

Managing multi-view business processes models in the Atlas tool

Pedro Sousa^{1,2}, Diogo Cardoso¹, and João Colaço¹

¹ University of Lisbon, Lisbon, Portugal
Av. Rovisco Pais, 1049-001 Lisboa
`pedro.manuel.sousa@tecnico.ulisboa.pt`
`diogocardoso@tecnico.ulisboa.pt`
`joao.p.colaco@tecnico.ulisboa.pt`
² Link Consulting, Lisboa, Portugal
Av. Duque Ávila 23, 1000-123 Lisboa

Abstract. Modelling notations as BPMN imply a dominant perspective in the detriment of others, thus motivating the need of different stakeholders to look for different models of the same process. However, in BPMN, once modelled, the process definition becomes fixed into a dominant perspective, making them harder to use by the other stakeholders. In fact, it is common to see different models of the same processes in different units of an organisation (such as quality, audit, risk or human resources), each focusing specific aspects. In this paper we present an overview of our method to consolidate different perspectives of the same business process into a consolidated model and than to generate on-the-fly different BPMN views according to specific stakeholders perspectives. The implementation work is ongoing but the core of the view generation method is already available.

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

1 Introduction

Business Process models are fundamental to document, analyse and sustain organisational change, as documented by multiple works related to business process management [5, 6].

Process blueprints are developed according to specific goals as well as to the modeller's perspective. This means conflicting specifications may exist for the same process. However, and despite a number of notations and techniques assisting the modelling task, there is no agreement on the modelling criteria that can be used by different stakeholders. Organisations are then faced with disparate blueprints for the same process and no formal procedures to sort out their relevance. In fact, these models are probably accurate while representing the actual organisation but from the modeller's view of that particular process. Given that business processes often cross multiple organisational units, they

are often shared among different stakeholders and represented from multiple perspectives, such as quality, auditing, information technology and security. As a result, process blueprints must be able to address the different stakeholders perspectives and interests and their management and sharing may be simplified if they are handled by a process repository.

The problem of multiple process views starts with the definition of an Activity. Activities comprise a number of atomic tasks, and it is up to the modeller to decide how to aggregate them into activities. This means different modellers can compose tasks into activities differently, leading to different representations of the same process. The work presented here is not new. The basic general ideas were presented in [10], that were further developed in [8] and later in [2]. More recently, a method for view integration was presented in [4] and a method for view generation is presented in [3].

Our approach allows consistent modelling of business processes and efficiently supports different business process views by:

- Consistently integrating views from different stakeholders into a common and consolidated process model. This is sustained on a flexible notion of activity, specifying the properties of an activity that enables further aggregation and/or disaggregation according to the desired level of detail.
- Consistently generating views from a common and consolidated business process model according to the stakeholders requirements, that trigger activities aggregation or disaggregation.

In figure 1 we present the three elements of our approach. On the left the Process View Integration method, the Repository in the middle and the Process View generation on the right.

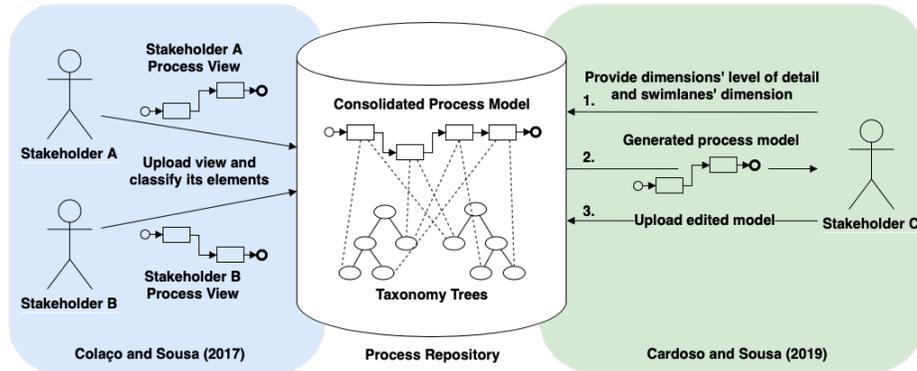


Fig. 1. View Integration (on the left) and View Generation (on the right)

The Process View Integration Method This method merges distinct views of a business process into a single, consolidated business process model [4, 8]. 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 a set of dimensions. As default dimensions we consider the *Who*, *Where*, *How*, *When*, *What*, *Why*, but 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.

The Repository The repository holds:

- The organisational taxonomy;
- 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 2, 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 leaves 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 2. Since the *BPMN* standard allows a wide scope for the swimlane elements (i.e. pools and lanes) [7], 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.

