Applications for Business Process Repositories Based on Semantic Standardization

Maya Lincoln[1] and Avi Wasser[2]

[1] ProcessGene Ltd
maya.lincoln@processgene.com
[2] University of Haifa, Israel
awasser@haifa.ac.il

Abstract. In recent years, researchers have become increasingly interested in developing frameworks and tools for the generation, customization and utilization of business process model content. One of the enabling central techniques for automated repository standardization is Natural Language Processing (NLP). This work reviews previous works on NLP standardization, and presents a set of derived Business Process Management (BPM) applications. We then discuss how these applications can be extended and improved for better utilization of the process repositories by (1) deploying a larger set of semantic models; and (2) integrating complementing applications.

Keywords: Business process model standardization, Business process repositories, Natural language processing.

1 Introduction

Business process repositories are considered an important resource of organizational knowledge. These repositories facilitate visibility into the business of organizations, and therefore have a central role in enterprise analysis, strategy, and Information Technology (IT) efforts [4]. Therefore, in recent years, researchers have become increasingly interested in developing methods and tools for promoting the utilization of process repositories, and the topic has been discussed intensively both in academia and in industry [17]. This work is aimed at suggesting methods for enabling the utilization of such process repositories for business process management applications.

Our work presents a content-based utilization framework that relies on the standardization of the content layer of business process repositories, as a basis for enabling several applications that leverage the usage of these knowledge reservoirs.

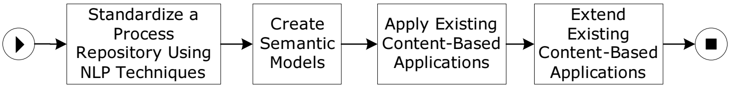

Fig. 1. A high-level framework for utilizing standardized process repositories for content-based applications

P. Herrero et al. (Eds.): OTM 2012 Workshops, LNCS 7567, pp. 173–182, 2012.

We propose a four-step meta process to utilize process repositories, as illustrated in Fig. 1 (using "Yet Another Workflow Language" (YAWL) [14]). First, we apply previously suggested methods for creating an operationally meaningful decomposition of a process repository. We use state-of-the-art Natural Language Processing (NLP) techniques to automatically decompose the content layer (text) of the repositories into its structured linguistic components (objects and actions and their qualifiers). As part of this decomposition, each business activity is encoded automatically as a *descriptor*, using the Process Descriptor Catalog ("PDC") notation [10]. The collection of all descriptors formulates a *descriptor space*, and distances between every two space coordinates are calculated in terms of business process conduct proximity. Second, by analyzing the generated decomposition, we create seven action and object models, that represent operational aspects of the process repository, as suggested in [8,7]. Third, we present applications from previous works that use action and object models for solving problems related to the utilization of the process repository in the following domains: (1) design of new process models; (2) validation of changes in the repository; (3) search of process segments in the repository, (4) similarity measurement between process models; and (5) construction of process data ontologies, as a basis for further understanding the logic of process models. As a fourth step, we discuss how these applications can be extended and improved for better utilization of the process repositories by (1) deploying a larger set of semantic models; and (2) integrating complementing applications.

The suggested framework is demonstrated using a process repository that consists 31 real-life processes and 183 related activities from the high-tech industry.

The rest of the paper is organized as follows: we present related work in Section 2, positioning our work with respect to previous research. In Section 3 we present an activity decomposition model that is gathered from previous works, and is used in this work as the foundation for creating action and object taxonomies. We then present applications from previous works that currently rely on some parts of the standardized repository, and discuss how they can be extended in Section 4. We conclude in Section 5.

2 Related Work

Research on standardization and analysis of the content layer of business process models mainly focuses on the analysis of linguistic components - actions and objects that describe business activities. Most existing languages for business process modeling and implementation are activity-centric, representing processes as a set of activities connected by control-flow elements indicating the order of activity execution [16]. In recent years, an alternative approach has been proposed, which is based on objects (or artifacts/entities/documents) as a central component for business process modeling and implementation. This relatively new approach focuses on the central objects along with their life-cycles. Such object-centric approaches include artifact-centric modeling [12,2], data-driven modeling [11] and proclets [13].

