeholders to look for different models of the same process. In fact, it is common to see that in different units of an organisation, such as quality, audit, risk or human resources, there are different models of the same processes, each focusing on specific aspects. Unfortunately, these models tend to lack consistency because of the effort required to keep them consistent. To tackle this problem, we are developing an approach that aims to generate stakeholder-specific models on the fly, based on some arbitrary stakeholders' concerns. We derive the generated models from a consolidated business process model, which is previously designed, and its organisational taxonomy, thus ensuring the consistency between the generated models.

Keywords: Business Process Modelling, Process Views, Process Repository, BPMN

1 Introduction

Business processes are designed to achieve specific goals and the task of business process modelling is expected to improve the understanding and communication across the different stakeholder groups [9]. However, these goals are difficult to achieve because organisations often have to manage multiple process diagrams that represent the same business process, which can lead to several inconsistencies, such as heterogeneous schemes for naming its activities and entities, usage of different modelling styles and process hierarchies with arbitrary depth and level of detail [4]. We argue that these issues arise due to two main reasons [15, 16]:

- On the one hand, business processes often cross multiple organisational units and also tend to cross inter-organisational boundaries. Therefore, process models are often shared among different stakeholders, which have contrasting concerns and focus on distinct perspectives of the same business process, such as performance, auditing, information systems, people or compliance.

– On the other hand, a business process model is a representation of the modellers perspective regarding a given process: different modelling teams may choose different ways to model the same business process according to their interests or focus. Thus, they will most likely achieve different specifications for the same process.

Having different models of the same business process is, nonetheless, beneficial for the organisation since each organisational unit has its own concerns and has a process representation that suits these concerns. However, keeping the various models consistent can be a very demanding task. It can be even harder when there are frequent changes to the business process since all the models must be updated accordingly.

To facilitate the consistent modelling of business processes from different perspectives, this paper presents an approach that enables to generate views of a common business process model according to the requirements of its stakeholders. Hence, our approach can be considered an application of ISO 42010 [10] to business process modelling. ISO 42010 states that a view addresses one or more of the concerns of the system stakeholders. A view is a partial expression of a systems architecture with respect to a particular viewpoint. A viewpoint establishes the conventions by which a view is specified, depicted and created.

To ensure that the views are inherently consistent, our approach is limited to the generation of views from a consolidated model that must be previously designed. The design of this consolidated model is out of the scope of this paper but is presented in [6].

The generation of the views is based on arbitrary stakeholders concerns. These concerns, which we call dimensions, are mapped to each of the process activities. For example, one could assign to each activity its level of risk, its actors, location, etc. Furthermore, a hierarchical structure, that we call taxonomy tree, is associated to each dimension and allows the existence of various levels of detail: a possible taxonomy tree for a dimension depicting process participants is an organisational structure. It is the tweaking of these dimensions and respective levels of detail that allows the creation of different views of a business process.

The expected contribution of our work is twofold. First, to provide an approach for the generation of business process views specified in *BPMN 2.0*[12]. Second, to support the stakeholders in the task of business process modelling by proposing a systematic way of representing their concerns.

The remainder of this paper is structured as follows: the next section introduces a scenario that is used to illustrate the problem and the approach. Section 3 reviews relevant background and related work. Section 4 explains our research methodology. Section 5 describes our approach and section 6 shows examples of its application using the developed tool. The research validation is debated in section 7. We discuss conclusions and future work in section 8.

2 Illustrative Scenario

This section describes the research problem by presenting the illustrative example of an automobile repair company, with the intention of promoting the readers understanding of the problem. This scenario will be used throughout the paper:

"The ACME automobile repair company specialises in bodywork repairs. When a damaged car arrives at the ACME's garage, its service manager assesses the vehicle damage. Based on the service manager's analysis, a panel beater planishes the damaged parts or goes to the company's warehouse to pick up replacement parts or both. After the body work repair, a painter prepares the car for spraying. The spraying is done on the company's painting greenhouse, where the service manager also inspects the quality of the finished job. If he/she believes the quality is subpar, the painter resprays the vehicle and it is inspected again. Otherwise, the car is ready to be delivered. This process is monitored by two departments: the Human Resources (HR) department and the Facilities Department. To perform this monitoring, each department models their own view of the process focused in the resources that each has to manage: actors for the first and premises for the second department."

