International Journal of Information Systems and Project Management

Volume 5 | Number 3 Article 5

2017

BPFlexTemplate: A Business Process template generation tool based on similarity and flexibility

Latifa Ilahi
University of Sousse

Sonia Ayachi-Ghannouchi
University of Manouba

Ricardo Martinho
Polytechnic Institute of Leiria

International Journal of
Information Systems and Project Management
ISSN (print):2182-7796, ISSN (online):2182-7788, ISSN (cd-rom):2182-780X
Available online at www.sciencesphere.org/ijispm

BPFlexTemplate: A Business Process template generation tool based on similarity and flexibility

Latifa Ilahi
ISITCom Hammam Sousse, University of Sousse/RIADI Laboratory ENSI, University of Manouba
Tunisia
www.shortbio.org/latifa.ilahi@ipleiria.pt

Sonia Ayachi Ghannouchi
ISG Sousse, University of Sousse/RIADI Laboratory ENSI, University of Manouba
Abdelaziz El Bahi Street, Sousse 4000
Tunisia
www.shortbio.org/sonia.ayachi@isgs.rnu.tn

Ricardo Martinho
Center for Health Technology and Services Research (CINTESIS) / Polytechnic Institute of Leiria, Portugal
School of Technology and Management, Polytechnic Institute of Leiria, Morro do Lena – Alto do Vieiro,
Apartado 4163, 2411-901 Leiria
Portugal
www.shortbio.org/ricardo.martinho@ipleiria.pt

Abstract:
In large organizations with multiple organizational units, process variants emerge due to many aspects, including local management policies, resources or socio-technical limitations. Organizations then struggle to improve a business process which has no longer a single process model to redesign, implement and adjust. In this paper, we propose an approach to tackle these two challenges: decrease the proliferation of process variants in these organizations, and foresee, at the same time, the need of having flexible business processes that allow for a certain degree of adjustment. To validate our approach, we first conducted case studies where we collected six real-world business process variants from two organizational units of the same healthcare organization. We then proposed an algorithm to derive a template process model from all the variants, which includes common and flexible process elements. We implemented our approach in a software tool called BPFlexTemplate, and tested it with the elicited variants.

Keywords:
BPM; Process variants; Template model; Generalisation; Healthcare processes; Similarity metrics.

DOI: 10.12821/ijispm050304

Manuscript received: 5 June 2017
Manuscript accepted: 14 September 2017

1. Introduction

Business processes can be generally described as a set of logically related tasks, behaviors, resources and data that organizations develop over time to produce specific business results [1]. Process models are then designed and evolved to conduct business processes execution, and to have a more or less uniform record of all its executions (instances or process cases). These executions should, in turn, reflect what really happens/is happening in organizations. This way, management is enhanced since metrics and process key performance indicators can be derived from similar recorded cases of a certain process model. Processes can then be improved by redesigning their models to tackle less optimized parts, considering the analysis performed to their recorded cases.

Nevertheless, large organizations often have several organizational units that are managed distinctly, but are also supposed to follow the same process models for similar, cross-organizational business processes and process improvement. Examples of these processes include student enrolment within distinct faculties of the same university, or a medical appointment scheduling across several public hospitals of the same National Health System. Although organizations usually impose uniformity in modeling and executing these processes to offer the same service and to measure performance, many organizational units make several adjustments over time, resulting in a proliferation of process variants. The problem then becomes twofold: 1) organizations loose general business process management ability because measuring, comparing, monitoring and optimizing processes that have many variants is difficult, and 2) organizational units have been improving their process variants overtime, and may not want to adopt/obey to a certain "general" and completely "rigid" process model in all its length.

Therefore, organizations with several organizational units can contain a family of business process model variants. By comparing these variants, we can observe that some of their process elements (tasks, resources, data, decision nodes, and sub-processes) are common, while others may differ, be added or deleted, or simply be in a different order.

Considering the well-known Business Process Management (BPM) lifecycle [2], to redesign, model, implement and maintain each process variant of a process family would be too costly and cumbersome (and therefore difficult to carry or use) for organizations. On the other side, a process variant of a certain organizational unit can enclose valuable good practices and optimization efforts, either to be applied in a local context, or even suitable for the remaining units of the organization.

The challenges that arise from this setup include, on one side, the need to align business processes across all organizational units for better process management, and on the other side, the urge to take advantage of the most succeeded process optimizations made across individual units over time. Accordingly, on the gap between these commonalities (fragments that are shared by all process variants) and to cope with this need for process flexibility, a proper management of these family variants along the entire BPM life cycle has become crucial.

In this paper, we propose an approach to tackle these challenges. The main output of our approach is a template process model that we derive from a certain number of process model variants. This derived template model can then be used seamlessly across all organizational units of a certain organization, reducing variants proliferation. At the same time, it foresees flexibility (the ability to make adjustments) taking into account the adjustments registered in all existing variants. This way, the organization can perform business process management more effectively by assuring the overall use of a single (template) process model which reflects common good practices across all organizational units. Additionally, these organizational units can make use of the template's flexibility features, to reflect their particular adjustments.

Our approach includes the use of good process modeling practices, and an overall algorithm to deal with variants comparison/similarities and derive the flexibility-enabled template process model. We also implemented a software tool to help process engineers to derive this template model. We validate our approach by performing case studies that include the elicitation of six real-world business process variants from the healthcare domain.

This paper is organized as follows: The next section introduces the main background notions including Business Process (BP), BPM, BP variability, BP comparisons and similarity and BP flexibility. Section 3 highlights the related works. This is followed by section 4 where we explain our proposed approach called "BPFlexTemplate". In section 5, we present our case studies that allowed us to validate our approach. Then, in section 6, we describe the BPFlexTemplate software tool. Finally, section 7 renders some conclusions and future work.

2. Background

Besides referring to general concepts of business processes, our proposed approach is based on the combination of three main research topics of Business Process Management (BPM): 1) process variability and the techniques used to deal with it; 2) process model comparison and similarity approaches; 3) process flexibility concepts and techniques. The next subsections provide background on overall BPM as well as in these three topics.

2.1 Business processes and Business Process Management (BPM)

Business processes (BP) can be considered as the arteries of modern organizations. They determine how work is done in the organization. A business process is a collection of related events, activities and decisions that involve a number of actors and resources, and that collectively lead to an outcome that is of value to an organization or its customers [3]. So, a BP may be considered as the set of performed activities in coordination, within an organizational and technical environment. In other words, it defines what shall be done (activities), by whom (organizational and technical environment) and how it shall be done (coordination).

A business process model is the main artifact for representing the respective process. It is described in a graphical way using elements (including activities, gateways, events, sub processes, resources and data) of a specific language or notation like the Business Process Model and Notation (BPMN) [4] or Event-driven Process Chain (EPC) [5]. Business process support is present in several business domains. Examples include the healthcare domain [6], [7], [8], [9], [10], automotive engineering [11], and public administration [12]. In all these domains, business process models are designed and evolved in order to have a more or less uniform record of all instances (cases), in order to reflect what really happen in organizations. This way, management is enhanced since metrics and process key performance indicators can be derived from similar recorded cases of a certain process model.

Business Process Management (BPM) is considered as the science of overseeing how work is performed in an organization. This performance aims to ensure consistent outcomes and to take advantage of improvement opportunities [3]. In this context, the main term "improvement" may take different meanings depending on the objectives of the organization. The powerful interests of the BPM-based approach give rise to many expectations and prospects, including, for instance, the areas of Business Process Reengineering (BPR), or the supervision of Business Activity Monitoring (BAM). Many research works assume that process design/modeling is the core of each of these approaches [13].

Before evolving and improving business processes within organizations, a clear and shared vision of the organization's "as is" business processes is required. The process modeling step is considered as a way to achieve this. Widespread in many domains, process models are used to understand, develop and communicate. A process modeling based approach can serve several objectives, including the identification of tasks to be potentially supported, or the optimization of the already in place Information Systems. Also, it contributes to document the process in order to help writing requirements, making decisions, eliminate confusion and facilitate exchange between involved actors. Therefore, a process design/modeling approach improves collaboration between functional structures within an organization and between organizations. In addition, it provides the reduction or elimination of the dependence of an activity in relation to employees who are assigned to it, and improves the management and knowledge capitalization. Also, it allows checking and improving processes and sub-processes in order to continuously make the processes of the company evolving [14] and [15].

Many works are dealing with the representation of the Business Process Management (BPM) cycle. For instance, in Figure 1 we can observe the proposal of van der Aalst in [2]. It starts by the (re)design phase. In this step, a process model is (re)designed, meaning either the creation of a new process model, or the adjustment of an existing model. Then, in the implementation/configuration phase, the model is configured into a running system usually known as a Business Process Management System (BPMS). In the run and adjust phase, the processes are enacted and adjusted when needed. Also in this phase, the process is not redesigned and no new software is created; only predefined controls are used to adapt or reconfigure the process [2].