Although most works in the above domain are either object or activity centric, only a few works combine the two approaches in order to exploit an extended knowledge scope of the business process. The work in [5] presents an algorithm that generates an information-centric process model from an activity-centric model. The works in [10,8,7] present the concept of business process descriptor that decomposes process names into objects, actions and qualifiers, and suggest several taxonomies to express the operational knowledge encapsulated in business process repositories. In this work we take this model forward by: (a) testing it on real-life processes from the high-tech domain; (b) showing how the suggested taxonomies can assist in common usages of business process management.

3 The Activity Decomposition Model

This section describes a formal model of business process decomposition and analysis, gathered from previous works. We first introduce the descriptor model (Section 3.1). Then, based on the descriptor model, we introduce seven taxonomies of objects and actions (Section 3.2). To illustrate the taxonomies we make use of the high-tech repository (see Chapter 1).

3.1 The Descriptor Model

In the Process Descriptor Catalog model ("PDC") [10] each activity is composed of one action, one object that the action acts upon, and possibly one or more action and object qualifiers, as illustrated in Fig. 2. Qualifiers provide an additional description to actions and objects. State-of the art Natural Language Processing (NLP) systems, *e.g.*, the "Stanford Parser,"[1] can be used to automatically decompose process and activity names into *process/activity descriptors*. For example, the activity "Develop requested functionality" generates an activity descriptor containing the action "develop," without an action qualifier, the object "functionality" and the object qualifier "requested."

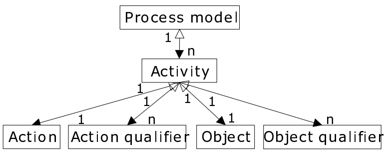

Fig. 2. The activity decomposition model

[1] `http://nlp.stanford.edu:8080/parser/index.jsp`

3.2 Action and Object Based Taxonomies

The descriptor model has two basic elements, namely objects and actions, and it serves as a basis for several state of the art taxonomies, as follows: (1) in [9] it was enhanced to create the *action hierarchy model*, the *object hierarchy model*, and the *action sequence model*, and the *object lifecycle model*; and (2) in [7] it was enhanced to create the *action scope model*, the *object grouping model*, and the *action influence model*.

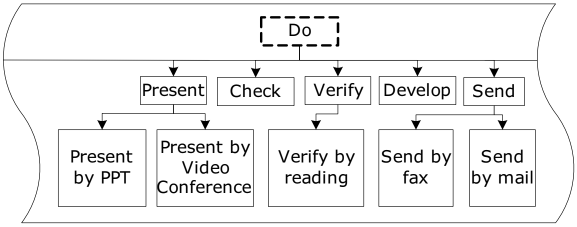

Fig. 3. Segment of an action hierarchy model from the high-tech repository

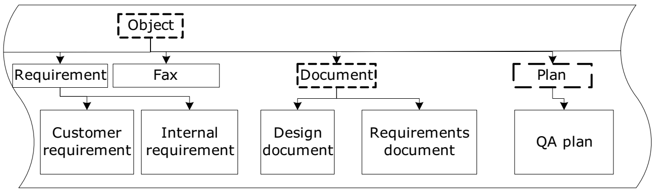

Fig. 4. Segment of an object hierarchy model from the high-tech repository

The Action and Object Hierarchy Models. The action and object hierarchy models organize a set of activity descriptors according to the hierarchical relationships among business actions and objects, respectively. This hierarchical dimension of actions and objects is determined by their qualifiers (see illustrations in Fig. 3 and Fig. 4). For example, consider the complete action "Send by fax." It is a subclass (a more specific form) of "Send" in the action hierarchy model, since the qualifier "By fax" limits the action of "Send" to reduced action range.