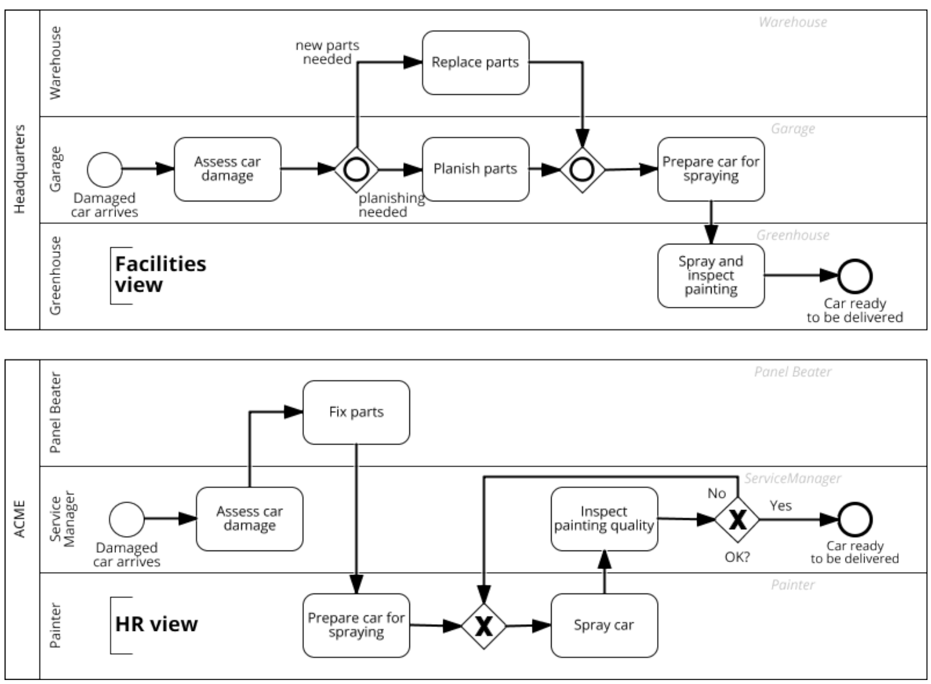

Fig. 1. BPMN process views of the car repairing process, designed by the Facilities department (top) and HR department (bottom).

The process is performed by three actors, in three distinct locations and has six activities. Figure 1 shows the business process views designed by the HR and Facilities department. On the one hand, the HR department grouped the "Replace parts" and "Planish parts" activities because they are both performed by the panel beater. On the other hand, the facilities department grouped the "Spray car" and "Inspect painting quality" activities since they are performed in the same location (greenhouse). We refer to the former as the *Who* dimension and the latter as the *Where* dimension.

In the scope of our problem, the question that now arises is: "How do we model additional views of the process or update any of the existing views while maintaining consistency across all views?". In this very simple scenario, a process change would be easy to keep up with. However, in more complex processes (with more decision gateways, exception flows, etc.) it would take a lot of effort to keep consistency across views. To our knowledge, there are no mechanisms to simplify the work of keeping consistency across views nor to assist the creation of new views, thus making these tasks costly and inefficient.

3 Related Work

Our research is closely related with business process variability modelling. [18] shows there have been significant research efforts in the past decade in the area of variability modelling. [18] classifies a process variability modelling approach based on how it captures the relation between a set of elements of a process and the corresponding elements in its variants. This classification resulted in four groups: node configuration, element annotation, activity specialisation and fragment customisation approaches. The approaches that use element annotation and activity specialisation are the ones that most closely resemble the business process view generation approach that we outline in section 5.

On the one hand, element annotation approaches, like [7] and [17], rely on the annotation of model elements with properties of the application domain: in our case, we annotate the stakeholder concerns to the process activities. On the other hand, activity specialization approaches, like [11] and [2], assign variants to the process activities. In our work we use a different, but also hierarchical, abstraction technique: functional decomposition, i.e. recursively breaking down a process as sub-activities.

Regarding business process views, [21] also highlights the existence of conflicting process specifications for the same organisational process, depending on the distinct stakeholders perspectives and on the modellers view of that particular process. That work defines the rules for identifying business process activities by applying the Zachman Framework [23]. The Zachman framework describes an architecture using a two-dimensional classification matrix based on the intersection of six contextual dimensions (what, where, when, why, who and how) with six rows according to reification transformation that represents a view of the solution from a particular perspective such as the planners or the designers perspective [22, 23]. However, [21] does not use such specification but only the six communication questions enumerated above as independent concerns for the decomposition of a business process [19, 4]. Thus, each of these six dimensions focuses on a specific and independent concern. The combinations of these concerns characterise aggregate parts of the process or the process as a whole [5]. These criteria for activity decomposition support business process modelling by facilitating the task of different stakeholders consistently modelling the same process. The application of such rules is the basis of our approach.

Pereira [14, p.134-137] and later Caetano [3] continued in the same direction of Sousa [21], basing their approach on the use of the contextual dimensions of the Zachman Framework to portray stakeholders concerns and adding important contributions towards facilitating the generation of process views. Namely, it proposes the arrangement of the concepts associated with each dimension into a hierarchical structure: a taxonomy tree. Our work will fill the gap in knowledge existing in these works because they focus on describing a conceptual tool without formally defining the algorithms that support the generation of the views. Moreover, they do not apply the problem to a specific process modelling notation, like *BPMN*.

Finally, as a complement to our approach, the method proposed by [6] aids process stakeholders in integrating business process views into a process repository, also using the six dimensions. In this case, they are used to guide the annotation of the process elements by the stakeholders. This method is used to populate a process repository which serves as the knowledge base for our approach: we further detail this relation in section 5.

