Electronic Government and Electronic Participation
E. Tambouris et al. (Eds.)

doi:10.3233/978-1-61499-570-8-201

A Data Oriented Approach to Derive Public Administration Business Processes

Riccardo Cognini[a], Flavio Corradini[a], Fabrizio Fornari[a], Andrea Polini[a] and Barbara Re[a]
[a] *Computer Science Division, School of Science and Technologies University of Camerino 62032 -- Camerino (MC), ITALY*
{name.lastname@unicam.it}

Abstract. The delivery of services to citizens by Public Administrations requires to set up and coordinate complex Business Processes. Typically homogeneous Public Administrations, such as municipalities, have to provide the same services to all citizens. Nevertheless their concrete implementation, and the supporting Business Process model and data object models, can slightly differ from one Public Administration to the other due to organizational factors. If such variability is not explicitly represented and managed, each office will have to reflect on and analyse the requirements posed by the delivery of the service; then they will have to derive a specific process and data model. On the other hand the explicit modeling of variability can reduce the work to be done and permits to define general specifications from which specific model variants can be derived according to specific needs. In this paper we propose a novel approach, inspired by Feature Modeling techniques, for data object variability modeling that can be used to provide high level blueprints from which detailed Business Processes and data object specifications can be derived. Finally, a complex scenario has been applied to validate the approach with encouraging results.

Keywords. Business Process Variability, Data Object Variability, Feature Modeling

1. Introduction

Complex organizations have to support many different Business Processes (BPs) to provide services/products to customers and to reach organization objectives. Typically BPs can present similarities when applied to support similar services or to make similar products. The explicit management of such variability, both with respect to documents (following referred as data objects) and to the activities to carry on, can be extremely useful in order to reduce complexity. Therefore methods explicitly supporting the variability modeling and the management of data objects and of BPs are very much needed [1]; the delivery of services to citizens by Public Administrations (PAs) can be re-conducted to such a situation. The case of PA is particularly interesting in reference to the possibility of modeling and managing data and BP variability [2]. At a certain level of abstraction all PAs will share the same abstract data object and BPs. Nevertheless, when details have to be defined in order to concretely support the service delivery, the BP models start to differentiate in order to include organization dependent characteristics [3]. When variability is not explicitly managed, the result is that each PA works independently to design its BPs. This certainly causes a waste of time and money since most of the BPs requirements are the same also for other PAs, therefore its explicit modeling will permit to save time and money to PAs, and at the same time common modeling will reduce interoperability risks.

Business Processes in the PA sector are regulated by the legislation for all PA delivering a service, nevertheless some degree of freedom is left regarding the concrete implementation of such supporting BPs. This means that some behavior is not completely generalizable since each single PA has specific needs, for instance in relation to the location, the size of the territory which it covers and other many possible factors.

According to the given scenario we present a novel approach to model public services from laws, starting with the generation of a shared data view in order to reduce the cost of the BP design step. The data object view allows to have an immediate idea of which are the data involved and provides a way to facilitate the development of Configurable Process Models (CPMs). A CPM consists in a single model that groups many BP variants sharing a common goal [4] then it provides rules defining how each single BP variant can be modeled and adapted to each single PA. In our case, we use a Feature Model (FM) for the data object modeling and the Business Process Feature Model notation to represent CPMs.

The paper is organized as follows. In Section 2 background material is provided, then the proposed approach is presented in the Section 3, and its application to a real case study is presented in Section 4. Finally, Section 5 presents related works and Section 6 closes the paper drawing some conclusions and opportunities for future works.

2. Background

2.1. Feature Modeling

Feature modeling is an approach emerged in the context of Software Product Lines to support the development of a variety of products from a common platform introducing the concept of a family of BPs [5].

A FM is a graphical model which makes use of a tree representation, with the root representing the general product to develop; it permits to express different relationships among the possible features that can be included in a specific variant of the product. In particular, in the first feature modeling approach proposed, named Feature-Oriented Domain Analysis (FODA), *mandatory*, *optional* or *alternative* constraints on features have been introduced [6]. A *Mandatory* feature represents a characteristic that each product variant must have. For instance considering the production of different mobile devices we could define a constraint requiring that any mobile device variant has to include a screen. An *Optional* feature is used to represent characteristics that a product can have, but a fully functional product can also be derived without including such a feature. For instance this could be the case of mechanisms supporting connection to 4G networks that could be included only in high-profile products. An *Alternative* feature represents characteristics that cannot be present together in a product. For instance a mobile device can have a standard screen or a touch screen, but not both. It is also possible to express relationships such as "*at least one feature in a set of features is needed in each product*"; this is done via *OR features* constraints. Additionally, *include relationship* constraints have been added to express that a feature selection implies the selection of another feature that is on a different branch of the tree; *exclude relationship constraints*, instead, are used to express that a feature selection requires to discard another feature that is on a different branch of the tree.