It is worth noting that some higher-hierarchy objects and actions are generated automatically by removing qualifiers from lower-hierarchy objects and actions. For example, the object "Plan" was not represented without qualifiers in the high-tech process repository, and was completed from the more detailed object: "QA plan" by removing its object qualifier ("QA") (see Fig. 4).

The Action Sequence Model. The action sequence taxonomy model organizes a set of activity descriptors according to the relationships among business

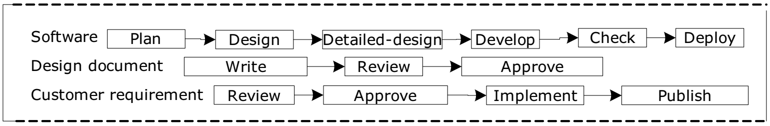

Fig. 5. Segment of an action sequence model from the high-tech repository

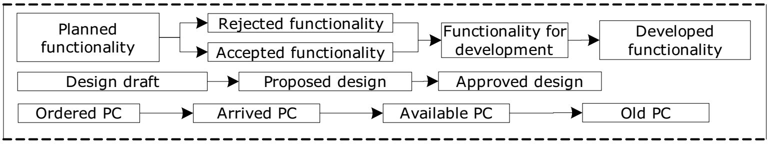

Fig. 6. Segment of an object lifecycle model from the high-tech repository

actions and objects in terms of execution order. In this model, each object holds a graph of ordered actions that are applied to that object. A segment of the action sequence model of a high-tech repository is presented in Fig. 5. For example, the object "Student" is related to the following action sequence: "Interview" followed by "Accept," "Sign," and finally "Update."

The Object Lifecycle Model. The object lifecycle taxonomy model organizes a set of activity descriptors according to the relationships among business actions and objects in terms of execution order. For example, the object "Functionality" is part of the following object lifecycle: "Planned functionality"–> "Rejected/Accepted functionality"–>"functionality for development"–> "Developed functionality."

The Action Scope Model. The action scope model represents the relationship between an action within a process name (a "primary action") and the actions in its corresponding process model. The fact that a process repository consists of pre-defined process models is being used for learning about the scope of actions in the following way. Each primary action in the repository is related with a set of directional graphs of actions that represent the order of actions within this primary action's segments. Since such a primary action can be part of more than one primary names, and since the same complete action may be represented more than once in the same process model segment - each edge in the action scope model is labeled with its weight, calculated by the number of its repetitions in the related process model segments. Graph splits are also represented in the action scope model.

Consider the following two processes from the high-tech repository: "Develop customer requirement" and "Develop a new idea." These processes are illustrated in Fig 7a. Using these two process models, it is possible to generate an action scope model for the action "Develop" (Fig 7b). According to this example, there are two optional action paths compatible to the "Develop" action starting by

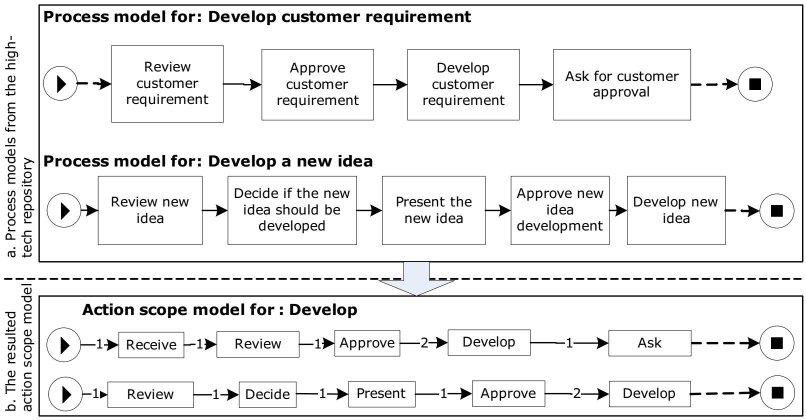

Fig. 7. A segment of the action sequence model for the action "Develop"

either "Receive" or "Review." Since "Develop" follows "Approve" twice in this model, the respective edge weight is set to 2.