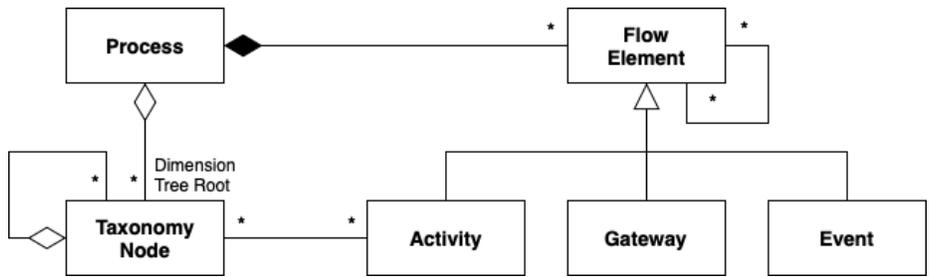


Fig. 2. UML class diagram of the process repository meta-model.

The Process View Generation Method This method produces views of a business process from a consolidated business process model [3], according to the desired hierarchy level of each dimension.

One needs to aggregate the activities in the consolidated model according to a simple rule. The basic rules for aggregating and/or disaggregating activities are listed 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.
- Likewise, an activity α can be disaggregated into two consecutive activities β and δ if and only if for any dimension the following condition applies:
 - there exists more than one taxonomy concept mapped to α at the chosen level of detail.

To better illustrate the problem and our solution, we present a scenario in section 1.1 and afterwards we present the overall solution.

1.1 An 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 to the ACME’s garage, its service manager assesses the vehicle damage. Based on the service manager’s analysis, a panel beater planes 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 low, 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.”

The process is performed by three actors, in three distinct locations and has six activities. Figure 3 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

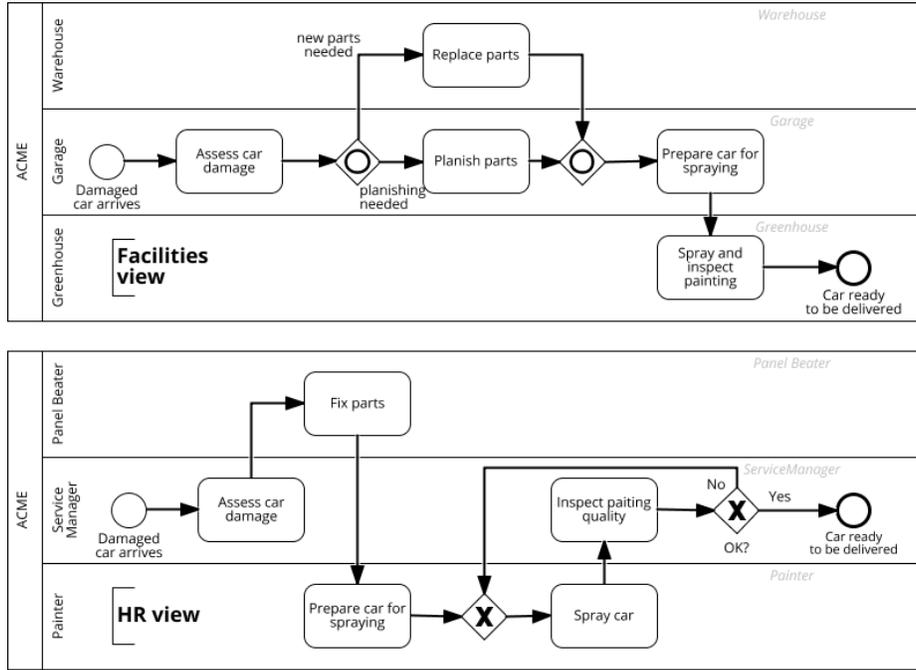


Fig. 3. BPMN process views of the car repairing process, designed by the Facilities department (top) and HR department (bottom).

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.

2 The Approach

2.1 View Integration

The method for view integration is presented in detail in [4], and is briefly presented here.

The proposed view integration method is represented in figure 4, where the Stakeholder A and Stakeholder B lanes can represent any stakeholder willing to integrate its specific process view. The lane Repository represents the business process repository supporting our method. The view integration process starts with the identification of a modelling need, followed by the activity of modelling a view of a given process. The respective view is then uploaded into the repository. At this point, the classification of the view elements is performed by the stakeholder, by giving the information about the desired taxonomy. These