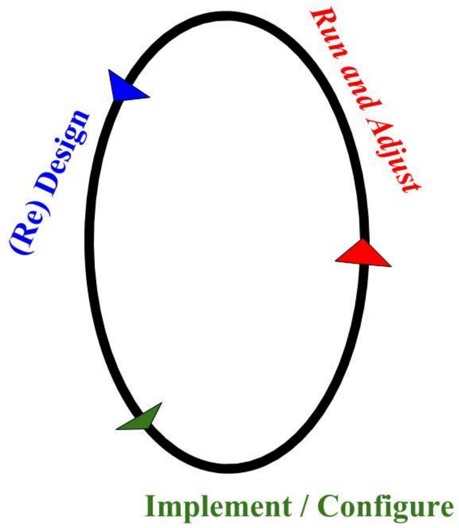

Fig.1. BPM lifecycle (adapted from [2])

2.2 Process variability

For a particular business process, we can have different variants reflecting adjustments made out of a reference process model, taking into account specific requirements of the process context [16]. In fact, the design of processes spans several layers of abstraction, from theoretical lifecycle models, via organizational models of best practice, to the plans of actual process instances [17]. These design efforts usually result in a template model, which can be defined as a generalization of other models, which represent a large number of process model variants [18]. This template model should also be adjustable to get a particular or specific model variant that best suits a certain context. To represent all these specificities, Jorgensen proposed the lifecycle of process model evolution that is illustrated in Figure 2. He divided process models into two categories. The first one is "Particular models", which aim to support performance of a particular (adjusted) process model. The second one is "General models", which abstract common properties from a number of actual processes, represent normative standards for the organization, or are templates for reuse and adaptation into particular models.

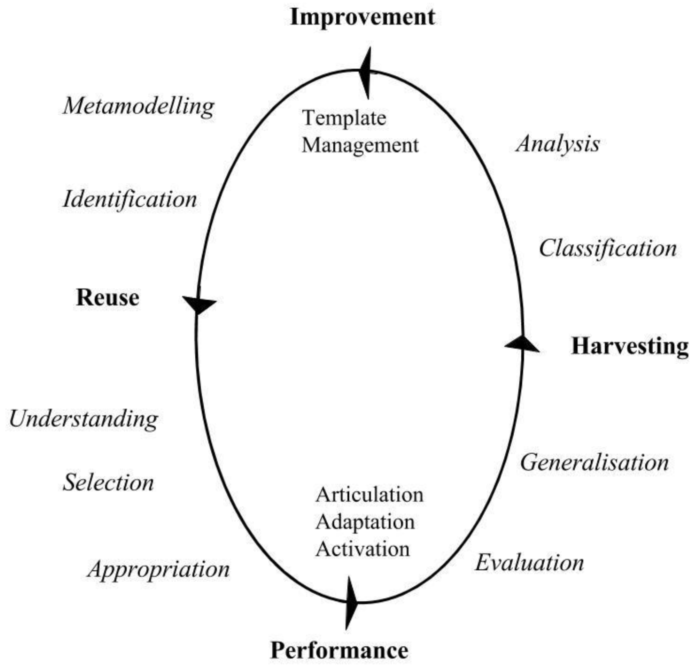

Fig.2. Lifecycle of process model evolution (adapted from [17])

The proposed lifecycle of process model evolution presents four main phases: improvement, reuse, performance and harvesting. Applying a general process model to a particular situation is a case of reuse. The reuse may also refer to copy and paste of a previously developed particular model into a new process. During the performance phase, users need to add new details, remove irrelevant parts etc. of the model.

The process of generalizing one or more particular models is called harvesting. The goal of harvesting is to provide templates that can be reused in the future, and to utilize practical experience as input to assessment and improvement of the general models [17]. Therefore, for a business process, different variants may exist. These constitute, in turn, an adjustment of a general, master process (e.g., a reference process) to meet specific needs of a certain process context [19].

In fact, organizations may need to have a collection of related process variants, which are denoted as a business process family. In practice, a process family may comprise more than one process variant [11] [20]. Take for instance the automotive industry, where we can find a process family dealing with vehicle repair and maintenance, which comprises more than 900 process variants [21]. They have some fragments that are shared by all process variants, but also show vehicle-specific variations. Also, in healthcare, we can find reports on more than 90 process variants for handling medical examinations in a hospital [20] and [22]. In another example, we may refer to the check-in procedures at an airport, which are characterized by a high degree of variability as well. The variability can be caused, for instance, by the type of check-in (if it is online, at the self-service machine or at the counter) [20].

Since a process model can represent several process perspectives such as functional, behavioral, organizational, informational, temporal and operational perspectives, variants can come up from changing any of these perspectives, either individually or in a combined form. The main objective of existing approaches that are dealing with variability is the contribution to avoid model redundancies, foster model reusability, and reduce modeling efforts [23].

2.3 Process comparison and similarity

Over time, many organizations have built many repositories of BP models that serve as a knowledge base for their ongoing BP management efforts [24]. These repositories may contain hundreds or even thousands of models. However, to better manage and improve these large repositories requires effective search or comparison techniques. To search if there is an existing "similar" model or to compare two models is useful to prevent duplication and to have a continuous and effective improvement of the process [25]. A process engineer needs to identify common or similar business processes between, for instance, recently merged companies, in order to analyze their overlap and to identify areas for consolidation. Dijkman et al. in [24] use the term process model similarity query to refer to such search queries over process model repositories.

To answer a similarity based comparison query between BP models involves determining the degree of similarity between one model and another one (or more) [26]. In this context, similarity can be defined from several perspectives, including the following: 1) Node matching similarity which is based on a comparison of the labels that appear in the process models (task labels, event labels, etc.), using either syntactic or semantic similarity metrics (presented in next paragraph), or a combination of both. These metrics start by calculating an optimal matching between the nodes in the process models by comparing their labels. Based on this matching, a similarity score is calculated taking into account the overall size of the models. 2) Structural similarity which is based on the topology of the process models seen as graphs. It is based on the observation that nodes in process models with their relations constitute a mathematical graph. Based on that observation it uses existing techniques for graph comparison based on graph edit distance [27], which is commonly used in information retrieval. 3) Behavioral similarity which is based on the execution semantics of process models. It takes into account the causal relations between tasks in a process model. These causal relations are represented in the form of a causal footprint [28].

Existing approaches for process model elements comparison which is based on similarity metrics can be divided into those based on: 1) Syntactic similarity, where we consider the syntax of labels; 2) Semantic similarity, where we look at the semantics of the words within the labels; 3) Attribute similarity, where we look at the attribute values; 4) Type similarity, where we look at the node types; and 5) Contextual similarity, where we do not only consider the similarity of two nodes, but also the context in which these nodes occur [24].

2.4 Process flexibility

BPM solutions include methods, techniques, and tools to support the design, enactment, management, and analysis of operational business processes of organizations. Accordingly, continuously changing conditions are forcing organizations to rapidly adapt their processes. Then, flexibility requirement has also been following the evolution of BPM, since it reflects the ability to adapt business processes to predicted and unpredicted changing scenarios. Regarding the literature, there are several business process flexibility taxonomies including those proposed by Schonenberg et al. [29] and Regev et al. [30]. We will briefly describe the taxonomy of Regev et al. [30], since it is the one we use in our approach.

This taxonomy focuses on changes that may occur during the lifecycle of a business process. These changes can be of one of three dimensions of change [30]:

- The abstraction level of change which corresponds to the level of application of change in a business process. The change may concern the specification or the process instance;
- The purpose of the change regarding different aspects of the process which are subject to change. The change may concern the process activities (functional), the control flow (behavioral), process the data (informational) or the various protocols used in the process (operational);
- The properties of the change as the degree of change. These concern the extent of change which can be incremental (change a part of the process) or revolutionary in order to create a new process; the duration of

change that can be temporary or permanent; the swiftness of change that can be either immediate or deferred; and the anticipation of change that can be scheduled or ad hoc.

3. Related work

Business process flexibility has been an important research trend for the last two decades, due to the rigidity of process modeling languages and Business Process Management Systems available in the early 1990s. It is often viewed in terms of the ability of an organization's processes and supporting technologies to adapt to changes [42]. Regev et al. [30] observed that a process can only be considered flexible if it is possible to change it without needing to replace it completely, and proposed a business process flexibility taxonomy in [30]. Related works on this matter are aimed at providing process engineers with language extensions and mechanisms (see for instance [33]) to incorporate flexibility in process models and/or instances (by implementing BPMSs that support runtime flexibility).

On the other hand, we can find in literature many examples of algorithms and approaches towards the comparison and consolidation of process models. Notably, the authors in [24,43] propose similarity algorithms to compare process models and elements through their labels, structureand behavior. In [44] the authors present a merging algorithm that takes as input a collection of process models and generates a configurable process model [45]. A similar objective is announced by the authors in [46], where process versions are merged and integrated into a business process model, considering the differences and the change operations history of an initial model. Li et al. [45] take also into consideration the changes/deviations recorded in a Process-Aware Information System (PAIS) regarding a certain business process, and propose to users to decide on the change operations to be applied regarding a newly discovered process variant [47].

Regarding the use of software tools to help process engineers perform these approaches and algorithms, the panorama we found in literature is ratherlimited. In [48], the authors aim for a visual approach to detect significant differences between process variants, based on what was recorded in event logs. They use colored transition systems to model behavior and to highlight the differences, and implement their approach as a Process Comparator plugin of the well known ProM framework [49]. For another aim, Armas et al. [50] proposed the BP-Diff web-based tool that takes pairs of process models in BPMN [4] format and outputs behavioral difference diagnosis in the form of textual statements and graphically overlaid on the process models [50].

In [40], Ivanov et al. propose a Model-View-Controller (MVC) web-based BPMNDiffViz tool, which uses structural matching to compare process models represented in BPMN. Specifically, BPMNDiffViz allows for a colored and quantified visualization of graph differences, stores them, and provides statistics. It calculates the minimum graph edit distance between two process models (number of operations that should be performed to transform one model to another) using an A* algorithm, and also accounts for the string edit distance for each pair of the corresponding graph nodes.