The Object Grouping Model. The object grouping model represents the relationship between a primary object and the objects in its corresponding model segments. Since such a primary object can be part of more than one primary process segment, and since the same object may be represented more than once in the same process model segment - each object in the object grouping model is labeled with its weight calculated by the number of its repetitions in the related process model segments.

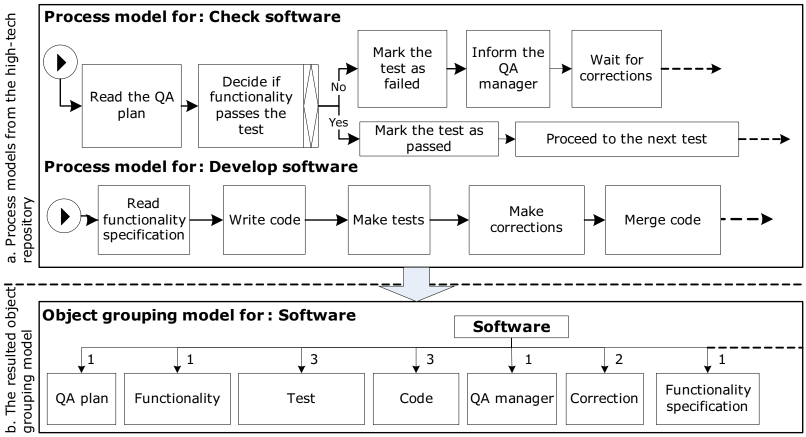

Fig. 8. A segment of the object grouping model for "Software"

To illustrate, consider two processes from the high-tech repository: "Check software" and "Develop software." These processes are represented by corresponding graph segments as illustrated in Fig 8a. Using these two process models, it is possible to generate an object grouping model for the object "Software," as illustrated in Fig 8b.

The Action Influence Model. An action influence model represents the relationship between a primary action and the flow of states (object qualifiers) of the primary object in model segments that correspond to the primary action. Each edge in the action influence model is labeled with its weight representing the number of its repetitions in the related process model segments.

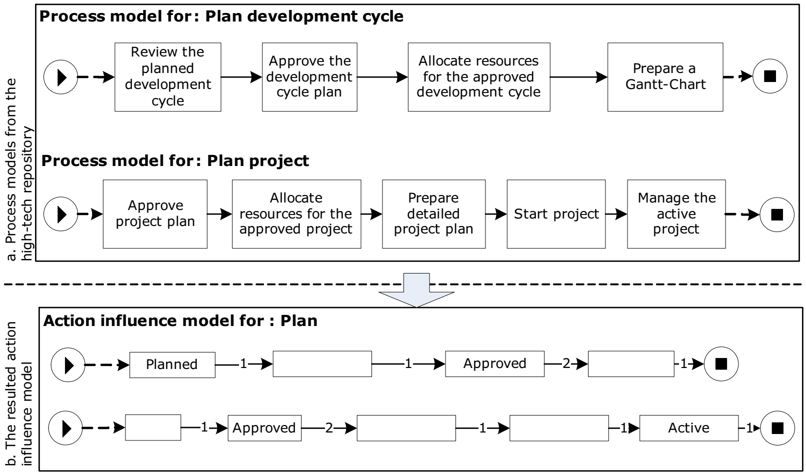

Fig. 9. A segment of the action influence model for the action "Plan"

To illustrate, consider the two process models named: "Plan development cycle" and "Plan project." They both deal with *plan*, but focus on different objects (see illustration in Fig. 9a). By following changes to the qualifiers of the primary object in these process models we end up with the action influence model for "Plan" as illustrated in Fig. 9b.

4 Applications

In Section 2 we showed how the content layer of business process repositories can be standardized using the activity decomposition model. In this section we show how such standardized process repositories can be utilized for several applications in the domain of business process management. The presented applications are based on previous works that applied semantic analysis of standardized business process repositories, using natural language processing techniques.

