$\mathcal{PN}av$: Process Navigator for the Design of New Business Process Models

Maya Lincoln and Avigdor Gal

Technion - Israel Institute of Technology
mayal@technion.ac.il, avigal@ie.technion.ac.il

Abstract. In this demonstration we introduce a prototype of $\mathcal{PN}av$, a process navigator that assists designers in designing new process models. To do that, $\mathcal{PN}av$ generates activity suggestions for the newly generated process models. The business logic for such suggestions is extracted from process repositories through the analysis of existing business process model activities.

1 Introduction

Enterprise process repositories contain hundreds of business processes, developed over the years to support enterprise activities. Such repositories contain a large number of activities that can be re-used when redesigning existing processes or whenever the need for new processes arises. Process modeling is considered a manual, labor intensive task, whose outcome depends on personal domain expertise with errors or inconsistencies that may lead to bad process performance and high process costs [3]. Hence, reuse of activities can save design time and support non-expert designers in creating new business process models.

In this demonstration we introduce a prototype of $\mathcal{PN}av$, a process navigator that assists designers in designing new process models. To do that, $\mathcal{PN}av$ generates activity suggestions for the newly generated process models. The business logic for such suggestions is extracted from process repositories through the analysis of existing business process model activities. Each activity is encoded automatically as a *descriptor*, using the PDC notation [2,1]. 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.

In the Process Descriptor Catalog model ("PDC") [2,1] each activity is composed of one action, one object that the action acts upon, and possibly one or more action qualifiers and object qualifiers. For example, the activity name"Manually complete a supplier maintenance form" is decomposed into (action='complete' , object='form' , action qualifier='manually' , object qualifier='supplier maintenance').

The model has two basic elements, namely objects and actions, and we delineate four taxonomies from them: an *action hierarchy model*, an *object hierarchy model*, an *action sequence model* and an *object lifecycle model*. The business action and object taxonomy models organize a set of activity descriptors according

to relationships among business actions and objects both hierarchically and in terms of execution order.

An example from the Oracle Business Model (OBM)[1] is given in Fig. 1 and Fig. 2, showing a section of the business action and object taxonomy models as a set of activity descriptors according to the relationships among business actions and objects both longitudinally (hierarchically) and latitudinally (in terms of execution order), as detailed next.

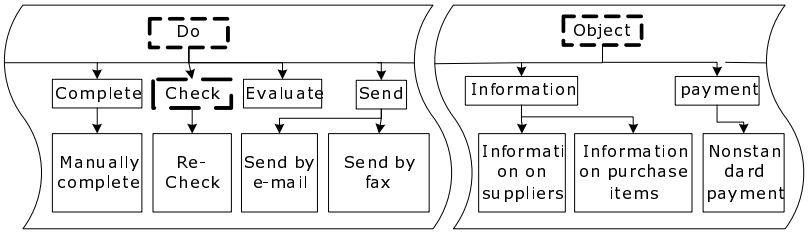


Fig. 1. Segments of the action hierarchy model and the object hierarchy model extracted from the OBM for procurement processes.

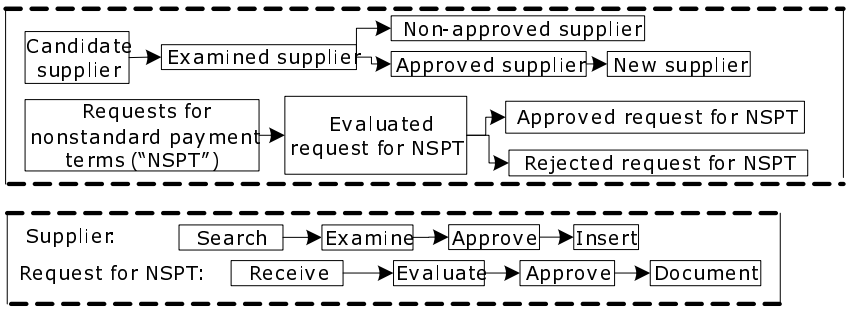


Fig. 2. Segments of the action sequence model and the object lifecycle model extracted from the OBM for procurement processes.

The longitudinal dimension of actions and objects is determined by their qualifiers. To illustrate the longitudinal dimension of the OBM workflows, a segment of the action hierarchy model and a segment of the object hierarchy model, both related to procurement processes, are presented in Fig. 1. To illustrate the latitudinal dimension of the OBM process repository, a segment of the action sequence model and a segment of the object lifecycle model are presented in Fig. 2.