Once a FM has been defined it is possible to derive, according to the constraints, a specific product defining a *configuration* that expresses explicit features selection. In Figure 1 we depict a simple scenario of a family of mobile phones. Each mobile phone has to provide a display that can be touch screen or a standard display. Each phone can also have Internet connection, if it is included the 3G connection has to be mandatory and, optionally, 4G connection can be available. The features in gray in the FM are selected and they represent the configuration to generate a mobile phone product variant with a touchscreen display and a 3G Internet connection.

Researchers have proven that basic FM models are too restrictive to represent all the relationships between features useful to characterize a family of products [7]. As a result the FM notation has been extended to permit the definition of feature cardinality, permitting to define how many features in a set are needed to have a working product [8]. It means that features can be arranged into feature groups, where each feature group has a group cardinality. A group cardinality is an interval of form $<m\text{-}n>$, where $m,n \in \mathbb{Z} \wedge 0 \leq m \leq n \leq k$ where k is the number of features in the group.

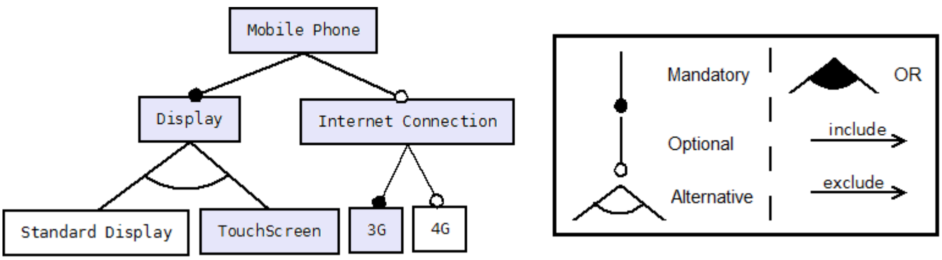

Figure 1. Feature Model Example and Configuration (Selected Features in Gray.

2.2. Business Process Modeling

A BP is ``a collection of related and structured activities undertaken by one or more organizations in order to pursue some particular goal" [9]. The accuracy of the BP modeling phase is critical for the success of an organization, in particular in scenarios in which it is necessary to adapt to changing requirements. In order to design a BP, different classes of languages have been investigated and defined.

In our work we refer to BPMN 2.0, an Object Management Group standard [10]. It is the most used language by domain experts due to its intuitive graphical notation. We have mainly used process diagrams, focusing on the point of view of system users. The BPMN 2.0 elements in Figure 2 are the core elements of the language and those we will use in our approach. In particular the following are concepts that can be modeled in BPMN 2.0, and their respective interpretation. Events are used to represent something that can happen. Activities are used to represent a generic work to perform within a BP. Gateways are used to manage the flow of BP both for parallel activities and choices. Data Objects permit to model documents, data, and other artifacts used and updated during the BP, in most of the cases activities take data objects as input, modify them, and give them back as output.

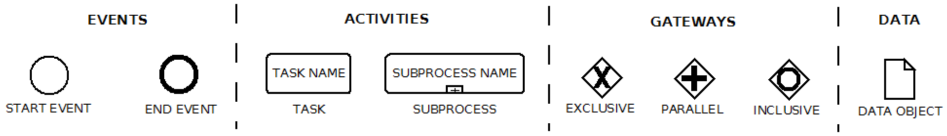

Figure 2. BPMN 2.0 Core Elements.

3. Approach

The proposed approach is organized in three main steps as shown in Figure 3. The main actors involved in the approach are: a regional *Competence Center* and a set of PAs. The *Competence Center* should include BP designers, domain experts and legislation experts; instead, the PAs are the organizations that have to deliver the services described in the referring laws. In particular the approach is organized as follows.