4. The BPFlexTemplate approach

Adopting a BPM based approach may contribute to optimize organizational aspects and the products/services that are delivered when executing the organization's business processes. This way, management is enhanced since metrics and process key performance indicators can be derived from similar recorded cases of a certain process model. Hence, large organizations often have several organizational units that are managed distinctly, but are also supposed to use process models for similar, cross-organizational business processes. As far as we could perceive from literature and from the case studies we conducted, it is common for this kind of organizations to have several variants from a determined business process, due to factors such as local management policies, resource limitations, socio-technical limitations or even culture.

This is the case not only for public governance organizations such as the ones related with education, health or justice, but also for large national or global companies which struggle to streamline their processes along their several organizational units.

For our BPFlexTemplate approach, we adapt the BPM process model lifecycle illustrated before in Figure 2, proposing techniques used in the Evaluation, Generalization, Classification, Analysis and Improvement sub-phases, to obtain a template process model suited to this kind of organizations (cf. Figure 3).

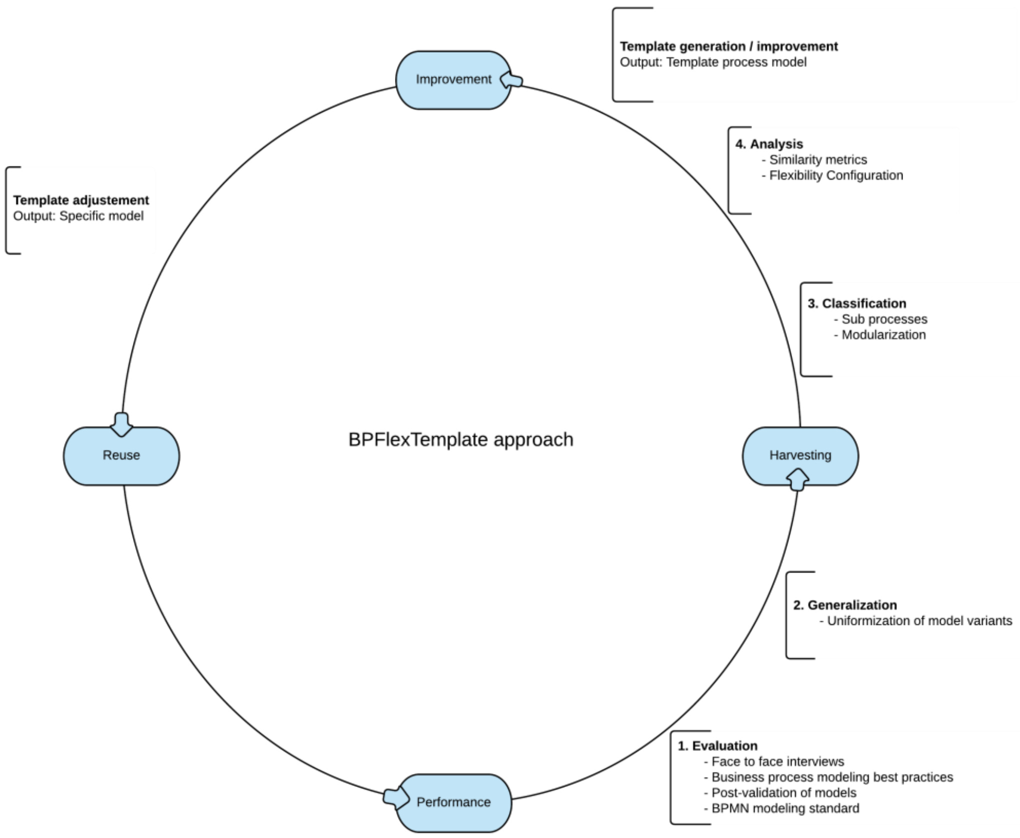

Fig.3. Adaptation of the BPM process model cycle to include our BPFlexTemplate approach

The proposed approach focuses on the right part of the cycle, denoting the techniques used across the Harvesting phase. Harvesting refers to using the models and experience from one or more processes to generate a new template, a variant or a revision of an existing template [17]. It involves manual adaptation of local models to make them suitable for different contexts. Typically, Knowledge Management or methodology experts perform such knowledge capture. The template derived may include both specific language constructs and an initial model.

In the Evaluation sub-phase, we perform process model eliciting by using face to face interviews and best practices regarding business process modeling, based on the works of Mendling et al. in [31] and Pinggera et al. in [32]. Mendling et al. [31] propose a set of seven process modeling guidelines (7PMG). It is a guide to users towards improving their process models' quality. Concerning Pinggera et al. [32] work, they propose the process of "process modeling", which consists of a cycle of three successive phases: 1) comprehension, 2) modeling and 3) reconciliation. We use these best practices to elicit the different model variants in place within distinct units of a certain organization, taking advantage of the BPMN standard and graphical notation to express these model variants. We then perform a post-validation of these model variants with the interviewees.

Then, in the Generalization sub-phase, uniformisation of process elements is performed by identifying the inputs, processing and outputs of the tasks involved, and labeling them the same in all process model variants, in case they match regarding their overall behavior.

Further ahead, the Classification sub-phase aims to provide a better way to modularize the elicited model variants, by dividing them into more manageable and reusable sub-processes. This will provide a set of sub process templates, which can also be later combined differently according to the specific needs of a certain organizational unit.

Finally, the Analysis sub-phase is performed, where these uniformised and classified process model variants can be compared, in order to derive a single template model. We perform these comparisons using similarity metrics. These comparisons will allow us to derive a single template process model, which will foresee process elements that are either present to most process variants, or specific only to some. We classify these process elements as common and flexible, respectively. This Analysis sub-phase includes also the configuration of the type of flexibility allowed for a certain flexible process model element (such as an activity, a gateway, a document data object or a resource allocation – see, for instance, the work of Domingos et al. in [33]). This flexibility type can be one of those mentioned in the referred taxonomy proposed by Regev et. al. in [30].

Therefore, the overall algorithm for this Analysis sub-phase may be summarized on the following Listing 1 (written based on the pseudo code guidelines presented in [34]):

```
FOR each process model variant
   CALL  performSimilarityAnalysis  WITH  currentTemplateModel,  modelVariant,
   similarityType RETURNING similarityResults
   CALL classifyTemplateElements WITH similarityResults, classificationCriteria
   RETURNING classifiedTemplateModel
END FOR

FOR each flexible process element in classifiedTemplateModel
   CALL configureFlexibility WITH processElement, flexibilityType
END FOR
```

Listing 1 – Algorithm for the Analysis sub-phased of the BPFlexTemplate approach

The functions performSimilarityAnalysis, classifyTemplateElements and configureFlexibility may be implemented as components that follow the Template method design pattern from Gamma et. al [35]. This means that they can be invoked with different parameters and then implemented within distinct components (classes) which share the same interface. As an example, and looking forward into our real-world case study, we can apply the following general parameters:

- Function *performSimilarityAnalysis*: invoke with the value of GED (Graph Edit Distance) for the *similarityType* parameter;

- Function *classifyTemplateElements*: invoke with the value of 75% for the *classificationCriteria* parameter, to denote the threshold value for the number of model variants in which a certain element must at least be present to be classified as a common process element;
- Function *configureFlexibility*: invoke with the value of ad-hoc sub process for the *flexibilityType* parameter.

Therefore, the main focus of this algorithm is to reduce model variability for the overall process model template to be used across all organizational units, and allowing for flexibility requirements at the same time that can reflect needed adjustments for a particular organizational unit. In this context, process engineers may control and align 3 parameters in this algorithm: 1) type of similarity metrics to be used to compare the model variants (structural, behavioral, …); 2) level of similarity for each process element (for instance, 75% means that only elements with a similarity of 0.75 or greater will be considered as common, and therefore candidates to be included in the set of common process elements in the template); 3) flexibility mechanism to be applied to process elements that are not similar, and therefore will belong to the flexible set of process elements in the template (for example, an ad-hoc solution means that flexible process elements will be kept and included within an ad-hoc sub process, where they can or cannot be executed, and with no particular order).

5. Case studies

To validate our BPFlexTemplate approach, we elicited six home healthcare process variants from two primary care health centers (organizational units) of the public National Health System (NHS) in Portugal (the main organization). The objectives of our case studies were to validate, on one hand, the assumption that home healthcare processes are not executed the same way within primary care centers of the same organization (the NHS) (and that therefore variants really exist). On the other hand, these case studies allowed us to also validate our global BPFlexTemplate approach presented in Figure 3, including the techniques, best practices and algorithms accordingly.

For the first three sub-phases of Evaluation, Generalization and Classification, we could also benefit from our previous experiences in modeling telemedicine processes [36], [37] and [38] and home healthcare processes in Tunisia [8], [9] and [39]. Regarding these case studies, the home healthcare processes that should be used across all primary care centers is (textually) described in national legislation and overall guidelines provided by the Ministry of Health.

For the Evaluation sub-phase of our BPFlexTemplate approach, we used the 7PMG techniques in the interviews performed with two chief nurses from the primary care health centers of Leiria and Lisbon. Then, we performed an uniformisation of all the process models elicited, and classified them into six process model variants, which were then validated by the same chief nurses. Like in our recent works [8], [9] and [39], we classified these model variants of home healthcare process into three sub-processes, namely: 1) patient admission; 2) organizational care; and 3) patient care.

Figures 4 and 5 show the two validated model variants of the organizational care sub-process (Leiria and Lisbon), after applying to them the techniques of evaluation, generalization and classification described in our BPFlexTemplate approach.