[1] http://www.oracle.com/applications/tutor/index.html.

Based on the activity decomposition model, it is possible to visualize the operational range of a business process model as a descriptor space comprised of objects and actions, related to each other and among each other in different relationship types. A navigation within this space can be a powerful tool for analyzing and utilizing the underlying business process knowledge encapsulated within a business process repository. More details of the descriptor space and how to navigate it can be found in [1].

The demo is intended for both academics, interested in new techniques for designing business process models and practitioners, interested in state-of-the-art technology for design support tools. The tool can help automate the reuse of constructs gathered from predefined process models. Such a tool saves design time and supports non-expert designers in creating new business process models. The proposed software tool, can be used in real-life scenarios, yet several research and development are required in order to enable a more commercialized version of the tool.

2 Maturity

$\mathcal{PN}av$ implements a client-server architecture, in which the client is responsible for presenting and collecting data from the user and the server is responsible for processing the user's input data and suggesting directions for advancing the design process. Server side logic is implemented in PHP using a MySql database. The client runs within an Internet browser and is implemented in HTML and JavaScript, with AJAX calls to the server.

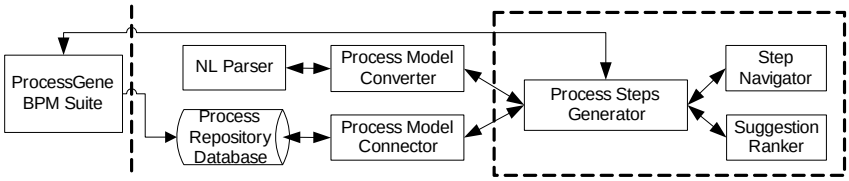


Fig. 3. $\mathcal{PN}av$ high-level architecture.

The server side high-level architecture includes five main components (see Fig. 3): (a) the navigator, responsible for managing and orchestrating the process design mechanism; (b) the process repository database that contains the existing business process repository, in terms of activity descriptors and object and action taxonomies; (c) the process model connector, which provides an interface for communicating with the process repository database; (d) the process model converter, responsible for converting inputted business process repositories into a normalized data structure as saved in the process repository database. Currently, our system supports the conversion of repositories expressed in BPEL

or BPMN; (e) the Natural Language (NL) parser, an existing web service for decomposing sentences into linguistic components. The parser we use is called "the Stanford Parser[2]. This web service decomposes activity names into descriptor components.

The navigator is further decomposed into four main components: (a) the process steps generator, responsible for producing suggested activities for each design phase. It communicates with all other components and presents the user at each stage with options for advancing the design process; (b) the step navigator, which is responsible for navigating in the process model database, and for retrieving a list of relevant activity options; (c) the suggestion ranker, responsible for ranking the suggested activity options at each stage.

The navigator is designed to connect to the ProcessGene BPM suite and to assist designers in designing new process models. Each time a user opens a new process model in ProcessGene[3], and defines the new process model name, $\mathcal{PN}av$ is activated and suggests the user to use its services. Once the user decides to use $\mathcal{PN}av$, she is guided in a step-by-step procedure that advises and supports the creation of the new process model.

We have conducted experiment sets with $\mathcal{PN}av$ using two case studies[1]. The first experiment set was based on an aviation process repository that covers airport activities starting from the passenger's entry to an airport, through document handling and security checks and terminating as the passenger boards the airplane[4]. The second experiment set was based on the Oracle Business Model, which serves for our demonstration script and will be discussed in details next.

$\mathcal{PN}av$ is currently installed and works well on a workstation running Windows XP, IIS6, PHP 4.8 and MySQL 5.0. This workstation serves as the server. A client, running Internet Explorer as the application container and presentation layer, will be available at the Demo site.

3 Script

We demonstrate the applicability of our ideas using 14 processes from the Oracle Business Model. Nine business processes are taken from the "Procurement" category, containing altogether 96 activities and five business processes are taken from the "Inventory" category, containing altogether 31 activities. The "Procurement" data set contains related, sequential activities and therefore encapsulates a focused operational area. The "Inventory" data set encapsulating a loosely coupled business logic regarding an extended business area.