4.1 Machine-Assisted Design of Business Process Models

The work in [8] suggests a generic method for designing new business process models related to any functional domain. The suggested method guides business analysts that opt to design a new business model, by suggesting process steps (activities) that are relevant to the newly created process model. The business logic for such suggestions is extracted from the following four models: the action and object hierarchy model, the action sequence model and the object lifecycle model. Each activity is encoded automatically as a *descriptor*, using the "PDC" notation. The collection of all descriptors formulates a *descriptor space*, and distances between every two space coordinates are calculated in terms of business process conduct proximity.

To elaborate this work, it is possible to calculate descriptor space distances also using the three additional models: the action scope model, the object grouping model, and the action influence model. In addition, it is possible to integrate the process model search method proposed in [7] in order to generate model segment suggestions at each design phase, instead of single activity suggestions.

4.2 Content-Based Validation of Business Process Modifications

The work in [6] presents a content-based validation framework that uses organizational standards to evaluate the correctness of both newly designed and modified processes. A unique feature of the proposed framework, in the context of process validation, is the creation of a repository that captures organizational standards by using natural language processing analysis to capture simultaneously action and object patterns. The paper's contribution to the compliance domain is in the *dynamic* construction and adjustment of patterns - avoiding the need to design and maintain external, static rules. The authors propose to *automatically* extract business logic from process repositories using the PDC model, and the taxonomy models of action sequence, object lifecycle, and object and action hierarchies that support the validation process. The proposed method includes three steps for content-based validation: (1) deficiency identification (using existing descriptors as a benchmark reference), (2) validation score calculation, and (3) generation of a ranked list of possible corrections.

Similarly to the former method, we suggest elaborating this work, by referring also to the three additional models: the action scope model, the object grouping model, and the action influence model. In addition, we suggest to integrate the method proposed in [15] for measuring the similarity between process models using semantic analysis. This method may assist in ranking the generated correction suggestions, by comparing the suggestions to the original user intention (the change he made to the process repository).

4.3 Searching Business Process Repositories Using Operational Similarity

The search framework proposed in [7] receives natural language queries and returns a ranked list of related process models. The business logic is extracted from process repositories through the analysis of the following three taxonomies: the action scope model, the object grouping model, and the action influence model (see Section 3). The proposed method *dynamically* segments a process repository according to the ad-hoc request as expressed in the user's search phrase.

We propose elaborating the segmentation process by referring also to execution flows as expressed by the action sequence model and the object lifecycle model. In addition, an integration of the query-by-example approach proposed in [1] may extend the returned result range and may produce better search results.

4.4 Measuring Similarity between Process Models

The work in [15] proposes a method for measuring the similarity between process models using semantic analysis. Similarity in this paper is based on textual proximity of process components.

We suggest elaborating this similarity method proposed in [15] by referring also to operational similarity, as expressed in the activity decomposition model. Distances in this case can be calculated in terms of separating in the seven taxonomies.

4.5 An Automatic Construction of Process Data Ontologies

Some works focus on automatic construction of process data ontologies, as a basis for further understanding the logic of process models. Several works have been suggested, each focuses on extracting different aspects of the business conduct as encoded in the process repository. The work in [1] proposes a query-by-example approach that relies on ontological description of business processes, activities, and their relationships, which can be automatically built from the workflow models themselves. The work in [3] automatically extracts the semantics from searched conceptual models, without requiring manual meta-data annotation, while basing its method on a model-independent framework.

We suggest extending the above methods by targeting the automatic extraction and usage of the operational layer (the "how-to") and the business rules encapsulated in the process repository. These can be explored using the taxonomies presented in Section 3.

5 Conclusions

We presented a framework to standardize the content layer of business process models as a basis for utilizing them for business process management purposes. The proposed framework provides a starting point that can already be applied in real-life scenarios, yet several research issues remain open. We mention two such extensions here. First, extending the list of relevant applications that can utilize the standardized model. Second, adding a case study and experiments to measure the efficiency of the proposed improvements to the listed applications.

