BPMNDiffViz: A Tool for BPMN Models Comparison*

Sergey Y. Ivanov[1], Anna A. Kalenkova[1], and Wil M.P. van der Aalst[2,1]

[1] National Research University Higher School of Economics, Moscow, 101000, Russia
syuivanov_1@edu.hse.ru,akalenkova@hse.ru
[2] Eindhoven University of Technology, Eindhoven, The Netherlands
w.m.p.v.d.aalst@tue.nl

Abstract. Automatic comparison of business processes plays an important role in their analysis and optimization. In this paper we present the web-based tool BPMNDiffViz, that finds business processes discrepancies and visualizes them. BPMN (Business Process Model and Notation) 2.0 - one of the most commonly used notations for process modeling was chosen as a representation. This tool implements a structural graph-based comparison analysis using an A* algorithm.

1 Overview

Today, various organizations are increasingly faced with modeling their business processes to reduce costs and to ensure a clear understanding of the processes used in the organization. Due to changes in legislation, introduction of innovations and other factors, business processes are constantly changing. Thus, system and business analysts involved in modeling of business processes need a tool for comparing process models and diagnosing their differences.

Comparison of business processes is an essential step within various scenarios of processes analysis. Process comparison is primarily intended to find discrepancies between a reference and a real process models. In that case, using *Process mining* [14] discovery techniques, a model of a real business process behavior can be obtained from an event log (in particular a BPMN model can be learned from an event log using ProM framework [6]). After that the comparison between this model and a reference process model is performed. Thus, the process model comparison serves as a conformance checking technique. Another example where the comparison of business processes can be used is the management of large process model repositories with hundreds or thousands models. Comparison techniques can significantly help in the classification of processes and finding duplicates. Last but not the least, business processes comparison is needed to analyze activities of two similar companies with different incomes.

Various metrics of business processes similarity were described in [4,5]. These metrics include: *label matching* similarity (either syntactic or semantic similarity of element names), *structural matching* similarity represented as a graph edit distance, and *behavioral matching* similarity. Label matching similarity is described in detail in [4].

* This work is supported by the Scientific Fund and the Basic Research Program of the National Research University Higher School of Economics and is supported by Russian Fund for Basic Research (project 15-37-21103).

Structural matching approaches typically use techniques such as greedy algorithms, A* algorithms, and context-based similarity measures [4,5,7]. Various behavioral matching methods and their implementations were proposed in [2, 3, 10, 15, 17].

Despite the range of techniques proposed in literature, tool support for structural processes comparison is still limited. A tool for merging and comparison of business processes called Synergia was presented in [12], the underlying approach [7] is based on finding structural mappings between graphs using heuristics and discovering maximal common subgraphs to merge them. This tool is used as a part of Apromore platform [8]. Another tool, which finds matches between business processes using structural metrics, is called ICoP Framework [16]. This tool constructs multiple mappings, groups and evaluates them in order to find the result mapping. A tool, which finds structural discrepancies between BPMN models using SiDiff platform [13], was presented in [11]. This tool was not realized yet and there is no information on the visualization of differences.

We present a *structural matching* tool called BPMNDiffViz, which compares process models represented in BPMN 2.0 format [1] - the most popular industrial standard for modeling business processes. In contrast to the previous *structural matching* tools BPMNDiffViz explicitly visualizes graph differences, stores them and provides statistics, assisting in analyzing model discrepancies [1]. BPMNDiffViz is mature enough and supports all BPMN modeling constructs including activities, sequence flows, subprocesses, gateways, events, data flow and others. We hope this tool will be used by the practitioners. The implemented matching algorithm finds the minimal graph edit distance between two processes (number of transformations, which should 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. Thus, the label matching and the structural matching metrics are used by BPMNDiffViz to find the minimal distance.

2 Functionality and Architecture

In this section we will present the functionality and architecture of the BPMNDiffViz tool. The proposed tool is based on client-server technology and is built on a three-tier architecture (Fig. 1). A server component and a client component (web forms) were implemented as parts of BPMNDiffViz tool.

The server component includes a server application and a database, which stores comparison results. Tomcat 7.0[2] is used as an application server and PostgreSQL[3] as a database management system. Spring MVC framework[4] is used to dispatch HTTP requests and responses. The server component allows to *load*, *store*, *search* on BPMN

[1] Note that we do not consider tools, such as `demo.bpmn.io/diff`, `academic.signavio.com`, which visualize graph differences using element identifiers. These tools are not applicable to compare models created by different tools, or even by one tool if models were generated independently.

