Ontology Based Method for Supporting Business Process Modeling Decisions

Avi Wasser[1] and Maya Lincoln[2]

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

Abstract. This work suggests a method for machine-assisted support of business process modeling decisions, based on business logic that is extracted from process repositories using a linguistic analysis of the relationships between constructs of process descriptors. The analysis enables the setup of a descriptor space, in which it is possible to analyze business rules and logic. The suggested method aims to assist process designers in modeling decision making, based on knowledge that is encapsulated within existing business process repositories. The method is demonstrated using a real-life process repository from the higher-education industry.

Keywords: Business process modeling decisions, Business process repositories, Semantic analysis, Process ontologies, Natural language processing.

1 Introduction

Our work presents a content-based utilization framework that standardizes the content layer of business process repositories, aiming to support the decision making of business analysts during their modeling work.

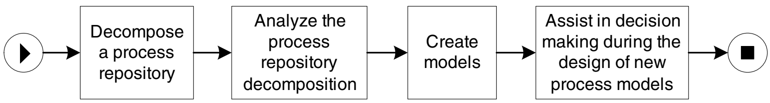

Fig. 1. A high-level framework for utilizing process repositories to support business process modeling decisions

We propose a four-step meta process to utilize process repositories for modeling decision support, as illustrated in Fig. 1 (using "Yet Another Workflow Language" (YAWL) [11]). First, we create 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

P. Herrero et al. (Eds.): OTM 2012 Workshops, LNCS 7567, pp. 505–514, 2012.

("PDC") notation [6] . As described in [4], 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, we analyze the generated decomposition and third, we create action and object models, that represent operational aspects of the process repository. In total, seven such models are proposed in this work. As a fourth step, we use the above action and object models for creating a methodology aimed at assisting process analysts in decision making during the design of new processes. We instantiate the generic methodology for the following four decision support mechanisms: (1) generation of a next activity for the designed process model; (2) validation of changes made to an existing process repository; (3) search for process models using natural language queries. This mechanism can support decisions such as whether some parts of the modeled process already exist in the current repository; and (4) comparison between process models. The suggested method was demonstrated using a process repository that consists 24 real-life processes and 170 related activities from the higher-education 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 used as the foundation for creating action and object taxonomies. We then present the suggested method for decision support during process design 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 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 works in [6,4,3] 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 higher-education domain; (b) showing how the suggested taxonomies can support business process modeling decisions.

Research on decision making during business process modeling mainly focuses on analyzing certain aspects of the rational encapsulated in the process model as a basis for reasoning and understanding the model. The work in [9] introduces a rationale-based architecture model that incorporates the design rationale, on which traceability techniques are applied for change impact analysis and root-cause analysis, thereby allowing software architects to better understand and reason about an architecture design. The work in [10] suggests a framework for supporting ontology based decision making, by modeling and visualizing ontological commitments using Semantic Decision Rule Language (SDRule-L). [1] proposes an approach to start from real known decision structures (like decision table hierarchies) and transform these into process models, in order to achieve tracability and maintainability during decision (re)design. The work in [8] focuses on context-aware decisions during process design, and the work in [7] presents an approach to process decision automation that incorporates data integration techniques, enabling significant improvements in decision quality. In this work we advance state of the art frameworks by extracting the *operational* logic encapsulated in process repositories using the process descriptor model and its derivative taxonomies.

3 The Activity Decomposition Model

This section describes a formal model of business process decomposition and analysis. 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 higher education repository (see Chapter 1).

3.1 The Descriptor Model

In the Process Descriptor Catalog model ("PDC") [6] 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, using UML relationship symbols. Qualifiers provide an additional description to actions and objects. In particular, a qualifier of an object is roughly related to an object state. 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 "Manually update student information" generates an activity descriptor containing the action "update," the action qualifier "manually," the object "information" and the object qualifier "student."

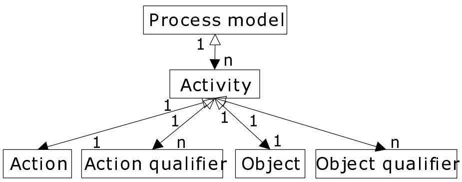

Fig. 2. The activity decomposition model

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 [5] it

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

was enhanced to create the *action hierarchy model*, the *object hierarchy model*, and the *action sequence model*; and (2) in [3] it was enhanced to create the *object lifecycle model*, the *action scope model*, the *object grouping model*, and the *action influence model.*

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: an addition of a qualifier to an action or an object makes them more specific, since the qualifier limits their meaning to a specific range.