4 Research Methodology

Our research fits in the design science paradigm [8], as we try to solve our problem through the development of new artefacts. Thus, we propose to use the design science research methodology (*DSRM* [13]) to guide our research. The *DSRM* is an iterative research methodology that focus on the creation and validation of artefacts that address a research problem and is divided in six phases.

The first two phases, which consist of identifying the problem, motivation and defining the research objectives, are materialised in the first two sections of this paper. The third phase (design and development) encompasses the definition of the approach that supports the generation of business process views, which is briefly described in the following section. The fourth phase (demonstration) includes the development of a tool that implements the generation algorithm. Example use cases of this tool are shown in section 6. The fifth phase (evaluation) is discussed in section 7 and sees the application of the tool in a real use case with the objective of validating our work. Lastly, the final phase (communication) involves the production of an additional paper to showcase our results.

5 The Approach

As previously stated, this work in progress is complemented by the approach presented in [6]. Thus, we start this section by briefly describing how both works are related and then proceed to explain in more detail our approach.

5.1 The Big Picture

Colaço and Sousa [6] define a method for merging distinct views of a business process into a single, consolidated business process model. The method also defines what we call an organisational taxonomy, which is a taxonomy tree for each dimension. A taxonomy tree is a collection of concepts organised into a hierarchical structure. The method guides the process stakeholders in constructing these trees by classifying process activities according to the Zachman contextual dimensions. However, other dimensions representing different concerns can equally be considered in this classification method, as for example risk and security, among others. The resulting mappings are stored in a process repository.

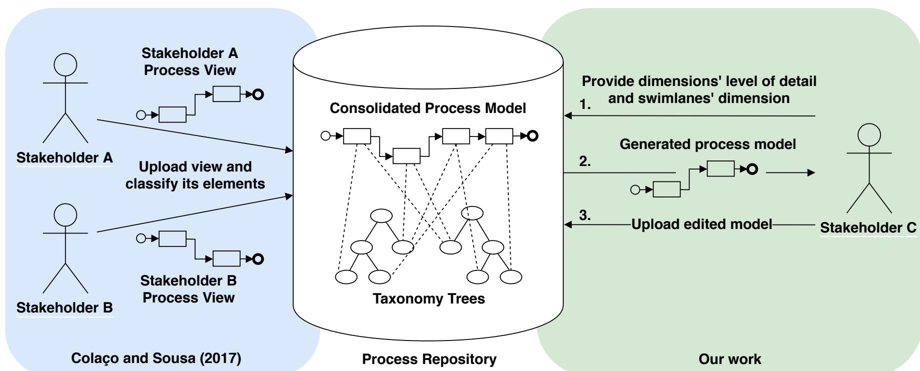

Fig. 2. Illustration of the relation between [6] and our work in progress.

Afterwards, using the approach we are developing, the stakeholders may generate process views, simply by providing the level of detail desired for each dimension and, if they want to represent swimlanes, the dimension they want to depict in the swimlanes. Furthermore, the stakeholder can choose to edit the generated model: rearrange the location of the graphical elements and change the generated activities names. This can then be uploaded to the repository to keep the naming and positioning of the generated activities in future requests. Such information is cleared whenever the corresponding activities are removed or changed in the process repository. Figure 2 resumes the relation between both works.

5.2 The Process Repository

Being the central element of our solution, we describe its content and structure that is relevant for the work presented here:

- An organisational taxonomy, which holds a hierarchic structure (tree) for each of the dimensions;
- The consolidated *BPMN* process model;

– The mapping between the activities of the consolidated process model and the leaf nodes of the taxonomy trees.

The meta-model of the repository is presented in figure 3, using an *UML* class diagram. A Process is composed of Flow Elements. A Flow Element can be an Activity, Gateway or Event and is connected to other Flow Elements by sequence flows (represented by the Flow Element class self-association). A Process also aggregates, for each dimension applied in the generation of views of that process, the root of the dimensions taxonomy tree, which is a Taxonomy Node. Taxonomy Nodes in turn aggregate other Taxonomy Nodes and the leafs of the taxonomy trees classify each Activity of the Process (represented by the association between the Taxonomy Node and the Activity classes).

In the current state of affairs, we only support swimlanes and the *BPMN* elements presented in the figure 3. Since the *BPMN* standard allows a wide scope for the swimlane elements (i.e. pools and lanes) [12], we derive them from the associations between the taxonomy trees node and the process activities. It is up to the stakeholder to determine which dimension should swimlanes depict in the generated view.