As presented BPMN process models depicted in Figures 4 and 5, these two variants from the organizational care process are noticeably different (even at eye sight). The existence of this variability could also be confirmed within the other four model variants for the patient admission and patient care sub-processes, as can be consulted in [9].

We will use these two depicted variants to illustrate how they are treated throughout the analysis sub-phase of our BPFlexTemplate approach. Regarding this sub-phase, we then performed a similarity comparison for these two model variants, according to our algorithm provided in Listing 1.

This comparison was based on the following similarity metrics: syntactic similarity (Syn sim), type similarity (Type sim) and semantic similarity (Sem sim) between each similar/near element. The values presented in Table 1 are obtained with manual calculations, according to the proposed methods of Dijkman et al. in [24].

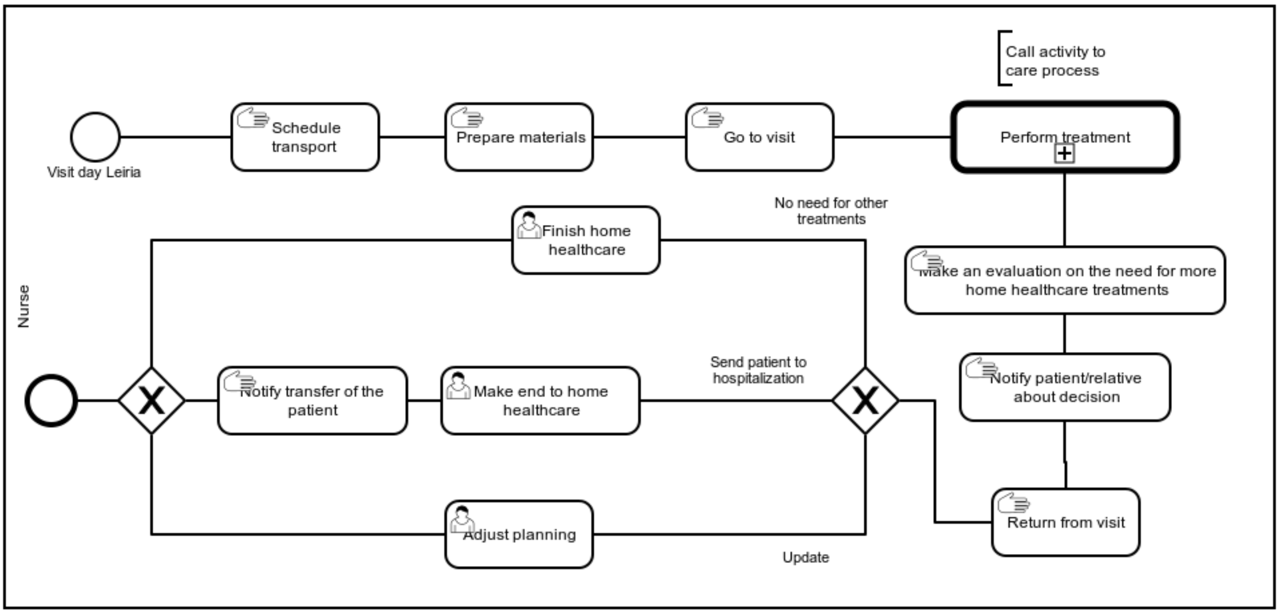

Fig.4. the organizational care sub process model of Leiria primary care health centre according to the BPMN Notation

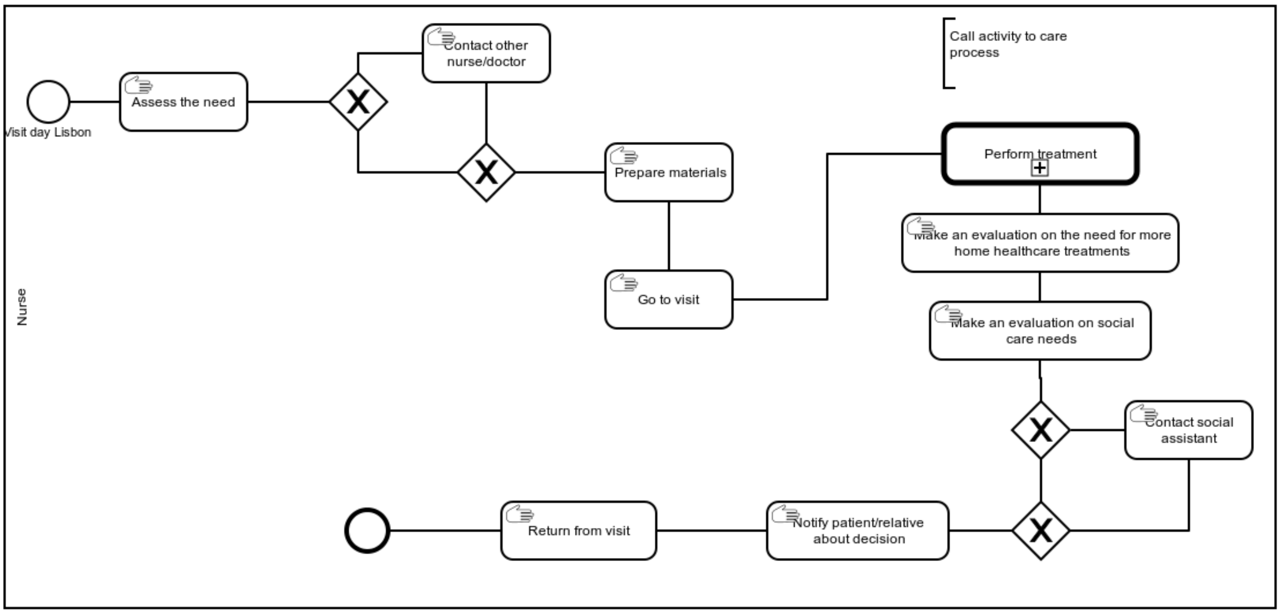

Fig.5. the organizational care sub process model of Lisbon primary care health centre according to BPMN Notation

Table 1 - Similarity results for the 2 model variants of the organizational care home healthcare process

Model variant1	Model variant2	Syn sim	Type sim	Sem sim	N.M sim	G.E.D. Sim
Organizational care of Leiria primary care centre	Organizational care of Lisbon primary care centre	0.53	0.58	0.58	0.80	0.32

Then we calculated the Node Matching Similarity (N.M sim) which is based on pair wise comparisons of node labels or attributes. This is obtained by calculating an optimal equivalence mapping between the nodes of the two model variants being compared. The node matching similarity between two business process graphs representing two process models is:

$$simnm(B1,B2) = \frac{2 \cdot \sum_{(n,m)\in M^{opt}_{Sim}} Sim(n,m)}{|\{n| \in N_1, \tau_1(n) \notin ts\} + |\{n|n \in N_2, \tau_2(n) \notin ts\}|}$$

The node matching similarity score is the sum of the label similarity scores of the matched pairs of nodes. The Graphic Edit Distance (GED) similarity is computed as one minus the average of the fraction of inserted or deleted nodes, the fraction of inserted of deleted edges and the average distance of substituted nodes.

All similarity values presented in Table 1 (all scaled from 0 – non similar, to 1 – similar) denote that there is a significant difference between the two analyzed model variants, proving a certain amount of variability for this process.

As presented in our proposed algorithm in Listing 1, and in order to derive our (flexible) template model from these variants, we can configure three parameters: 1) Type of similarity, 2) Classification criteria and 3) Flexibility mechanism. So, in this case, we chose the following values for these parameters: 1) *Structural Similarity*, 2) 100% and 3) the *ad-hoc sub-process* mechanism for these parameters. These configurations led us to derive the template model presented in Figure 6.

In this derived template model, we marked the common (rigid) process elements with a red contour and the flexible process elements in green. This means that, for this case, the red contoured elements represent the common part of the template, i.e., the one that is really being enforced in all organizational units, according to the configuration parameters for similarity and classification criteria. As for the green contoured elements, they represent the flexible part of the template, to which the ad-hoc sub process mechanism was applied. For instance, for activity Adjust planning, this means that it can be executed in any particular order, or even skipped according to an organizational unit's specific context and needs.

In the next section, we describe and illustrate how we have built a software tool that implements the algorithm presented in Listing 1, in order to enhance the work of a process engineer that wishes to perform this kind of analyses.