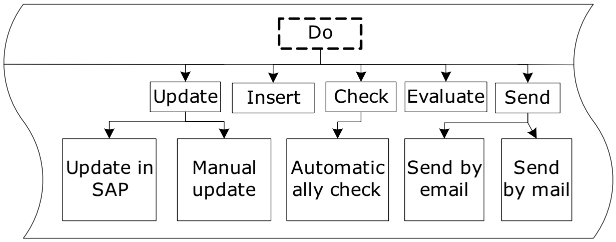

Fig. 3. Segment of an action hierarchy model

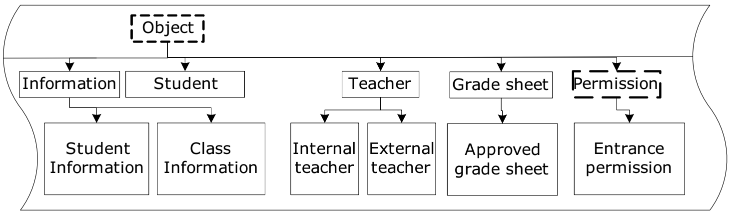

Fig. 4. Segment of an object hierarchy model

To illustrate the hierarchical dimension, a segment of the action hierarchy model of a higher education repository is presented in Fig. 3 and a segment of the object hierarchy model of a a higher education repository is presented in Fig. 4. In the action hierarchy model, for example, the action "Send by mail" is a subclass (a more specific form) of "Send," since the qualifier "by mail" limits the action of "Send" to reduced action range.

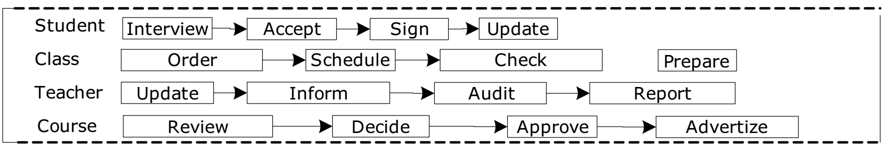

Fig. 5. Segment of an action sequence model

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

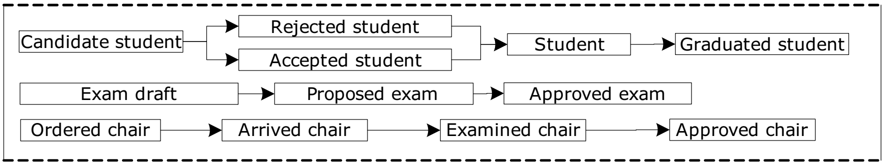

Fig. 6. Segment of an object lifecycle model

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. In this model, each object holds a graph of ordered objects that expresses the object's lifecycle, meaning - the possible ordering of the object's states. A segment of the object lifecycle model of a higher education repository is presented in Fig. 6. For example, the object "Student" is part of the following object lifecycle: "Candidate student"–> "Rejected/Accepted student"–>"Student"–> "Graduated student."

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 higher education repository: "Review student grades" and "Review teacher performance." These processes are represented in the higher education repository by corresponding graph segments as illustrated in Fig 7a. Using these two process models, it is possible to generate an action scope model for the action "Review" (Fig 7b). According to this example, there are two optional action paths compatible to the "Review" action starting by either "Receive" or "Review." Since "Decide" follows "Review" twice in this model, the respective edge weight is set to 2.

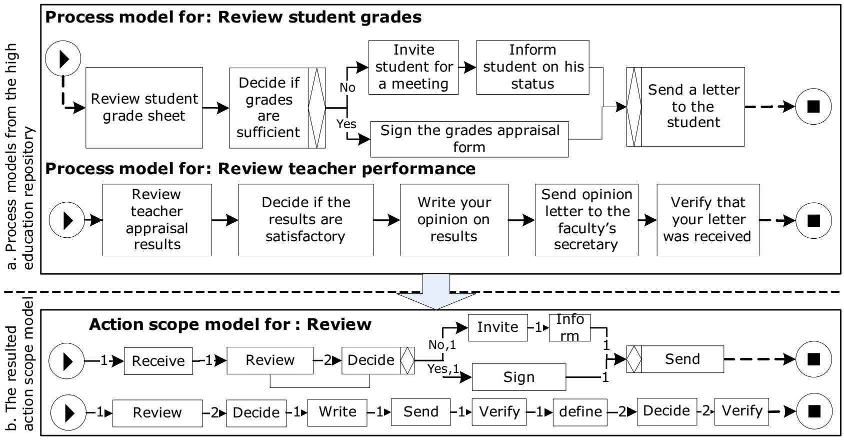

Fig. 7. A segment of the action sequence model for the action "Review"

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.