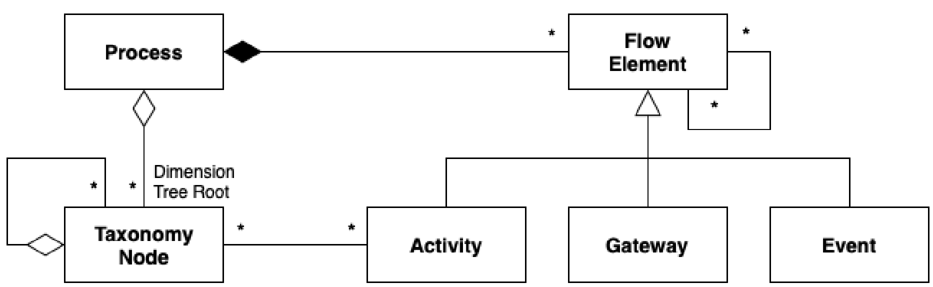

Fig. 3. UML class diagram of the process repository meta-model.

As an example of a process model based on the repository meta-model, figure 4 presents the information that would be present in the repository for the aforementioned car repairing process. For the sake of simplicity, Process and Flow Element objects are not represented.

5.3 Generation Algorithm

The algorithm developed to support the generation of views from the repository data is based on the application of a rule derived from one presented in [21]. That rule specifies that an activity α can be decomposed into two or more distinct discrete activities if and only if one of the conditions stated in Table 1 is satisfied.

In [21], there is one aggregation condition for every contextual dimension of the Zachman framework [23], as summarised in table 1. However, in our approach, the dimensions can be freely defined by the process stakeholders, and there are as many conditions as there are dimensions in the repository. i.e. the

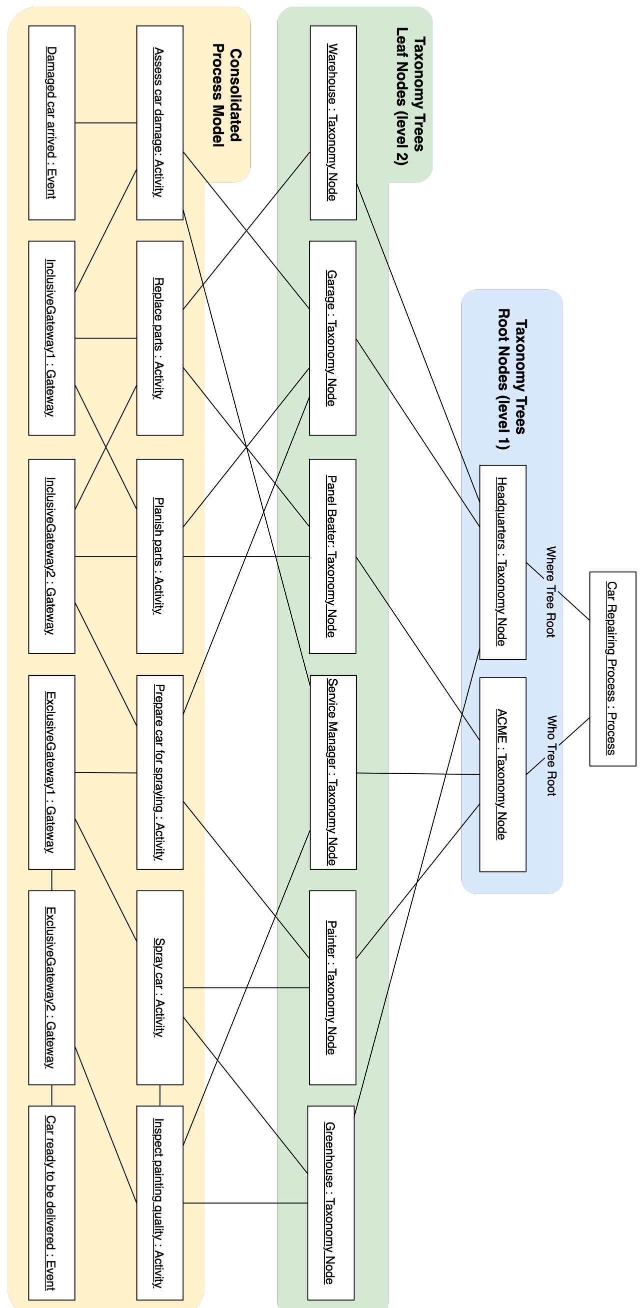

Fig. 4. UML object diagram of the repository content for the car repairing process.

Table 1. Criteria for activity decomposition presented in [21]

Dimension	Criteria
What	α is composed by two or more activities which receive/create different data entities.
How	α is composed by two or more activities which are processed using different applications.
Where	α is composed by two or more activities which occur in different locations.
Who	α is composed by two or more activities which are managed by different business actors.
When	α is composed by two or more activities which are performed in distinct periods of time.
Why	α is composed by two or more activities which exist to satisfy different purposes.

set of conditions is not fixed to those six dimensions. Our rule for aggregating process activities is presented below:

Two activities α and β can be aggregated into an activity δ if and only if for every dimension one of the following two conditions applies:

- *the taxonomy concepts mapped to α and β are the same*
- *the taxonomy concepts mapped to α and β are different but their ancestor at the chosen level of detail is the same*

Using the car repairing process as an example, if one chooses the *Where* dimension at level of detail 2 and the *Who* dimension at level of detail 1, two given activities can only be aggregated if they are both performed in the same location (garage, warehouse or greenhouse).