[2] `http://tomcat.apache.org/`

[3] `http://www.postgresql.org/`

[4] `http://docs.spring.io/`

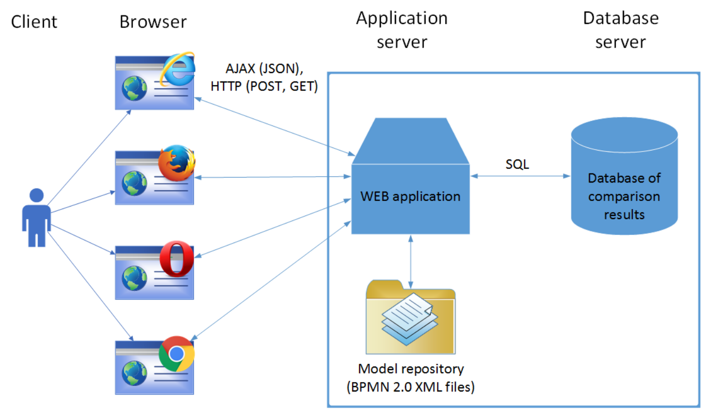

Fig. 1: Architecture of the BPMNDiffViz tool

diagrams, *compare* them, and *save* comparison results. To support the BPMN 2.0 notation Camunda framework[5] was utilized as a library for several reasons. It fully supports BPMN 2.0 elements. Furthermore, it allows good integration with Spring. Besides that, it incorporates a special JavaScript framework called *bpmn.io* to render models in a browser on a client side.

The client part is simply a set of web forms, which allows the user to perform the following actions: *load models* in BPMN 2.0 XML format, *visualize* BPMN models (by means of bpmn.io), *visualize* statistics of BPMN elements usage, *search* on loaded models, *set parameters* of comparison, *compare* two BPMN models, producing a final graph edit distance, a list of operations for transformation one model to another, and a comparison statistics, *visualize* and *save* comparison results.

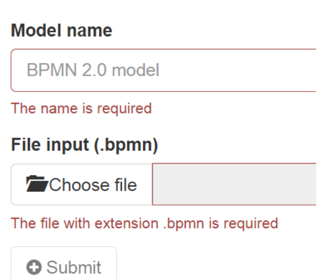

Fig. 2: Upload BPMN models

A model in BPMN 2.0 XML format can be loaded using a web form (Fig 2). After that this model is saved and appears in the list of loaded BPMN models. The search through this list is implemented as well. Each BPMN model can be visualized and rescaled (Fig. 3a.). For each such a model statistics of modeling elements usage is presented as a pie chart (Fig. 3b.). To compare two BPMN models first these models are selected from the list, then modification (insertion and deletion) costs and label matching algorithms [6] are specified for each type of BPMN elements. If modification costs were not explicitly specified, the tool will use default values. After

[5] `https://camunda.com/`

[6] Currently the system calculates Levenshtein distance [9] between two labels.

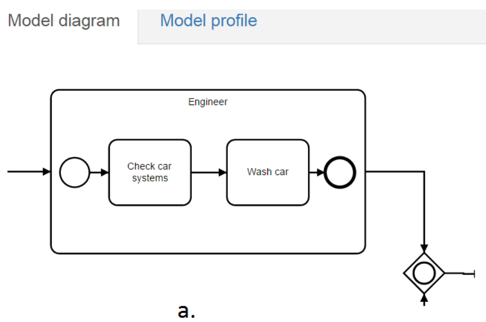

Fig. 3: Visualization of a BPMN model (this model contains 15 sequence flows)

that the comparison begins. A comparison result is visualized in such a way that all corresponding elements are highlighted in blue, all elements that should be deleted and added are highlighted in red and green respectively (Fig. 4 a.). A list of matchings, insertions and deletions and a total cost are visualized as well. Moreover, statistics of matched, deleted and inserted elements are explicitly shown in a form of pie charts. When a matching is selected from the list, the corresponding elements are highlighted (in color) and are moved to the center of the diagrams (Fig. 4 b.).