- The first step aims at defining a data object model that explicitly provides information about documents that are needed by PAs. This step includes knowledge acquisition through the study of laws of the involved PA services. It is carried on only once by the competence center.
- The second step aims at defining general models that constitute the basis for the definition of process variants for each specific deployment context. General models will be codified using CPMs notation, and they will be designed considering the data object model and the referring laws. This step should be carried on only once by the competence center and it will permit to derive models that will include activities that have to be carried on, the relations among them, and the data they possibly get in input or produce in output. The activities defined in the CPMs will be also linked to the data objects of the data object model.
- The last step concerns the derivation of the fully specified BP variants configuring the CPMs and the data object model. Each involved PA has to select the activities in the CPMs considering its organizational structure, then a set of BP variants is extracted (exactly one for each CPM).

Data Modeling via Feature Model. A data object model is a representation of data that may help a BP Designer during the design of a BP. In particular, with a data object model, a BP Designer can have an immediate view of the data objects that are *mandatory* or *optional* for a correct BP configuration. Then, having an overview of the data to include in the BP can facilitate the choice of which activities have to be included in the BP (e.g. activities that operate on those data). A data object model is generated from the first step of our approach and it refers to a high level representation of the data objects involved in the BP without going into details like their organization in tables, columns, or the physical means used to store them. In our case we use FM to represent the structure of data objects, whether they are composed by other data objects and to express variability in their composition. Moving from the root to the leaves means going from a data object to the parts that compose it.

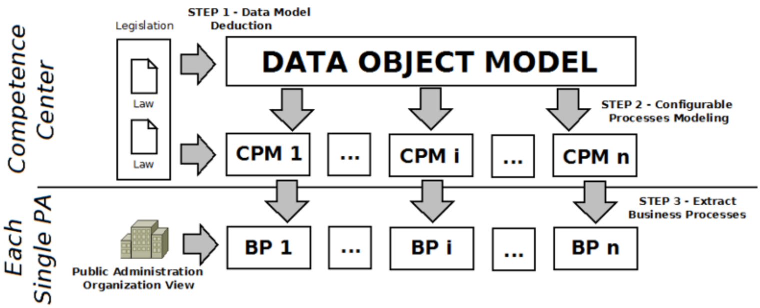

Figure 3. The Proposed Approach.

Configurable Processes Modeling via BPFM. A set of CPMs is the output of the step 2 of the approach, since many BPs can be described by a law. In this paper we use Business Process Feature Model (BPFM), a feature model extension to deal with variable BPs; its suitability to model families of BPs for PAs, has been proved in [11] [12]. A BPFM model is constituted by a tree of related activities. The root identifies the family of the BPs under analysis. Each internal (non-leaf) activity denotes a sub-process, and the external (leaf) activity represents a task. BPFM introduces different levels of detail in the BP family specification going up- down on the tree. BPFM also provides the possibility to define constraints between activities in two adjacent levels of the tree. Each constraint has only one father activity, and it has one or more child activities depending on the type of the constraint. Constraints are used to express if child activities have to be inserted in the BP variant and if they have to be included in each execution path of a BP variant.

Data Objects toward Configuration and Variant Derivation. In the third step each PA has to derive their BP variants from the data object model and the CPMs resulted in the steps 1 and 2. The PA has to consider its internal structure in order to select the data objects and the activities it needs in the BPs. The activities selection has to be done for each CPMs.

The selection of activities is generally done manually by a BP designer which has to select the activities to include in the BP. Since the Competence Center also provides a Data Object Model together with the CPM, the BP designer may be guided, in the choice of those activities, by the presence of the constraints imposed on the data objects.

4. SUAP Case Study

To describe our approach we focused on the SUAP case study. It involves more than 110 different BPs that support more than 150 data objects types in order to provide services to citizens. In particular, we applied the approach to the case of the Marche Region (Italy) that coordinates the implementation of the SUAP service for 239 municipalities or aggregations of municipalities, and 45 third parties administrations that can be involved as third parties. For the sake of space we focus in this paper to only one BP related to the *Standard request to start a Business Activity* for shops in a fixed location. This is the BP in which an entrepreneur requests to the municipality, and to third parties organizations, the permission to start a business activity. The entrepreneur will have to wait the office decision before starting the activity.