Our generation algorithm iterates the consolidated process model to identify patterns to which it applies the activity aggregation rule. All the activities contained in a piece of process flow that matches a pattern are then evaluated against the aggregation rule. If the rule can be applied, the matched process flow is grouped into a single activity. It can take several iterations to generate the final view because new patterns may be generated in each iteration. The algorithm stops when one can no longer apply any aggregation rule during an entire iteration. Figure 5 shows the three patterns considered. These patterns are composed of some of the patterns identified in [1].

Pattern A is the simplest. Any two sequential activities that respect the aggregation rule can be grouped into a single activity. If any, the intermediate events between the activities are also aggregated into the resulting activity.

Pattern B refers to the splitting of the process flow into an unspecified number of branches which must all join at the same gateway. The gateway type is not relevant. The branches must only contain at most one activity and any number of events. If the activities of all branches respect the aggregation rule, then both gateways and the branches can be grouped into a single activity.

Pattern C depicts the classic rework pattern. Both the main branch and rework branch must contain at most one activity and any number of events. If both activities respect the aggregation rule, then both branches and gateways can be grouped into a single activity,

It is expected that not all process flows respect these patterns. When that happens, it is up to the modeller to group the process elements of the generated view as he/she sees fit.

In figure 6, a sample execution of the pattern matching is shown reducing the initial process into a single activity in seven iterations: the patterns matched

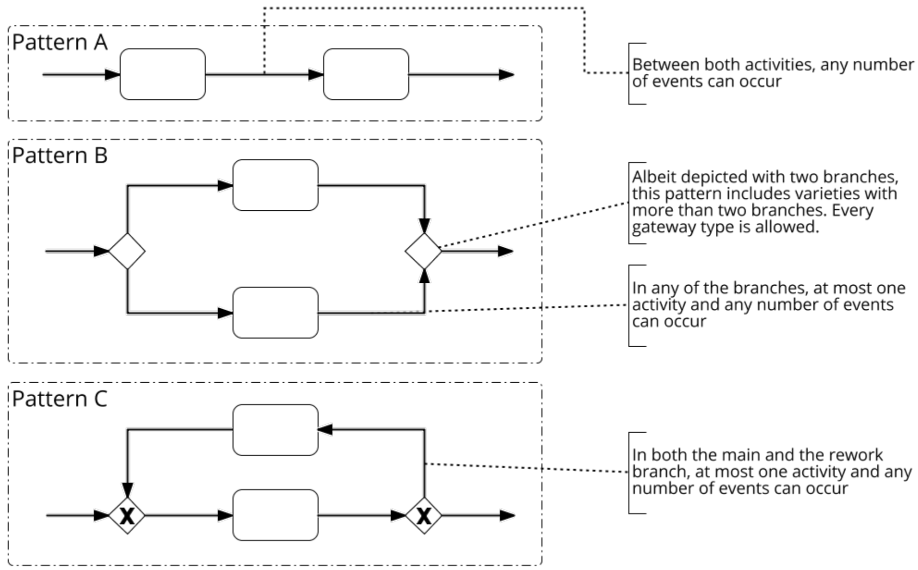

Fig. 5. Patterns that the generation algorithm tries to match during its execution.

in the first six iterations are represented in different colours and numbered from 1 to 6. Each time a pattern is matched, it is replaced by an activity which in turn may be part of a new pattern to be matched in the next iteration. The iteration stops when there is no further pattern matching. The view generated in this case is simply composed of an activity, preceded by the start event and followed by the end event.

In a final phase, the algorithm assigns a name to each generated activity, which is simply the aggregation of the names of the activities that originated it. Then, it looks for a matching stakeholder defined name and uses it instead, whenever one is found. This mapping between generated names and stakeholder given names is updated whenever the stakeholder changes and uploads a generated model and cleared whenever at least one of the corresponding activities are removed or changed from the process repository. In what concerns the activities positioning, a similar approach is taken. In the final layout, a stakeholder defined position is searched for each generated activity, and used whenever a position is found. The position information of generated models is cleared whenever one of the corresponding activities in the process model are changed or removed.

6 Demonstration

In this section, we provide a glimpse of the tool developed to support the generation of process views. Moreover, we also show some examples of generated views of the car repairing process that we believe will further help the readers understanding of our solution.