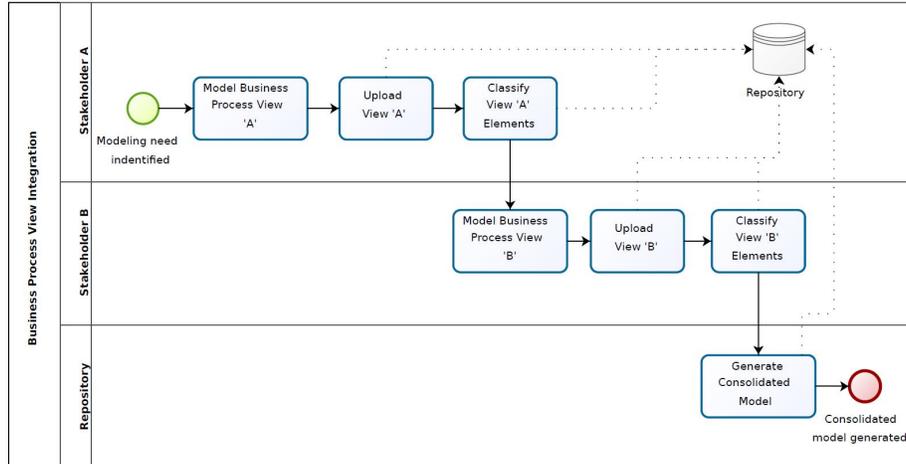


Fig. 4. Method for Business Process View Integration ([4])

activities are repeated in the context of another view of the same process, highlighting the incremental aspect of the proposed method. In this case, we assume there are only two views to be integrated, but there could be more. The process ends with the generation of the consolidated model in the repository, based on the information introduced as input by the stakeholders.

We now present the case for the two process views shown in Figure 3.

Integrating the Facilities View For this example of view integration, consider the following :

- The Who and Where dimensions will be used as the taxonomy to classify the process activities;
- The gateway elements will not be considered for classification, therefore they are connected automatically to the previous and the following elements in the consolidated business process model.

The repository is initially empty. The steps for classifying the elements of the "Facilities view" are the following:

- Pool: ACME. One asks the question *"From the following list, how do you classify this pool?"*, and the logic answer is *"Organisation"* and a new taxonomy node is created in both dimensions Who and Where.
- Lanes: *"Warehouse"*, *"Garage"* & *"Greenhouse"*. The same question is now answered regarding the process lanes. The answer is the same for all the lanes, however, there are two valid answers: *"Location"* and *"Organisational Unit"*. We consider *"Location"* as the answer, resulting in the assignment of all the lanes to the Where dimension. As a result, three child nodes having the *"ACME"* as a parent are created in the Where taxonomy tree.

- Events: "*Damaged car arrives*" and "*Car ready to be delivered*". In this case, the events are automatically assigned as nodes of the When taxonomy tree root.
- Activities: In this step it is asked to the stakeholder to classify each activity considering the Who and Where dimensions, with the resulting classification:
 - Assess car damage - Who: Service Manager; Where: Garage;
 - Planish parts - Who: Panel Beater; Where: Garage;
 - Replace parts - Who: Panel Beater; Where: Warehouse;
 - Prepare car for spraying - Who: Painter; Where: Garage;

Regarding the "*Spray and inspect painting*" activity, the classification is the following: the Where dimension has the value "*Greenhouse*" and the Who dimension is composed of two actors "*Painter*" and "*Service Manager*". When the latter case occurs, the activity must be drilled down so the resulting classification of the tasks only has one value for each dimension. Therefore, using the modeller tool, the stakeholder replaces the "*Spray and inspect painting*" activity with two distinct tasks, connected in this order: "*Spray car*" and "*Inspect painting quality*". These tasks must then be classified as follows:

- Spray car - Who: Painter; Where: Greenhouse;
- Inspect painting quality - Who: Service Manager; Where: Greenhouse;

Integrating the Human Resources View At this point, the repository already has information regarding the "*Facilities view*". We will now describe the classification of the elements belonging to the "*HR view*". The steps for classifying these elements in the taxonomy are the following:

- Pool: "*ACME*". Now that the repository is not empty, based on the name of the element, it is suggested to the stakeholder that the lane "*ACME*" already exists. Since the stakeholder agrees with the classification of this lane as "*Organisation*", he selects the already existing nodes for this lane in the Who and Where dimensions.
- Lanes: "*Service Manager*", "*Panel Beater*" & "*Painter*". This step is analogous to the one described for the classification of the "*Facilities view*". The difference lies in the classification given to the lanes, which in this case is "*Actor*". The result of this step is adding the three lanes as child nodes of the "*ACME*" in the Who taxonomy tree.
- Events: "*Damaged car arrives*" and "*Car ready to be delivered*". Now that the same events are already present in the repository taxonomy, the stakeholder selects them and the taxonomy remains unchanged.
- Activities: In this step it is asked to the stakeholder to classify each activity considering the Who and Where dimensions, with the resulting classification:
 - Assess car damage - Who: Service Manager; Where: Garage;
 - Prepare car for spraying - Who: Painter; Where: Garage;
 - Spray car - Who: Painter; Where: Greenhouse;
 - Inspect painting quality - Who: Service Manager; Where: Greenhouse;