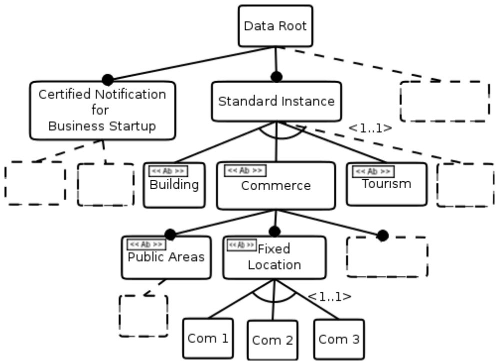

Figure 4. SUAP Data Object Model.

The Data Object Model of the SUAP case study is reported in Figure 4. It allows us to express that the Data Root must be composed by data objects such as: Certified Notification for Business Startup, Standard Instance, and other data objects that we do not represent for a matter of space. The use of cardinality in the feature model allows us to express that the data object instance must include only one part that composes it out of: Building, Commerce, Tourism and others. We use abstract data object to represent a categorization of data objects that go under a same topic; Commerce, in this case, groups together data objects such as: Public Areas, Fixed Location, and others. It is also mandatory that Commerce is composed by only one of those data objects. Finally, in our case study the Fixed Location data object must be composed by one out of the parts: Com 1, Com 2, Com 3. Representing data in this manner, gives to the BP designer the possibility to have a view of all the data objects that are necessary for configuring the business processes required by the law.

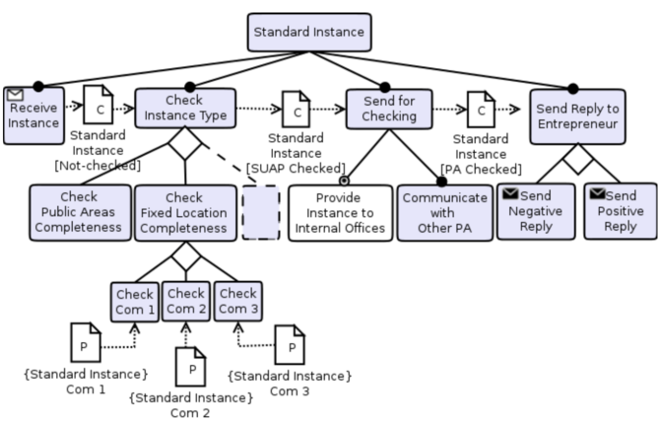

Figure 5. SUAP Configurable Process Variant.

After modeling the data view, the second step of our approach implies that all the CPMs are developed by a Competence Center. In this paper for a matter of space we just present the CPM of the *Standard request to start a Business Activity*, considering only the activities needed for the case of fixed location shop; we modeled it in BPFM notation. Since the paper is not focused on BPFM notation, then we just highlight the variability of the case. In particular, two possible BP variants can be generated from the BPFM model, one in the case of the SUAP office is provided by a single municipality and the other in the case the office is provided by an aggregation of municipalities. In the model it is represented by a *BPFM Special Case Constraint* linked to the activity *Provide Instance to Internal Offices*.

Performing the third step of our approach, the Competence Center has already designed the Data Object Model and the CPM; the BP designer of each single PA can then derive a BP variant defining a configuration that permits to express which data and activities to include. The activities in gray in the FM of Figure 5 represent a possible configuration, while Figure 6 illustrates the corresponding BP variant derived thanks to the mapping rules defined in BPFM.

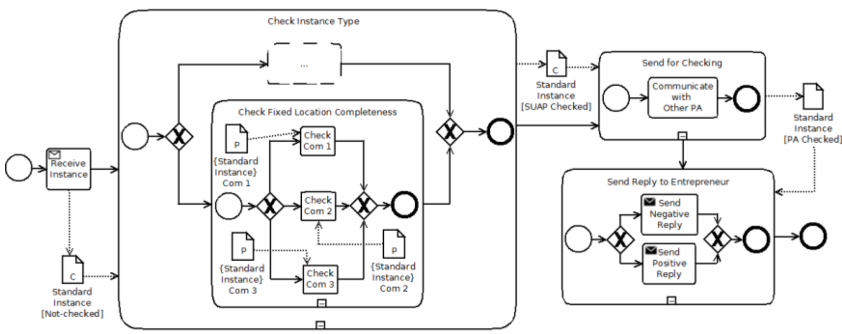

Figure 6. Business Process Variant.