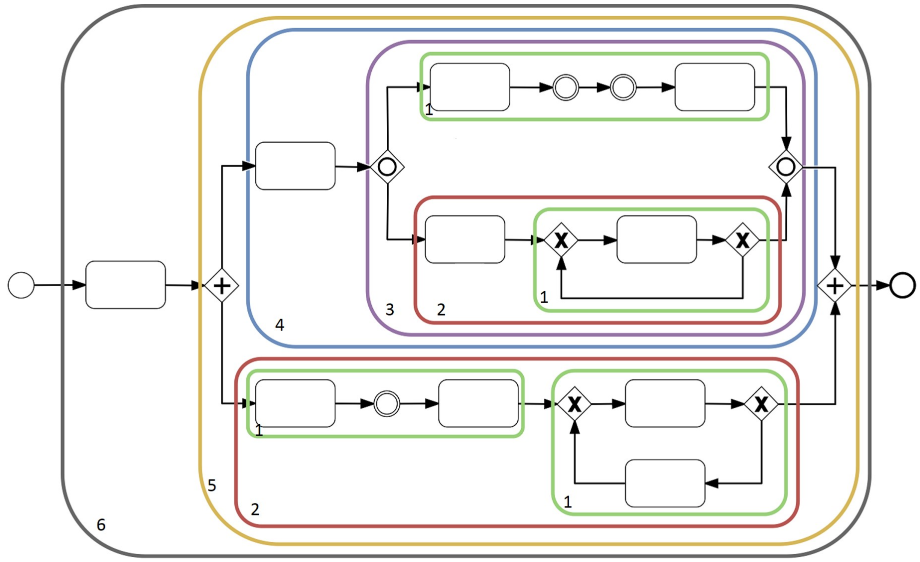

Fig. 6. Sample execution of the view generation algorithm. Pattern matching iterations are numbered from 1 to 6.

The tool was developed extending the *Atlas*[20] tool from Link Consulting[1] with the process view generator. It includes two major components:

- the **Process Repository**, holding the process model and the taxonomy tree for each dimension, is configured in *Atlas*. The construction of organisational taxonomies and the mapping of its concepts to the consolidated model is done solely through the process repository.
- the **Process Modeller**, was developed using the *bpmn.io*[2] library and implements the generation algorithm explained in section 5.3. The modeller integrates with the repository using *REST* services allowing the retrieval and upload of process data.

Regarding the generation of views, we show some examples of views in figure 8 that are generated from the repository content portrayed in figure 4. Only two dimensions are present: the *Who* and *Where* dimensions, each with a two-level taxonomy tree, representing process actors and locations, respectively.

View c) clearly focuses on the *Who* dimension as this dimension is represented with the highest level of detail (level 2), whereas the *Where* dimension was set to the lowest level of detail (level 1). Furthermore, swimlanes were used to represent the *Who* dimension. In turn, view b) was generated from the exact opposite input: the *Where* at the maximum level of detail; the *Who* at the lowest level of detail; and the swimlanes were chosen to represent the *Where* dimension.

[1] http://www.linkconsulting.com/atlas
[2] https://bpmn.io

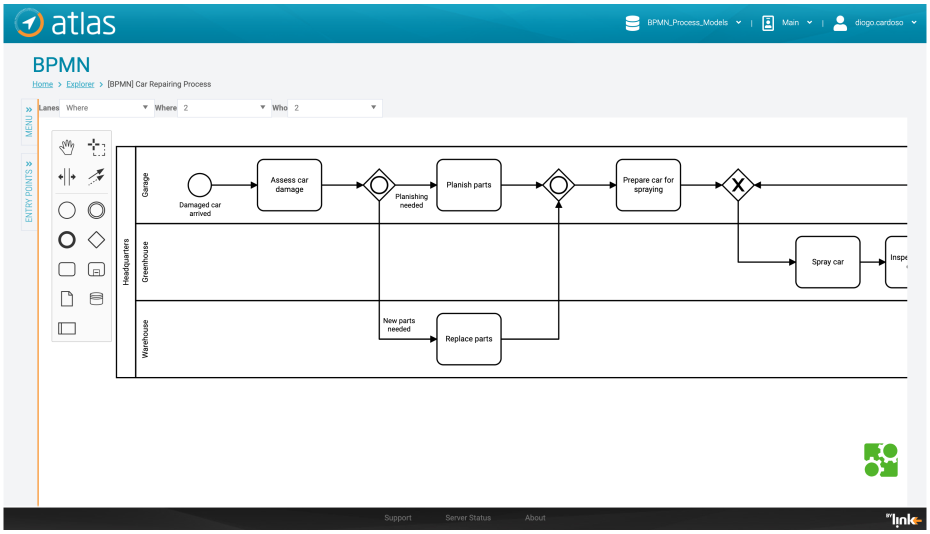

Fig. 7. Screen of the tool after the generation of a view.

Whereas in the previously described views the focus was on one dimension, in views a) and d) both dimensions were set to the lowest level of detail and to the highest and swimlanes represent the *Where* and the *Who* dimension, respectively. As expected, the higher the level of detail of the dimensions, the more the process activities are decomposed; if the lowest level of detail is chosen for all dimensions, there is no activity decomposition at all.

The name of a generated activity is simply the aggregation of the names of the activities that originated it. Nonetheless, as previously mentioned, the stakeholder may choose to edit the naming and positioning of the activities. These changes are kept in the repository and used in future generation requests.

7 Evaluation

To demonstrate and test the applicability and usefulness of our research work, we will apply the view generation approach in real-life cases study performed within real organisations, starting from a company in the automotive retail industry.