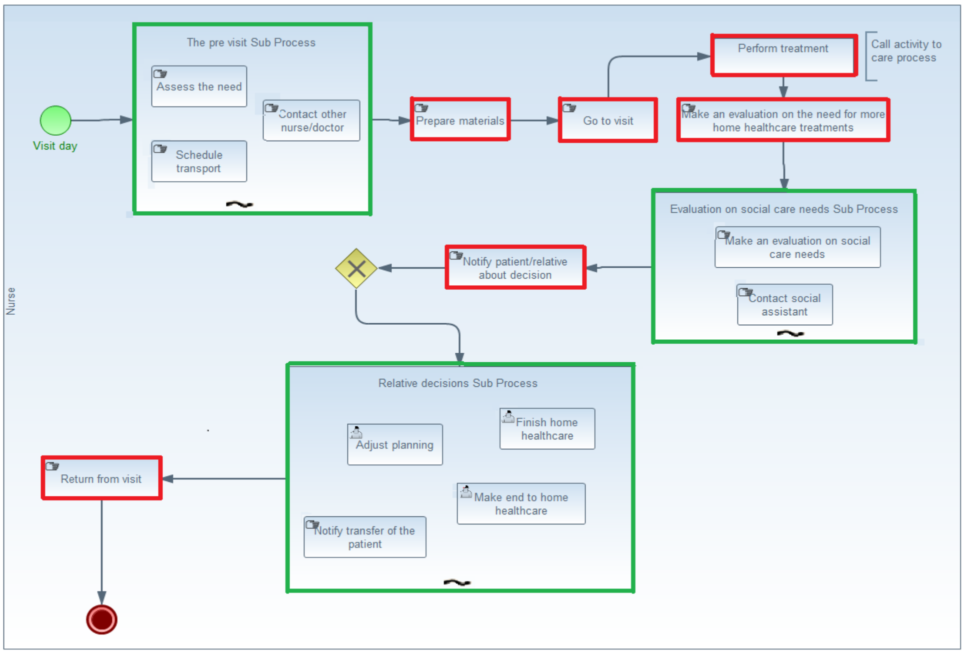

Fig.6. Derived template process model from organizational care of two organizational units [9]

6. BPFlexTemplate software tool

In a previous stage of our work (as in [9]) we could validate our approach by manually executing the Analysis sub-phase of our BPFlexTemplate approach. This implied manual similarity analyses calculations and process model variants comparisons, which revealed to be a cumbersome procedure. In order to automate this procedure, and make it more suitable to be used by organizations' process engineers, we implemented the BPFlexTemplate software tool, which is an extension of the BPMNDiffViz tool from [40]. In the latter, the authors propose a similarity structural matching Web application that compares process models represented in the BPMN 2.0 format. It allows visualizing graph differences, storing them, and providing statistics, assisting in analyzing process model discrepancies. The implemented comparison algorithm finds the minimal graph edit distance between two process models (number of transformations, which should be performed to transform one model to another) using an A* algorithm, and calculates the string edit distance for each pair of the corresponding graph nodes (BPMN process elements of a model). In terms of technology, the BPMNdiffViz tool is a Java-based web application that uses the Spring MVC (Model-View-Controller) framework [41].

6.1 Use cases for the BPFlexTemplate tool

To achieve our BPFlexTemplate software tool, we added to BPMNdiffViz features that concern our purpose of comparing process models and then deriving a template model that is based on our presented algorithm in Listing 1. These template models are also stored in the database and the process engineer can visualize them on the browser. Therefore, in our extended BPFlexTemplate tool, a process engineer can perform these three main features: 1) Select BPMN models from loaded new BPMN models or generated template models; 2) compare two or more BPMN models, produce final graph edit distance, exploit these comparison results and; 3) derive a template model. Figure 7 presents the use case diagram that illustrates these features.

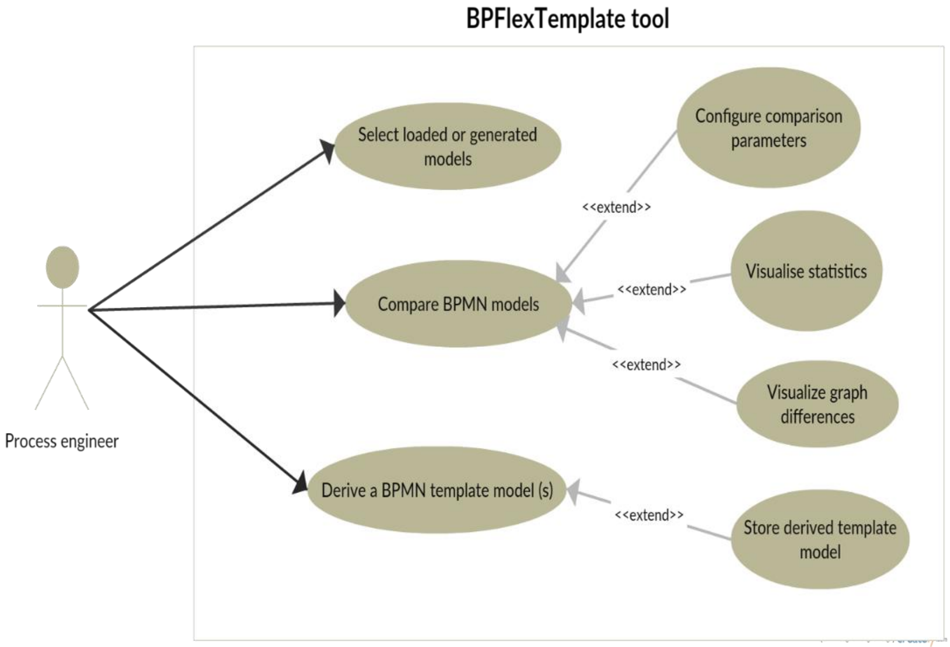

Fig.7. Use case diagram of BPFlexTemplate tool

Thus, based on these main features, it will allow to visualize all derived template models, compare this template with another model variant and generate another template. A process engineer may also choose the type of similarity that should be applied to derive differences and similarities between model variants, as well as the classification criteria (0 to 100%, since similarity scales usually range between 0 – non-similar, and 1 – completely similar). Finally, s/he can also pick the flexibility mechanism to be applied to the identified flexible (non-common) process elements in the template. For instance, s/he may choose between simply deleting them from the template, or including them in an ad-hoc sub process, as illustrated in our example of Figure 6.

To validate our tool, we picked up our elicited model variants from the case studies as input. Our aim is to illustrate that the tool can derive identical results to those presented in section IV where we performed manual similarity calculations and template design.

Figures 8, 9 and 10 show the initial steps where the user can upload or choose the previously loaded process models (variants) to be compared. Derived template models are also stored within the database to which the BPFlexTemplate connects. This means that, for instance, for a process with three variants, the user can first derive a template model by comparing two of them, and then perform a second comparison between this resulting template and the third variant.

As shown in Figures 8 and 9, we loaded the same two model variants analyzed in section IV, elicited within the two primary care health centers of Leiria and Lisbon.

The third step is the configuration of comparison parameters. We may have many combinations: For the similarity type, we can choose between 3 main types (Structural, Node Matching or behavioral) and for each type we can also choose the similarity type between models elements. Then, we may adjust the classification criteria between 0% to 100% (From non similar to exactly the same). Finally, we need to parameter the flexibility mechanism. This means how we want to deal with the flexibility challenge (Replace flexible parts with Ad-hoc sub processes, or delete them, etc.).