References

1. Belhajjame, K., Brambilla, M.: Ontology-Based Description and Discovery of Business Processes. In: Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., Ukor, R. (eds.) Enterprise, Business-Process and Information Systems Modeling. LNBIP, vol. 29, pp. 85–98. Springer, Heidelberg (2009)
2. Bhattacharya, K., Gerede, C.E., Hull, R., Liu, R., Su, J.: Towards Formal Analysis of Artifact-Centric Business Process Models. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 288–304. Springer, Heidelberg (2007)
3. Bozzon, A., Brambilla, M., Fraternali, P.: Searching Repositories of Web Application Models. In: Benatallah, B., Casati, F., Kappel, G., Rossi, G. (eds.) ICWE 2010. LNCS, vol. 6189, pp. 1–15. Springer, Heidelberg (2010)
4. Krumbholz, M., Maiden, N.: The implementation of enterprise resource planning packages in different organisational and national cultures. Information Systems 26(3), 185–204 (2001)
5. Kumaran, S., Liu, R., Wu, F.Y.: On the Duality of Information-Centric and Activity-Centric Models of Business Processes. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 32–47. Springer, Heidelberg (2008)
6. Lincoln, M., Gal, A.: Content-Based Validation of Business Process Modifications. In: Jeusfeld, M., Delcambre, L., Ling, T.-W. (eds.) ER 2011. LNCS, vol. 6998, pp. 495–503. Springer, Heidelberg (2011)
7. Lincoln, M., Gal, A.: Searching Business Process Repositories Using Operational Similarity. In: Meersman, R., Dillon, T., Herrero, P., Kumar, A., Reichert, M., Qing, L., Ooi, B.-C., Damiani, E., Schmidt, D.C., White, J., Hauswirth, M., Hitzler, P., Mohania, M. (eds.) OTM 2011, Part I. LNCS, vol. 7044, pp. 2–19. Springer, Heidelberg (2011)
8. Lincoln, M., Golani, M., Gal, A.: Machine-Assisted Design of Business Process Models Using Descriptor Space Analysis. In: Hull, R., Mendling, J., Tai, S. (eds.) BPM 2010. LNCS, vol. 6336, pp. 128–144. Springer, Heidelberg (2010)
9. Lincoln, M., Golani, M., Gal, A.: Machine-assisted design of business process models using descriptor space analysis. Technical Report IE/IS-2010-01, Technion (March 2010), http://ie.technion.ac.il/tech_reports/1267736757_MachineAssisted_Design_of_Business_Processes.pdf
10. Lincoln, M., Karni, R., Wasser, A.: A Framework for Ontological Standardization of Business Process Content. In: International Conference on Enterprise Information Systems, pp. 257–263 (2007)
11. Müller, D., Reichert, M., Herbst, J.: Data-Driven Modeling and Coordination of Large Process Structures. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 131–149. Springer, Heidelberg (2007)
12. Nigam, A., Caswell, N.S.: Business artifacts: An approach to operational specification. IBM Systems Journal 42(3), 428–445 (2003)
13. Van der Aalst, W.M.P., Barthelmess, P., Eliis, C.A., Wainer, J.: Proclets: A framework for lightweight interacting workflow processes. International Journal of Cooperative Information Systems 10(4), 443–482 (2001)
14. van der Aalst, W.M.P., Ter Hofstede, A.H.M.: YAWL: yet another workflow language. Information Systems 30(4), 245–275 (2005)
15. van Dongen, B.F., Dijkman, R., Mendling, J.: Measuring Similarity between Business Process Models. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 450–464. Springer, Heidelberg (2008)
16. Wahler, K., Küster, J.M.: Predicting Coupling of Object-Centric Business Process Implementations. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 148–163. Springer, Heidelberg (2008)
17. Yan, Z., Dijkman, R., Grefen, P.: Business Process Model Repositories-Framework and Survey. Technical report. Beta Working Papers