In this proof-of-concept, the first step is to select one business process involving several stakeholders and gather information about the chosen process from the various stakeholders, each stating its view. We follow the method proposed in [6] to merge the different process views from the different stakeholders and populate the Atlas process repository. In this process, the dimensions that better address the stakeholder's concerns will become explicit and defined, as well as the taxonomy tree for each dimension.

Afterwards, the participants will be asked to state their concerns and to generate views that better suit such concerns, and comment on the usefulness of the generated views, thus fulfilling the requirement of evaluation of our work.

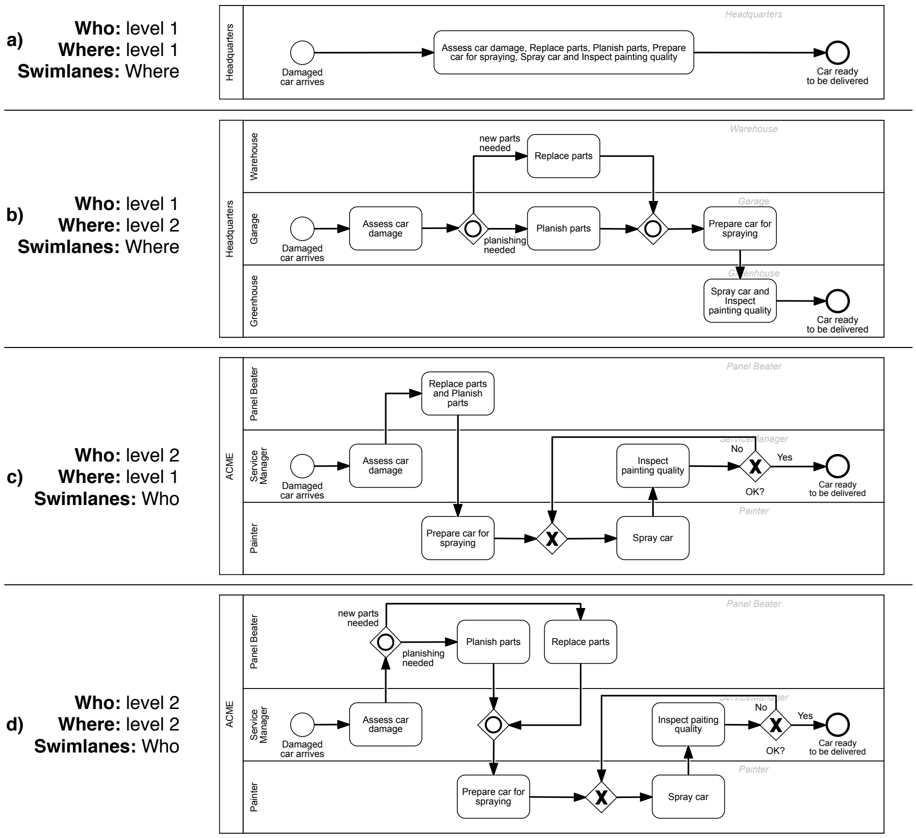

Fig. 8. The BPMN process views of the car repairing process, as generated by the process modeller from the repository content portrayed in figure 4.

8 Conclusions and Future Work

This work served the purpose of exposing the difficulties of having consistent process views, with each conveying the concerns of different stakeholders. Our contribution to this problem is grounded on applying a rule for business process activities aggregation and matching of workflow patterns with the ultimate goal of creating different process views based on existing consolidated models and organisational taxonomies.

Apart from the research work in progress (phases presented in section 4), as future work we intend to eliminate some of the limitations imposed on the consolidated business process model. We are assuming various simplifications on the consolidated models, although they are still compliant with the *BPMN 2.0* standard [12]. Namely, some elements, like data objects, data stores, message flows and boundary events, are totally disregarded.

Moreover, the consolidated models must comply with some modelling constraints: they shall have only one start event; activities and events have exactly one outgoing and one incoming arc (except for the start and end event which do not have, respectively, any incoming and outgoing arc); split gateways have only one incoming arc and arbitrary outgoing arcs while join gateways are the opposite. Despite these limitations, this work already provides a first glance on the generation of stakeholder-specific business process models in *BPMN 2.0*.

Finally, we also aim to improve the view generation algorithm by identifying further patterns and enhance the generation of the aggregated activities' names. Besides that, we aim to generate views in notations other than *BPMN*.

Acknowledgements

This research was supported by the Link Consultings project IT-Atlas n 11419, under the IAPMEI 2020 PO CI Operational Program.