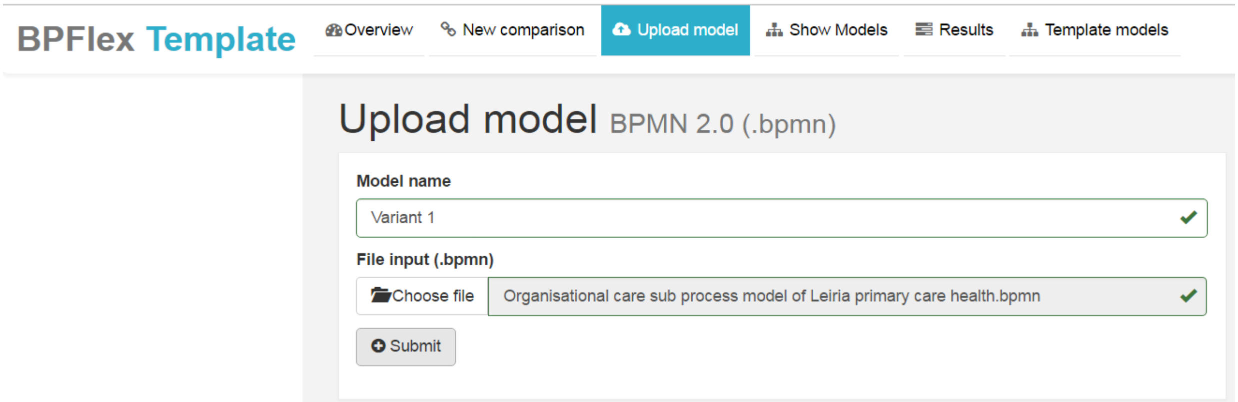

Fig.8.Upload model in the database

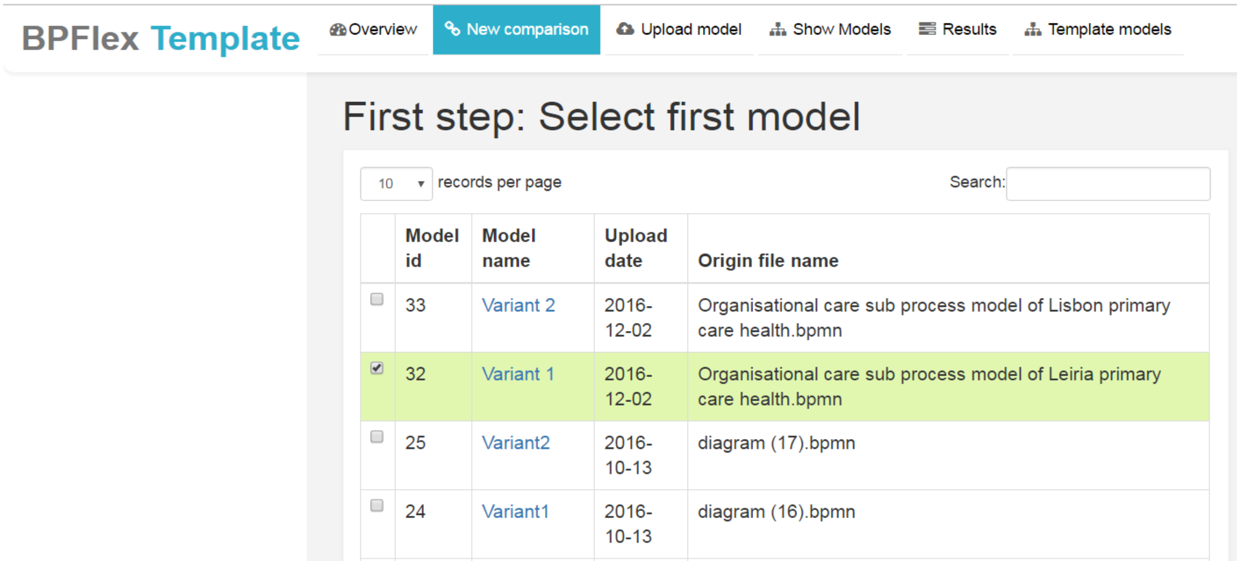

Fig.9. First step: Select the first model to be compared

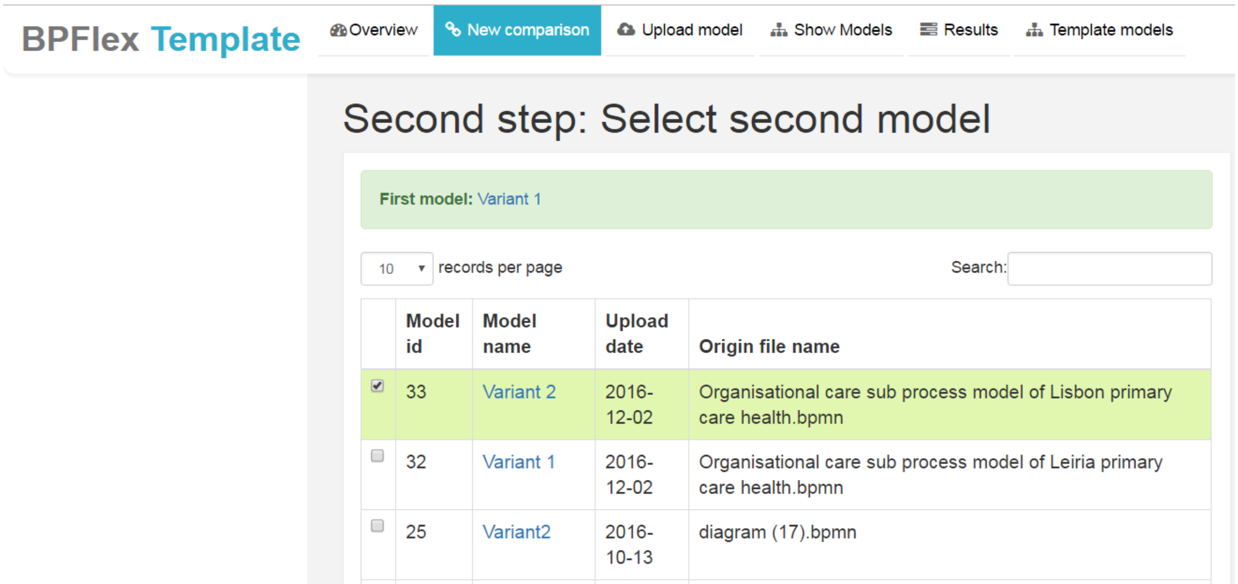

Fig.10. Second step: Select the second model to be compared

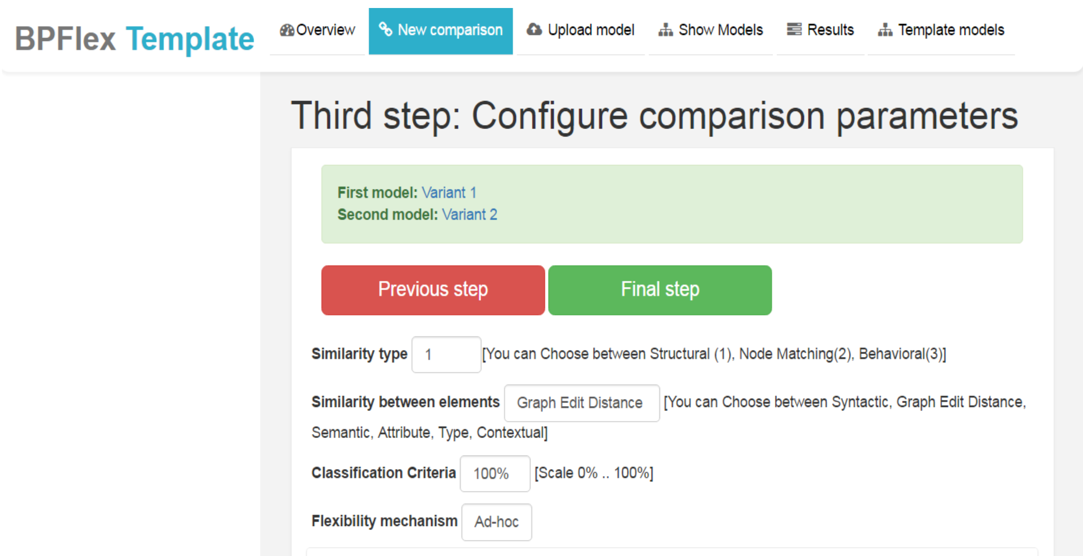

Fig.11. Configuration of the parameters to derive the template model for variants Variant1 and Variant2

Figure 12 presents the derived template model for the parameters configuration presented in Figure 11. We can observe that, although with a different layout, our BPFlexTemplate tool automatically derived an identical template model to that in Figure 6, for the same considered model variants.

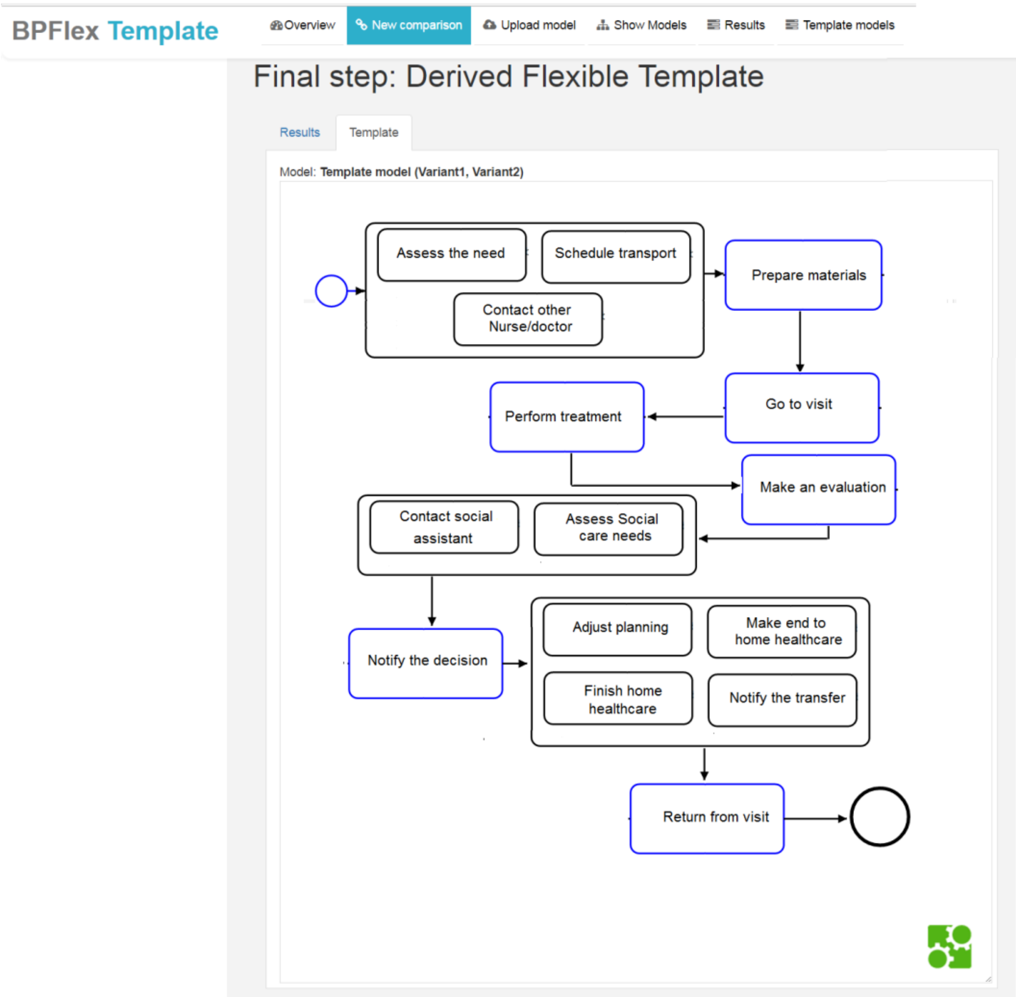

Fig.12. Organizational care derived template model in the BPFlexTemplate tool, with the parameters: Structural, 100%, Ad-hoc sub process

The process elements with a blue contour represent, in this case, the common part of the template, as the black ones illustrate the flexible part, where the ad-hoc sub process flexibility mechanism was applied.

For the same two variants, we also derived a second template model, taking into account different values for the classification criteria (50%) and the flexibility mechanism (Delete), as illustrated in Figure 13.