Using the selected 14 processes we created a process repository database. The demo user shall provide a starting activity and then interact with $\mathcal{PN}av$ to receive a ranked list of suggested activities, refine the suggestions and move

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

[3] http://www.processgene.com

[4] Many thanks to Samia Mazhar and the BPM Group at QUT for providing access to the aviation process data.

on to the next activity. It should be noted that the tool offers only activities – designer needs to add junctions.

3.1 Example for using $\mathcal{PN}av$ for Designing a New Process Model

To illustrate the proposed software tool we present a short example from the field of procurement. The newly designed process, "Verify supplier details", is related to the procurement field, but is not covered by the OBM. Its goal is to verify whether the supplier details, as declared in the new supplier form, are correct.

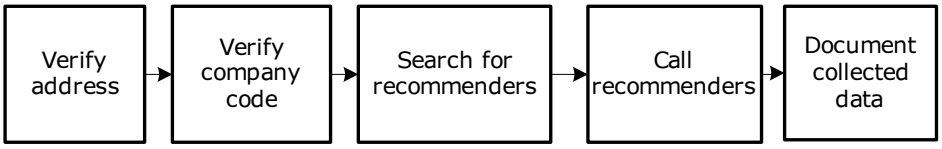
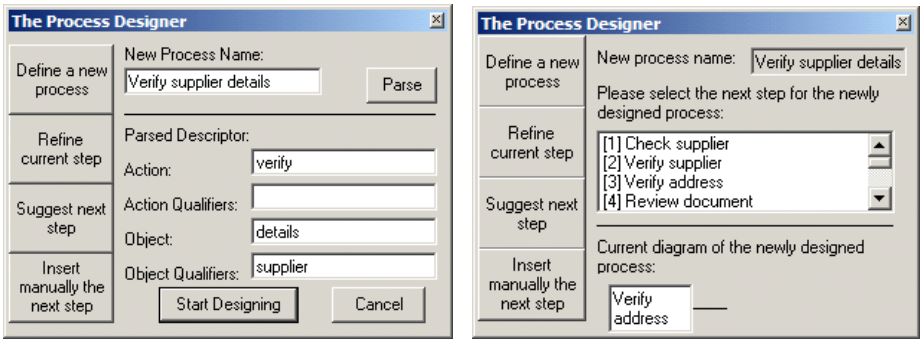


Fig. 4. The new designed process diagram for "Verify supplier details".

The example supports the design of a new business process for: "Verify supplier details." The generated output (new process model) of this example is illustrated in Fig. 4 as a YAWL diagram. The design process starts when the (human) process designer inserts into $\mathcal{PN}av$ the name of the new process ("Verify supplier details") which is then translated automatically into the following process descriptor: (action="verify", action qualifier=null, object="details", object qualifier="supplier") (see Fig. 5a) and determines that the first activity is: "Verify address." Respectively, the process delineator searches the descriptor space, looking for next activity possibilities. The result set includes the following activities: "[1] Check supplier," "[2] Verify supplier" and "[3] Verify address" (see Fig. 5b). The designer selects the option "Verify address" and decides that this activity is suitable.

The design process continues with four more design phases. The second phase required a refinement for the option "Search for additional data" - which was suggested as the next activity after "Verify company code." The refined option list includes the option: "[1] Search for recommenders," and this option was selected by the designer. Note that this activity was not represented "as is" in the business process repository.

The designer now wishes to design the new business process: "Review invoices to prevent fraud." An interesting observation in this design process is that the designer selects more often next step activities that share the same action applied on sibling objects. For example, the activity "Check signature" was followed by "Check date" and "Check payment terms." The business logic behind this phenomenon is that this process expresses a more independent business conduct in which there is only one party (the reviewer) which operates on one item (the invoice).



(a) The designer's request for designing the new process: "Verify supplier details"

(b) The third phase in designing the new process: "Verify supplier details"

Fig. 5. The designer's request for designing the new process: "Verify supplier details".

References

1. M. Lincoln, M. Golani, and A. Gal. Machine-assisted design of business process models using descriptor space analysis. In *Proceedings of the eigth International Conference on Buisness Process Modeling (BPM'2010)*, Hoboken, NJ, USA, September 2010.
2. M. Lincoln, R. Karni, and A. Wasser. A Framework for Ontological Standardization of Business Process Content. *International Conference on Enterprise Information Systems*, pages 257–263, 2007.
3. D. Muller, M. Reichert, and J. Herbst. Data-driven modeling and coordination of large process structures. *Lecture Notes in Computer Science*, 4803:131, 2007.