References

1. van der Aalst, W., ter Hofstede, A., Kiepuszewski, B., Barros, A.: Workflow patterns. Distributed and Parallel Databases **14**(1), 5–51 (2003). DOI 10.1023/A:1022883727209
2. Acher, M., Collet, P., Lahire, P., France, R.: Composing feature models. In: M. van den Brand, D. Gašević, J. Gray (eds.) Software Language Engineering, pp. 62–81. Springer Berlin Heidelberg, Berlin, Heidelberg (2010). DOI 10.1007/978-3-642-12107-4_6
3. Caetano, A., Pereira, C., Sousa, P.: Generation of business process model views. Procedia Technology **5**, 378 – 387 (2012). DOI 10.1016/j.protcy.2012.09.042. 4th Conference of ENTERprise Information Systems aligning technology, organizations and people (CENTERIS 2012)
4. Caetano, A., Silva, A., Tribolet, J.: Business process decomposition - an approach based on the principle of separation of concerns. Enterprise Modelling and Information Systems Architectures **5**, 44–57 (2010). DOI 10.18417/emisa.5.1.3
5. Caetano, A., Silva, A.R., Tribolet, J.: A role-based enterprise architecture framework. In: Proceedings of the 2009 ACM Symposium on Applied Computing, SAC '09, pp. 253–258. ACM, New York, NY, USA (2009). DOI 10.1145/1529282.1529337
6. Colaço, J., Sousa, P.: View integration of business process models. In: M. Themistocleous, V. Morabito (eds.) Information Systems, pp. 619–632. Springer International Publishing, Cham (2017). DOI 10.1007/978-3-319-65930-5_48
7. Czarnecki, K., Antkiewicz, M.: Mapping features to models: A template approach based on superimposed variants. In: R. Glück, M. Lowry (eds.) Generative Programming and Component Engineering, pp. 422–437. Springer Berlin Heidelberg, Berlin, Heidelberg (2005). DOI 10.1007/11561347_28
8. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Q. **28**(1), 75–105 (2004)

9. Indulska, M., Green, P., Recker, J., Rosemann, M.: Business process modeling: Perceived benefits. In: A.H.F. Laender, S. Castano, U. Dayal, F. Casati, J.P.M. de Oliveira (eds.) Conceptual Modeling - ER 2009, pp. 458–471. Springer Berlin Heidelberg, Berlin, Heidelberg (2009). DOI 10.1007/978-3-642-04840-1_34
10. International Organization Of Standardization: ISO/IEC/IEEE 42010:2011 - Systems and software engineering – Architecture description (2011)
11. Moon, M., Hong, M., Yeom, K.: Two-level variability analysis for business process with reusability and extensibility. In: Proceedings of the 2008 32Nd Annual IEEE International Computer Software and Applications Conference, COMPSAC '08, pp. 263–270. IEEE Computer Society, Washington, DC, USA (2008). DOI 10.1109/COMPSAC.2008.129
12. Object Management Group: Business Process Model and Notation (BPMN) (2011)
13. Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A design science research methodology for information systems research. Journal of Management Information Systems **24**, 45–77 (2007)
14. Pereira, C.: Using an Organizational Taxonomy to Support Business Process Design. Ph.D. thesis, Insituto Superior Técnico (2011)
15. Pereira, C.M., Caetano, A., Sousa, P.: Ontology-driven business process design. In: T. Skersys, R. Butleris, L. Nemuraite, R. Suomi (eds.) Building the e-World Ecosystem, pp. 153–162. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). DOI 10.1007/978-3-642-27260-8_12
16. Pereira, C.M., Sousa, P.: Business Process Modelling Through Equivalence of Activity Properties. In: Proceedings of the Tenth International Conference on Enterprise Information Systems, pp. 137–146 (2008)
17. Reijers, H., Mans, R., van der Toorn, R.: Improved model management with aggregated business process models. Data & Knowledge Engineering **68**, 221–243 (2009). DOI 10.1016/j.datak.2008.09.004
18. Rosa, M.L., Aalst, W.M.P.V.D., Dumas, M., Milani, F.P.: Business process variability modeling: A survey. ACM Comput. Surv. **50**(1), 2:1–2:45 (2017). DOI 10.1145/3041957
19. Sousa, P., Caetano, A., Vasconcelos, A., Pereira, C., Tribolet, J.: Enterprise Architecture Modeling with the Unified Modeling Language. In: P. Rittgen (ed.) Enterprise Modeling and Computing with UML, pp. 67–94. IGI Global (2006)
20. Sousa, P., Leal, R.T., Sampaio, A.: Atlas : the enterprise cartography tool. In: Proceedings of 8th the Enterprise Engineering Working Conference Forum, vol. 2229 (2018)
21. Sousa, P., Pereira, C., Vendeirinho, R., Caetano, A., Tribolet, J.: Applying the zachman framework dimensions to support business process modeling. In: P.F. Cunha, P.G. Maropoulos (eds.) Digital Enterprise Technology, pp. 359–366. Springer US, Boston, MA (2007). DOI 10.1007/978-0-387-49864-5_42
22. Sowa, J.F., Zachman, J.A.: Extending and formalizing the framework for information systems architecture. IBM Systems Journal **31**(3), 590–616 (1992)
23. Zachman, J.A.: A framework for information systems architecture. IBM Systems Journal **26**(3), 276–292 (1987)