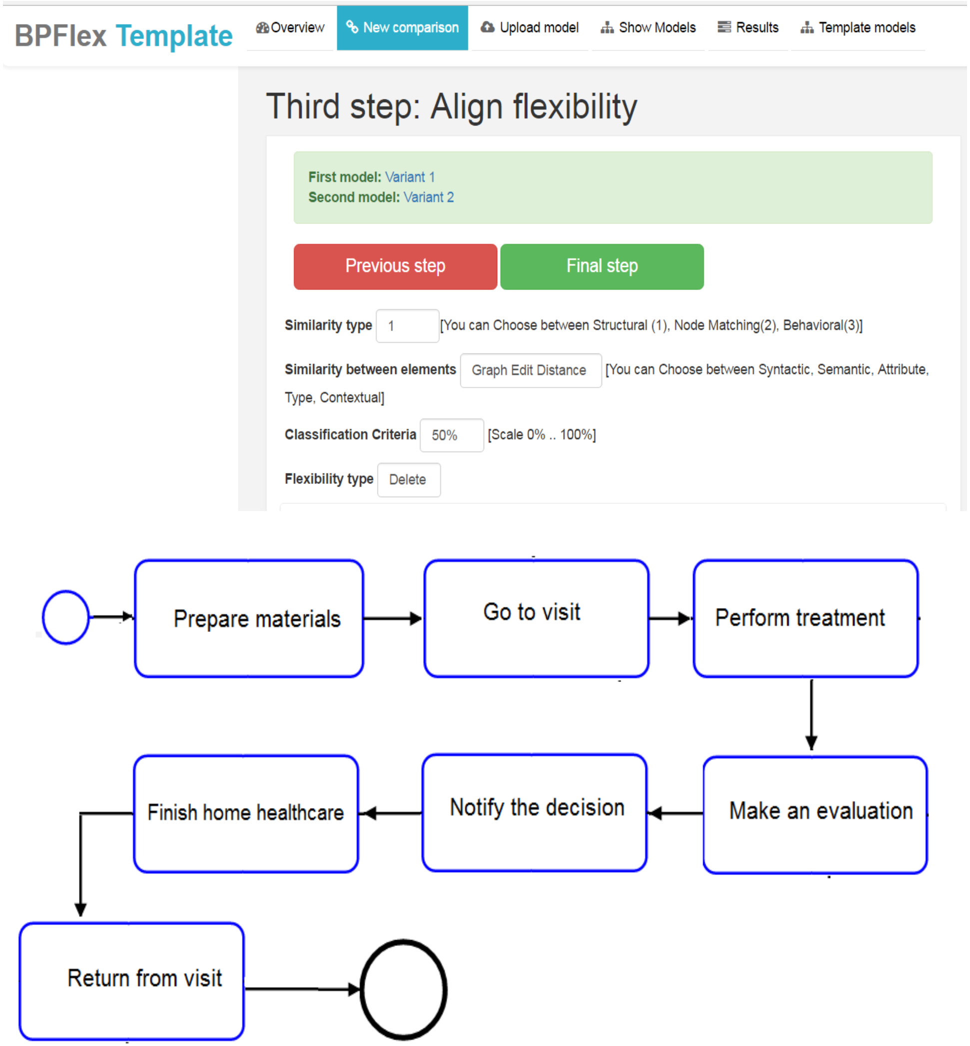

Fig.13. Organizational care template model in the BPFlexTemplate tool, with the parameters: Structural, 75%, Delete

Here, we can observe that by aligning the classification criteria to 50%, some process elements shifted to the *common* part of the template in Figure 13, in comparison to the former template of Figure 12, namely the activities "*Make end to home healthcare*" and "*Finish home healthcare*". Also, and since the *Delete* flexibility mechanism was chosen, this template only kept these *common* process elements, *deleting* the remaining elements from the derived template. This would be suitable, for instance, when a certain organization wishes to enforce a certain process template within its organizational units, which only considers activities that are common, discarding any particular adjustments made to the process in those organizational units.

7. Conclusion and future work

In this paper, we could demonstrate that process variants emerge and can evolve to be quite different in primary care health centers (organizational units) of the Portuguese National Health Service (top regulated organization). For this, we elicited process models for home healthcare in these primary care centers. Assessing the similarity results from the 3 pairs of model variants elicited, we could verify that they present significant differences. This means that the overall management and optimization of these processes may be difficult to achieve, since recorded cases and data differ and can hardly be handled as a coherent set.

Therefore, we proposed the BPFlexTemplate approach to align model variants with significant differences. The overall aim is to reduce the proliferation of process variants that can occur in organizations that are top regulated, but contain several organizational units that often adjust the process models they should follow.

The main output of this approach is a template process model that, based on modeling best practices and similarity results, is composed of two types of process elements: *common* and *flexible*. Common elements should be executed strictly, while *flexible* ones may or may not be executed, according to the organization’s flexibility mechanisms allowed. Nevertheless, the purpose is that this template model can fit all organizational units, and enforce them to follow common procedures, as well as allowing them for some flexibility without having to create model variants. This way, recorded cases and data from these processes will all fall into a unique (template) model, enhancing BPM and governance in general for this kind of organizations.

We also present in this paper a software tool to perform similarity studies and derive the template models, according to our proposed algorithm in Listing 1. Process engineers can use this tool to upload, store and compare model variants, and then derive a template model, according to a set of parameters that can be chosen. The tool can automate the calculations regarding similarity comparisons between model variants, as well as model template generation. These tasks can be rather time consuming, cumbersome and error prone when done manually.

In a near future, we are planning to improve our software tool to include more options regarding types of similarity and flexibility mechanisms that can be configured for model template generation. This will allow the template model to foresee a wider range of (different) variants, and to include other flexible process elements that better cover the overall adjustments made to a certain process.

Future work is also pointing us to better explore the term “Business Process Management (BPM) Governance”. This will imply, as a first step, to implement a process model repository where organizational units can access, improve and share their model variants. Taking primary care centers and home healthcare processes as our case studies, medical directors or chief nurses can, with this repository in place, access to the latest template models, and actually contribute to their improvement. Directors in the National Health Ministry can, in turn, have a notion of all variants of a certain process, and use the BPFlexTemplate approach to derive template models accordingly.

References

[1] K. C. Laudon and J. P. Laudon, Management Information Systems: Managing the digital firm, 15th ed. New Jersey, USA: Pearson, 2017.

[2] W. M. van der Aalst, “Business process management: a comprehensive survey,” ISRN Software Engineering, vol. 12, February, 2013.

[3] M. Dumas, M. La Rosa, J. Mendling and H. A. Reijers, Fundamentals of Business Process Management, Heidelberg, Germany: Springer, 2013.

[4] Object Management Group (2011, 01, 03). Business process model and notation (BPMN 2.0) [Online]. Available: http://www.omg.org/spec/BPMN/2.0.

[5] W. M. van der Aalst, “Formalization and verification of event-driven process chains,” Information and Software Technology, vol. 41, no. 10, pp. 639-650, July, 1999.

[6] R. Lenz and M. Reichert, “IT Support for Healthcare Processes – Premises, challenges, perspectives,” Data & Knowledge Engineering, vol. 61, no. 1, pp. 39-58, April, 2007.

[7] A. Russo and M. Mecella, “On the Evolution of Process-oriented Approaches for Healthcare Workflows,” International Journal of Business Process Integration and Management, vol. 6, no. 3, pp. 224-246, January, 2013.

[8] L. Ilahi, S. A. Ghannouchi and R. Martinho, “A Real-World Case Scenario in Business Process Modelling for Home Healthcare Processes,” in HEALTHINF, Rome, Italy, 2016, pp. 166-174.

[9] L. Ilahi, S. A. Ghannouchi and R. Martinho, “A Business Process Management Approach to Home Healthcare Processes: On the Gap between Intention and Reality,“ in Encyclopedia of E-Health and Telemedicine, M.M. Cruz-Cunha, I. M. Miranda, R. Martinho and R. Rijo, Eds., Hershey, USA: IGI Global, 2016, ch. 35, pp. 439-457.

[10] D. Domingos, A. Respício and R. Martinho, “Reliability of IoT-Aware BPMN Healthcare Processes,” in Internet of Things and Advanced Application in Healthcare, C. I. Reis, M. S. Maximiano, Eds., Hershey, USA: IGI Global, 2016, ch. 8, pp. 214-248.

[11] D. Müller, J. Herbst, M. Hammori and M. Reichert, “IT Support for Release Management Processes in the Automotive Industry,” Business Process Management, vol. 4102, no. 2006, pp. 368-377, October, 2006.

[12] J. Becker, L. Lis, D. Pfeiffer, M. Räckers, “A Process Modeling Language for the Public Sector – The PICTURE approach,” Wybrane Problemy Elektronicznej Gospodarki, pp. 271-281, 2007.

13] Lonjon. A. (2004). Business process modeling and standardization [online]. Available: http://www.bptrends.com.

[14] M. du Plessis, "The role of knowledge management in innovation," Journal of Knowledge Management, vol. 11, no. 4, pp. 20-29, 2007.

[15] V. Grover and M. Malhotra, “Business process reengineering: A tutorial on the concept, evolution, method, technology and application,” Journal of Operations Management, vol. 15, no. 3, pp. 193-213, 1997.

[16] A. Hallerbach, T. Bauer, M. Reichert, “Context-based configuration of process variants,” in 3rd International Workshop on Technologies for Context-Aware Business Process Management, Barcelona, Spain, 2008, pp. 31-40.

[17] H. D. Jørgensen, “Interactive process models,” PhD. dissertation, Fakultet for Informasjonsteknologi, Matematikk og Elektroteknikk, Norwegian University of Science and Technology, Trondheim, Norway, 2004.