Regarding the *"Fix parts"* activity, the classification is the following: the Who dimension has the value *"Panel Beater"* and the Where dimension is composed of two locations *"Warehouse"* and *"Garage"*. When the latter case occurs, the activity must be drilled down so the resulting classification of the tasks only has one value for each dimension. Therefore, using the modeller tool, the stakeholder replaces the *"Fix parts"* activity with two distinct tasks, connected directly or indirectly using inclusive gateways (the way of connecting these two tasks is not relevant for the activity classification in the taxonomy): *"Planish parts"* and *"Replace parts"*. These tasks must then be classified as follows:

- Planish parts - Who: Panel Beater; Where: Garage;
- Replace parts - Who: Panel Beater; Where: Warehouse;

After uploading and classifying the elements of the *"HR view"*, we reach the final state of the repository that contains the necessary information for the generation of the views presented in the next section. We state so because all the elements present in both views are classified in the taxonomy and the relationships between the elements are sufficient to produce a consolidated model and organisational taxonomy.

2.2 View Generation

The 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 they see fit.

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 view 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 views is cleared whenever one of the corresponding activities in the process model are changed or removed.

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).

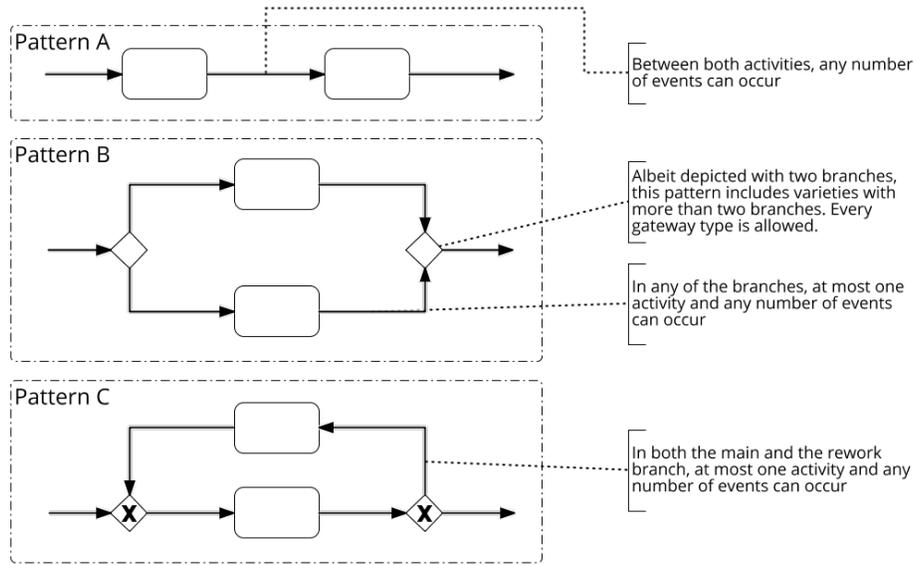


Fig. 5. Patterns that the generation algorithm tries to match during its execution.

3 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.

The tool was developed extending the *Atlas* [9] tool from Link Consulting³ 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*⁴ library and implements the generation algorithm. The modeller integrates with the repository using *REST* services allowing the retrieval and upload of process data.

The construction of the consolidate model, as well as future changes, is done through the process modeller. Thus, the modeller has three distinct functions:

- Creation of new consolidated models (see figure 6);
- Edition of existing models;
- Generation of views from the existing consolidated process models.