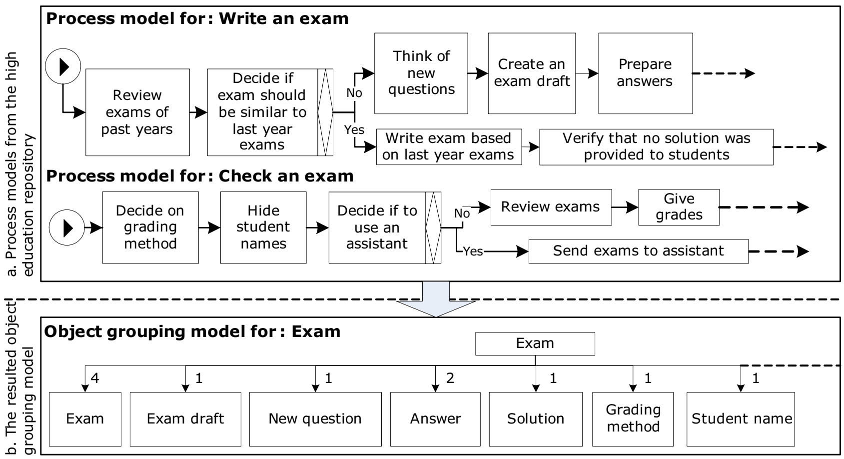

Fig. 8. A segment of the object grouping model for "Exam"

To illustrate, consider two processes from the higher education repository: "Write an exam" and "Check an exam." These processes are represented in the higher education repository 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 "Exam," 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.

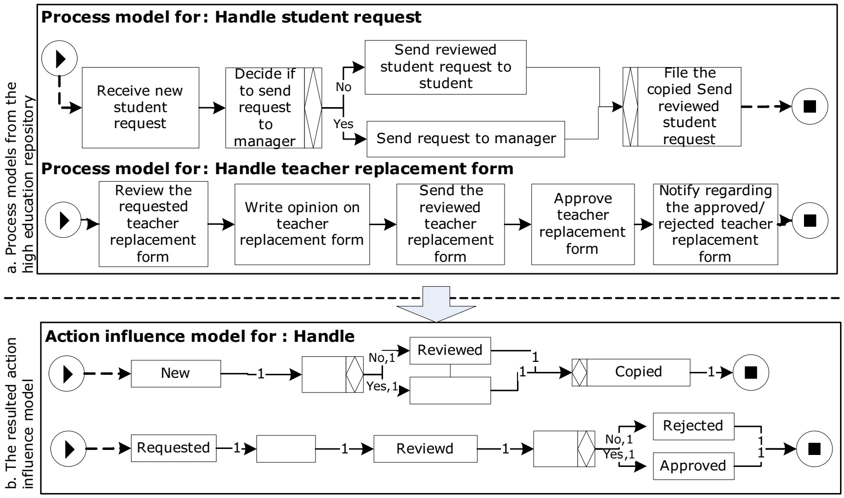

Fig. 9. A segment of the action influence model for the action "Handle"

To illustrate, consider the two process models named: "Handle student request" and "Handle teacher replacement form." They both deal with *handle,* but focus on different objects, "student request" and "teacher replacement form." These processes are illustrated 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 "Handle" as illustrated in Fig. 9b. In this example, the object state changes from "Reviewed" to "Copied" once in its corresponding process models and therefore the corresponding edge in the action influence model is labeled with weight of 1. In addition, empty rectangles represent no qualifiers.

4 Method for Supporting Business Process Modeling Decisions

In this section we present a framework for supporting business process modeling decisions based on the extraction of operational business logic as expressed in the descriptor model and its derivative taxonomies (Section 3).

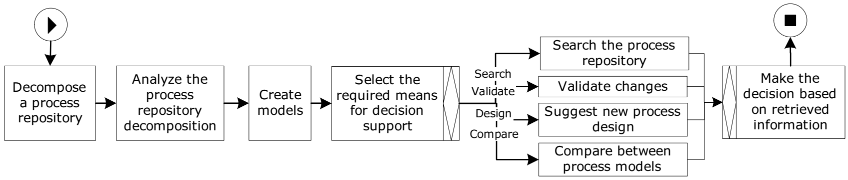

Fig. 10. A framework for supporting business process modeling decisions

We propose a six-step process to support business process modeling decisions, as illustrated in Fig. 10 (using "Yet Another Workflow Language" (YAWL) [11]). First, we create 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.