5. Related Works

We found other works that attempt to provide approaches and tools to simplify the management of Business Processes in the Public Administration sector. In [2] the authors underline the importance of using Configurable Process Models for the Swedish Public Sector, providing also positive results of evaluation questionnaire and feedback from municipal officials.

Other works attempt to derive a Business Process model directly from the law. Particularly interesting is the approach used in [13] which focuses on the natural language that composes the law and the importance of having a visual representation such as a derived Business Process model. The authors derive a formal, and therefore verifiable, model directly from the code of law to proof the thesis that many laws directly implicate executable process sets. They use an existing graphical and textual illustration of the Swiss obligation law as basis and they were able, with the help of the Semantic Process Language [14], to translate the illustration into an executable Module Net.

In our approach we do not refer to a direct translation of the law text into a model, but we rely on the presence of a competence center that does this work manually.

6. Conclusions and Further Work

In this paper we presented an three steps approach to derive public services BPs from legislation. In the first step a data object model is extracted from legislation by a competence center, then in the second step the competence center models a set of BP families highlighting the variability between the BP variants. In the last step each single PA extracts its own BP variant from the families with the help of the data object model. We applied the approach to the Marche Region SUAP scenario in which 239 municipalities and 45 third parties administrations are involved in the execution of more than 110 BPs. This first experiment made with the proposed approach, provided encouraging results and permitted to model quite easily a complex scenario, and to derive the corresponding BPs.

In the future we plan to apply the approach to other scenarios in the Italian Public Administration, including a check of the compliance between the data model and the CPMs in order to ensure the correctness of the models and of the relations modeled by the competence center.

Acknowledgments

This research has been partially founded by EU project LearnPAd GA: 619583 and by the Project MIUR PRIN CINA - 2010LHT4KM.

References

[1] Reichert, M. and Weber, B.: Enabling flexibility in process-aware information systems: challenges, methods, technologies. Springer Science & Business Media (2012)

[2] Lönn, C.M., Uppström, E., Wohed, P., Juell-Skielse, G.: Configurable process models for the Swedish public sector. Advanced Information Systems Engineering. Springer (2012) 190-205

[3] Erkoçak, E., Açıkalın, S¸ .N.: Complexity theory in PA and metagovernance. In: Chaos, Complexity and Leadership 2013. Springer (2015) 73-94

[4] van der Aalst, W.M., Dreiling, A., Gottschalk, F., Rosemann, M., Jansen-Vullers, M.H.: Configurable process models as a basis for reference modeling. In: Business Process Management Workshops, Springer (2006) 512–518.

[5] Pohl, K., B¨ockle, G., Linden, F.J.v.d.: Software Product Line Engineering: Foundations, Principles and Techniques. Springer-Verlag (2005)

[6] Kang, K.C., Cohen, S.G., Hess, J.A., Novak, W.E., Peterson, A.S.: Feature-oriented domain analysis (FODA) feasibility study. Technical report, DTIC (1990)

[7] Capilla, R., Bosch, J., Kang, K.C.: Systems and Software Variability Management. Springer (2013)

[8] Czarnecki, K., Helsen, S., Eisenecker, U.: Staged configuration using feature models. In: Software Product Lines. Springer (2004) 266–283

[9] Lindsay, A., Downs, D., Lunn, K.: Business process - attempts to find a definition. Information and Software Technology 45 (2003) 1015–1019

[10] OMG: (Business Process Model And Notation - BPMN)

[11] Cognini, R., Corradini, F., Polini, A., Re, B.: Modelling process intensive scenarios for the smart city. In: EGov - 13th IFIP WG 8.5 International Conference, EGOV 2014, Dublin, Ireland, September 1-3, 2014. Proceedings. (2014) 147–158

[12] Cognini, R., Corradini, F., Polini, A., Re, B.: Using data-object flow relations to derive control flow variants in configurable business processes. In Fournier, F., Mendling, J., eds.: BPM 2014 Workshops, LNBIP 202. (2014) 210 – 221

[13] Olbrich, S., Simon, C.: Process modelling towards e-government–visualisation and semantic modeling of legal regulations as executable process sets. Electronic Journal of e-Government 6(1) (2008) 43–54

[14] Simon, C.: Integration of planning and production processes. Mathmod 2006, Special Session: Petrinets: Current Research Topics and their Application in Traffic Safety and Automation Engineering (2006)