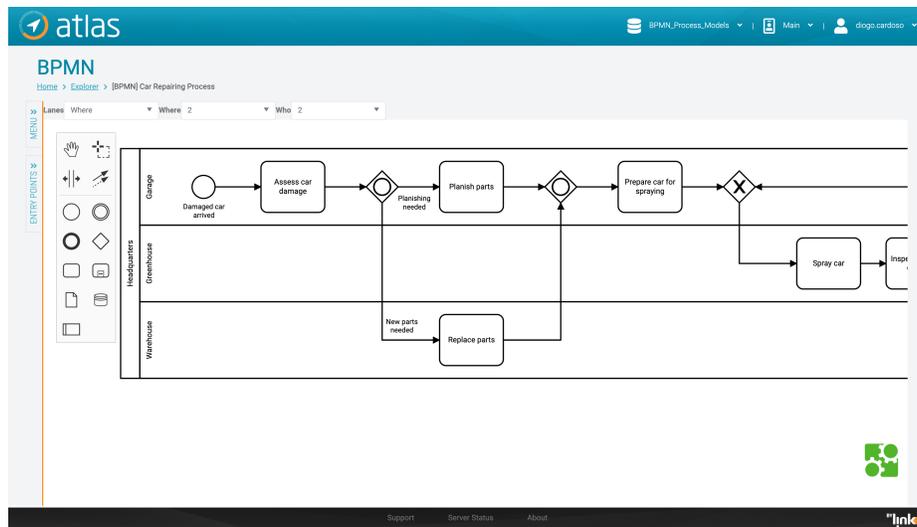


Fig. 6. Screen of the tool after the generation of a view.

Regarding the generation of views, we show some examples of views in figure 7 that are generated from the repository content created after using the view integration method with the views depicted in figure 3. Only two dimensions are present: the *Who* and *Where* dimensions, each with a two-level taxonomy tree, representing process actors and locations, respectively.

³ <http://www.linkconsulting.com/atlas>

⁴ <https://bpmn.io>

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, lanes 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 lanes were chosen to represent the *Where* dimension.

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 lanes 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.

4 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 view integration method 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 suits such concerns, and comment on the usefulness of the generated views, thus fulfilling the requirement of evaluation of our work.

5 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, 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

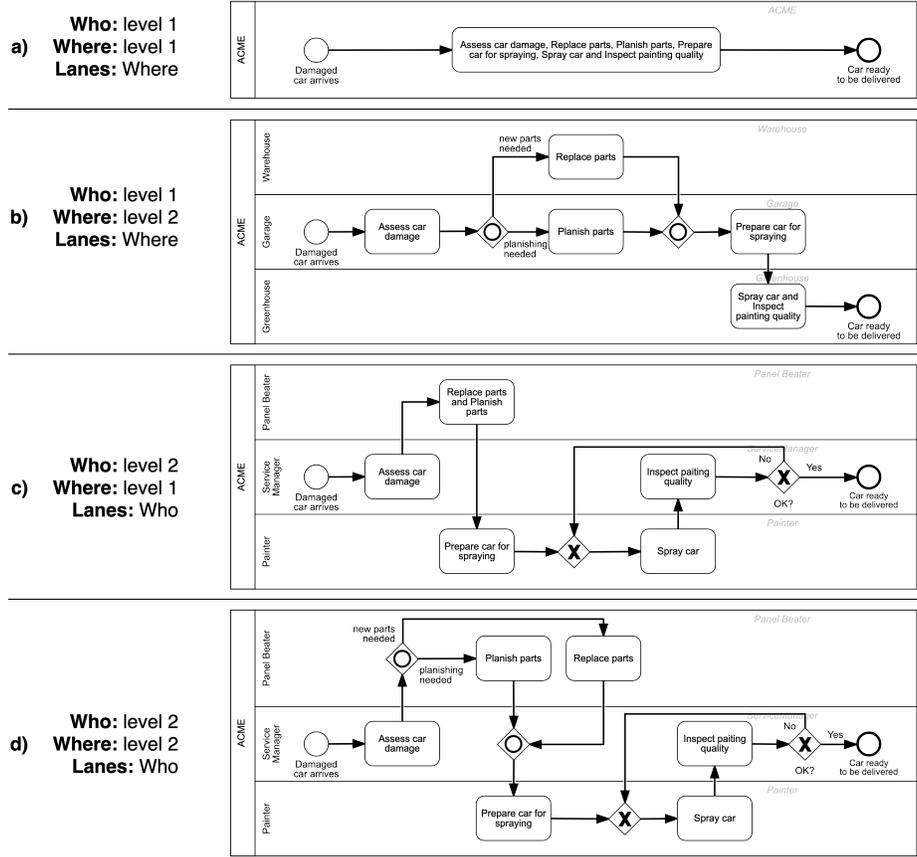


Fig. 7. The BPMN process views of the car repairing process.

they are still compliant with the *BPMN 2.0* standard [7]. 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. 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)
3. Cardoso, D., Sousa, P.: Generation of stakeholder-specific BPMN models. In: *Proceedings of the 2019 Enterprise Engineering Working Conference - EEWC19* (2019)
4. 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
5. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: *Fundamentals of Business Process Management*. Springer-Verlag (2013)
6. 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
7. Object Management Group: *Business Process Model and Notation (BPMN)* (2011)
8. Pereira, C.: *Using an Organizational Taxonomy to Support Business Process Design*. Ph.D. thesis, Instituto Superior Técnico (2011)
9. 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)
10. 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