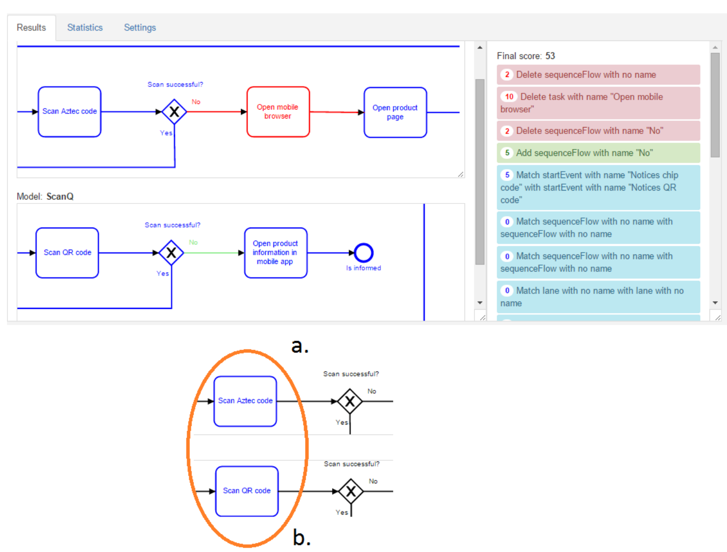

Fig. 4: A result of comparison

BPMNDiffViz was tested on various manually created models and models loaded from the Camunda collection. The tool has shown good performance for models that contain up to 30 elements. A link to the source code and a screencast are available at `http://pais.hse.ru/research/projects/CompBPMN/`.

In our future work, we plan to extended the tool to make it capable to recognize behaviorally similar constructions, which are structurally different. Also we plan to provide programming interfaces in order to achieve stronger integration with existing tools. Privacy aspects of the web-based tool will be worked out as well.

References

1. Business Process Model and Notation (BPMN). `http://www.omg.org/spec/BPMN/2.0/`.
2. A. Armas, P. Baldan, M. Dumas, and L. García-Bañuelos. Behavioral Comparison of Process Models Based on Canonically Reduced Event Structures. In *12th International Conference, BPM. Proc.*, pages 267–282, 2014.
3. A. Armas, P. Baldan, M. Dumas, and L. García-Bañuelos. BP-Diff: A Tool for Behavioral Comparison of Business Process Models. In *Proc. of the BPM Demo Sessions, BPM*, 2014.
4. R. Dijkman, M. Dumas, B.F. van Dongen, R. Käärik, and J. Mendling. Similarity of Business Process Models: Metrics and Evaluation. *Inf. Syst.*, 36(2):498–516, April 2011.
5. R. M. Dijkman, M. Dumas, L. García-Bañuelos, and R. Käärik. Aligning Business Process Models. In *Proc. of the 13th IEEE EDOC 2009*, pages 45–53, 2009.
6. A. Kalenkova, M. de Leoni, and W. M. P. van der Aalst. Discovering, Analyzing and Enhancing BPMN Models Using ProM. In *Proc. of the BPM Demo Sessions, BPM*, 2014.
7. M. La Rosa, M. Dumas, R. Uba, and R. Dijkman. Business Process Model Merging: An Approach to Business Process Consolidation. *ACM Trans. Softw. Eng. Methodol.*, 22(2), 2013.
8. M. La Rosa, H. A. Reijers, W. M. P. van der Aalst, R. M. Dijkman, J. Mendling, M. Dumas, and L. Garca-Bauelos. APROMORE: An Advanced Process Model Repository. *Expert Syst. Appl.*, 38(6):7029–7040, 2011.
9. V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady.*, 10(8):707–710, February 1966.
10. J. Mendling, B.F. van Dongen, and W.M.P. van der Aalst. On the Degree of Behavioral Similarity between Business Process Models. In *Proc. of WI-EPK 2007*, pages 39–58, 2007.
11. P. Pietsch and S. Wenzel. Comparison of BPMN2 Diagrams. In *Business Process Model and Notation - 4th International Workshop, BPMN 2012*, pages 83–97. Springer, 2012.
12. M. La Rosa and F. Gottschalk. Synergia Comprehensive Tool Support for Configurable Process Models. In *Proc. of the BPM Demo Sessions, BPM*, 2009.
13. C. Treude, S. Berlik, S. Wenzel, and U. Kelter. Difference Computation of Large Models. In *Proc. of the the SIGSOFT*, pages 295–304, 2007.
14. W. M. P. van der Aalst. *Process Mining - Discovery, Conformance and Enhancement of Business Processes.* Springer, 2011.
15. B.F. van van Dongen, R. M. Dijkman, and J. Mendling. Measuring Similarity between Business Process Models. In *CAiSE*, volume 5074 of *LNCS*, pages 450–464. Springer, 2008.
16. M. Weidlich, R. M. Dijkman, and J. Mendling. The ICoP Framework: Identification of Correspondences between Process Models. In *CAiSE*, volume 6051 of *LNCS*, pages 483–498. Springer, 2010.
17. M. Weidlich, A. Polyvyanyy, N. Desai, J. Mendling, and M. Weske. Process Compliance Analysis Based on Behavioural Profiles. *Inf. Syst.*, 36(7):1009–1025, 2011.