Second, we analyze the generated decomposition and third, we create seven action and object models, that represent operational aspects of the process repository (Section 3.1).

As a fourth step, the process analyst decides which support is required for his current decision making. Decision support options include the following mechanisms: (1) generation of a next activity for the designed process model; (2) validation of changes made to an existing process repository; (3) search for process models using natural language queries. This mechanism can support decisions such as whether some parts of the modeled process already exist in the current repository; and (4) comparison between process models. In the fifth step we use the above action and object models for providing the required assistance. Although each mechanism is different they all share the following steps. They start with a user request phrased in natural language. Then, this input is decomposed into linguistic components. As a result of this phase both the user intention (input) and the methodology's underlying knowledge (the process repository) are represented in a common language. The next phase includes an analysis of the process repository and the user input aimed at fulfilling the specific goal. As a result, a set of solution suggestions is generated and then ranked according to the user's input, so that higher ranked suggestions are believed to be closer to the user needs. The specific methods for next activity generation, validation, search and comparison are detailed in Sections 4.1,4.2,4.3, and 4.4, correspondingly. Finally, at the sixth step the process analyst receives suggestions and supporting information and makes his decision accordingly.

4.1 Generation of the Next Activity for the Designed Process Model

This mechanism 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. For this purpose we use the framework suggested in [4], in which 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. Each time the process analyst requests activity suggestions as part of the design process, the mechanism outputs a list of options, sorted according to the option's relevance to the modeled process and the current process repository.

4.2 Validation of Business Process Modifications

This mechanism assists process analysts in deciding whether modifications they made in the process repository are valid. To achieve this goal, we use the framework for content-based validation presented in [2], that uses organizational standards to evaluate the correctness of both newly designed and modified processes. This framework *automatically* extracts business logic from process repositories using the PDC model. Each process step is encoded automatically as a *descriptor* that represents objects, actions, and qualifiers. The collection of all process descriptors formulates taxonomy models of action sequence, object lifecycle, and object and action hierarchies that support the validation process.

The 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.

4.3 Search for Process Models Using Natural Language Queries

The mechanism at this phase enables deciding whether a process model is already represented in the process repository. To do that, we use the search framework proposed in [3], which 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.

4.4 Comparison between Process Models

This mechanism assists process analyst in deciding whether two or more process models are similar, and indicating which segments they share in common. To provide this capability, we use the work in [12] that proposes a method for measuring the similarity between process models using semantic analysis. Similarity in this work is based on textual proximity of process components. We elaborate this similarity method 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.

5 Conclusions

We proposed a framework for machine-assisted support of business process modeling decisions, based on business logic that is extracted from process repositories using a linguistic analysis of the relationships between constructs of process descriptors.

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 supporting decision mechanism - to support additional decision use-cases. Second, adding a case study and experiments to measure the efficiency of the proposed framework.

References

1. De Roover, W., Vanthienen, J.: On the Relation between Decision Structures, Tables and Processes. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM-WS 2011. LNCS, vol. 7046, pp. 591–598. Springer, Heidelberg (2011)
2. 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)
3. 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)
4. 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)
5. 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_Machine-Assisted_Design_of_Business_Processes.pdf
6. 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)
7. Niedermann, F., Maier, B., Radeschütz, S., Schwarz, H., Mitschang, B.: Automated Process Decision Making Based on Integrated Source Data. In: Abramowicz, W. (ed.) BIS 2011. LNBIP, vol. 87, pp. 160–171. Springer, Heidelberg (2011)
8. Ploesser, K., Recker, J., Rosemann, M.: Supporting Context-Aware Process Design: Learnings from a Design Science Study. In: Muehlen, M.Z., Su, J. (eds.) BPM 2010 Workshops. LNBIP, vol. 66, pp. 97–104. Springer, Heidelberg (2011)
9. Tang, A., Jin, Y., Han, J.: A rationale-based architecture model for design traceability and reasoning. Journal of Systems and Software 80(6), 918–934 (2007)
10. Tang, Y., Meersman, R.: Sdrule markup language: Towards modeling and interchanging ontological commitments for semantic decision making. In: Handbook of Research on Emerging Rule-Based Languages and Technologies: Open Solutions and Approaches. IGI Publishing, USA (2009)
11. van der Aalst, W.M.P., Ter Hofstede, A.H.M.: YAWL: yet another workflow language. Information Systems 30(4), 245–275 (2005)
12. van Dongen, B.F., Dijkman, R.M., 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)