[18]R. Dijkman, M. Rosa and H. Reijers, "Managing large collections of business process models—Current techniques and challenges," Computers in Industry, vol. 63, no. 2, pp. 91-97, 2012.

[19] A. Hallerbach, T. Bauer, M. Reichert, “Configuration and management of process variants,” in Handbook on Business Process Management, J. von Brocke, Eds., Heidelberg, Germany: Springer, 2010, pp. 237-255.

[20] C. Ayora, V. Torres, B. Weber, M. Reichert and V. Pelechano, "VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems," Information and Software Technology, vol. 57, pp. 248-276, 2015.

[21] A. Hallerbach, T. Bauer and M. Reichert, "Capturing variability in business process models: the Provop approach," Journal of Software Maintenance and Evolution: Research and Practice, vol. 22, no. 6-7, pp. 519-546, 2010.

[22] C. Li, M. Reichert and A. Wombacher, "Mining business process variants: Challenges, scenarios, algorithms," Data & Knowledge Engineering, vol. 70, no. 5, pp. 409-434, 2011.

[23] C. Ayora, V. Torres and V. Pelechano “Feature Modeling to deal with Variability in Business Process Perspectives,” Journal of Colloid and Interface Science, vol. 12, pp. 111-124, 2012.

[24] R. Dijkman, M. Dumas, B. van Dongen, R. Käärik and J. Mendling, "Similarity of business process models: Metrics and evaluation," Information Systems, vol. 36, no. 2, pp. 498-516, 2011.

[25] B.F. van Dongen, R.M. Dijkman and J. Mendling, “Measuring similarity between business process models,” in 20th International Conference on Advanced Information Systems Engineering (CAiSE), Montpelier, France, 2008, pp. 450–464.

[26] R. Dijkman, M. Dumas and L. García-Bañuelos, “Graph matching algorithms for business process model similarity search,” in International Conference on Business Process Management, Uml, Germany, 2009, pp. 48-63.

[27] P. Hart, N. Nilsson and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100-107, 1968.

[28] B. Dongen, J. Mendling and W. M. van der Aalst, “Structural patterns for soundness of business process models,” in 10th IEEE EDOC Conference (EDOC), Hong Kong, China, 2006, pp. 116–128.

[29] H. Schonenberg, R. Mans, N. Russell, N. Mulyar and W. M. van der Aalst, “Towards a Taxonomy of Process Flexibility,” in in 20th International Conference on Advanced Information Systems Engineering (CAiSE), Montpelier, France, 2008, pp. 81-84.

[30] G. Regev, P. Soffer, R. Schmidt, “Taxonomy of Flexibility in Business Processes,” in CAISE’06 Workshop on Business Process Modelling, Development, and Support (BPMDS'06), Luxemburg, 2006, pp. 236.

[31] J. Mendling, H. Reijers and W. van der Aalst, "Seven process modeling guidelines (7PMG)," Information and Software Technology, vol. 52, no. 2, pp. 127-136, 2010.

[32] J. Pinggera, S. Zugal, M. Weidlich, D. Fahland, B. Weber, J. Mendling and H. A. Reijers, “Tracing the process of process modeling with modeling phase diagrams,” in International Conference on Business Process Management, Clermont-Ferrand, France, 2011, pp. 370-382.

[33] D. Domingos, R. Martinho, J. Varajão, “Controlled Flexibility in Healthcare Processes: A BPMN-Extension Approach,” in Encyclopedia of E-Health and Telemedicine, M.M. Cruz-Cunha, I. M. Miranda, R. Martinho and R. Rijo, Eds., Hershey, USA: IGI Global, 2016, ch. 40, pp. 521-535.

[34] J. Dalbey (March, 2016). Pseudocode standard [Online]. Available: http://users.csc.calpoly.edu/~jdalbey/SWE/pdl_std.html

[35] E. Gamma, R. Helm, R. Johnson and J. Vlissides, Design patterns. Boston, USA: Addison-Wesley, 2016.

[36] L. Ilahi and S. Ghannouchi, "Improving Telemedicine Processes Via BPM," Procedia Technology, vol. 9, pp. 1209-1216, 2013.

[37] L. Ilahi, S. A. Ghannouchi and R. Martinho, “Healthcare information systems promotion: from an improved management of telemedicine processes to home healthcare processes,” in International Conference on Technological Ecosystems for Enhancing Multiculturality, Salamanca, Spain, 2014, pp. 333-338.

[38] L. Ilahi, R. Martinho, S. Ghannouchi, D. Domingos and R. Rijo, "Similarity Based Approach for Comparing Home Healthcare Processes Models in Portugal," Procedia Computer Science, vol. 100, pp. 1250-1259, 2016.

[39] L. Ilahi, R. Martinho, S. A. Ghannouchi, D. Domingos and R. Rijo, “Towards a Business Process Management Governance approach using process model templates and flexibility,” in IEEE World Congress on Services (SERVICES), San Francisco, USA, 2016, pp. 27-34.

[40] S. Y. Ivanov, A. A. Kalenkova, W. M. van der Aalst, “BPMNDiffViz: A tool for BPMN models comparison,” in International Conference on Business Process Management, Innsbruck, Austria, 2015, pp. 35-39.

[41] Pivotal Software. (2016). Spring MVC [Online]. Available http://docs.spring.io/.

[42] R. Snowdon, B. Warboys, R. Greenwood, C. Holland, P. Kawalek and D. Shaw, "On the architecture and form of flexible process support," Software Process: Improvement and Practice, vol. 12, no. 1, pp. 21-34, 2007.

[43] R. Dijkman, M. Dumas, L. Garcia-Banuelos, R. Kaarik, “Aligning business process models,” in 13th IEEE EDOC Conference (EDOC), Auckland, New Zealand, 2009, pp. 45-53.

[44] M. La Rosa, M. Dumas, R. Uba and R. Dijkman, "Business Process Model Merging," ACM Transactions on Software Engineering and Methodology, vol. 22, no. 2, pp. 1-42, 2013.

[45] M. Rosemann and W. van der Aalst, "A configurable reference modelling language," Information Systems, vol. 32, no. 1, pp. 1-23, 2007.

[46] C. Gerth, “Process Model Merging,” in Business Process Models. Change Management, Heidelberg: Springer, 2013, pp. 165-175.

[47] C. Li, M. Reichert and A. Wombacher, “Discovering Reference Models by Mining Process Variants Using a Heuristic Approach,” in International Conference on Business Process Management, Uml, Germany, 2009, pp. 344-362.

[48] A. Bolt, M. De Leoni and W. M. van der Aalst, “A visual approach to spot statistically-significant differences in event logs based on process metrics,” in International Conference on Advanced Information Systems Engineering, Ljubljana, Slovenia, 2016, pp. 151-166.

[49] B. F. van Dongen, A. K. A. de Medeiros, H. M. W. Verbeek, A. J. Weijters, W. M. van der Aalst, “The ProM framework: A new era in process mining tool support,” Applications and Theory of Petri Nets, pp. 444–454, 2005.

[50] A. Armas-Cervantes, P. Baldan, M. Dumas, L. Garcıa-Banuelo, “BP-diff: A tool for behavioral comparison of business process models,” in International Conference on Business Process Management, Haifa, Israel, 2014, pp. 1-6.

Biographical notes

Latifa Ilahi
Latifa Ilahi is a PhD Student on Computer Sciences at the Higher Institute of Computer Sciences and Communication Technologies, University of Sousse, Tunisia. She is a graduate of the University of Kairouan in Tunisia where she obtained her master's degree in Smart Information Systems. She is a member of the RIADI research laboratory at the University of Manouba in Tunisia. Her PhD was realized in collaboration with the Polytechnic Institute of Leiria, Portugal. Her research interests include Health Informatics, Business Process Management, Business Process variability and flexibility.

www.shortbio.org/latifa.ilahi@ipleiria.pt

Sonia Ayachi Ghannouchi
Sonia Ayachi Ghannouchi is an Associate Professor in Enterprise Computing at the High Institute of Management of Sousse, in the University of Sousse. She holds a PhD in Computer Science from the University of Manouba in Tunisia and an HDR in Enterprise Computing from the University of Sousse in Tunisia. Her taught courses include: "Databases", "Information systems", "Software Engineering" and "Business Process Reengineering". She's an active researcher in RIADI Laboratory at ENSI School in the University of Manouba. Her research interests include: Software engineering and Reengineering, Business Process Modelling, Business Process Management, Process Mining, e-learning and e-health. She's involved in the supervision of several master degree and PhD theses.

www.shortbio.org/sonia.ayachi@isgs.rnu.tn

Ricardo Martinho
Ricardo Martinho is an Associate Professor at the School of Technology and Management – Polytechnic Institute of Leiria, Portugal. He teaches several subjects related to Enterprise Information Systems, Enterprise Application Development, Software Engineering (Agile Methods) and Healthcare Computer Programming and Information Systems. He graduated in Electrical Engineering – Computer Science at University of Coimbra, received his MSc in Computer Science - Information Systems Programming from IST - Technical University of Lisbon, and his PhD from University of Trás-os-Montes and Alto Douro. He is currently a member of the CINTESIS research center and supervises several MSc and PhD theses in the Computer Science – Health Informatics research areas. He has several publications in conference proceedings, book chapters and journals, related to the Business Process Management, Software Engineering and Health Informatics research areas. He serves as editor-in-chief, member of editorial board and reviewer for several books and international journals, and has served in several committees of international conferences.

www.shortbio.org/ricardo.martinho@ipleiria.pt