Behavior-Based Process Comparison in Apromore

Abel Armas-Cervantes[1], Nick R.T.P. van Beest[2], Marlon Dumas[3], Luciano García-Bañuelos[3], and Marcello La Rosa[1]

[1]Queensland University of Technology, Australia
{abel.armascervantes,m.larosa}@qut.edu.au
[2]Data61, CSIRO, Australia
nick.vanbeest@data61.csiro.au
[3]University of Tartu, Estonia
{marlon.dumas,luciano.garcia}@ut.ee

Abstract. This paper presents the integration of three behavior-based comparison operations between logs and/or models into the Apromore process model repository. Each of these operations takes as input a pair of process artifacts (two event logs, two models, or one log and one model) and describes their behavioral differences by means of natural language statements. The generated difference diagnosis has a range of applications of interest to both practitioners and researchers. For example, the difference diagnosis can offer guidance for reconciling discrepancies between business process variants (*model2model* comparison); can be used to pinpoint and explain differences between actual and expected behavior for conformance checking purposes (*model2log* comparison); or can explain dissimilarities between normal and deviant executions of a process (*log2log* comparison).

Keywords: Apromore, behavioral comparison, deviance mining, conformance checking, consolidation of variants

1 Overview

The behavioral comparison of process models and event logs is a recurrent primitive in business process analysis. Conformance checking, deviance mining and (behavior-based) process model comparison can be seen as specific instances of this general primitive. In particular, conformance checking aims at finding if the observed behavior captured in an event log complies with the behavior specified in the model; deviance mining compares the behavior captured in event logs to explain why a business process deviates from its normal or expected behavior; finally, (behavior-based) process model comparison aims at explaining the behavioral differences between process models that may correspond to variants of the same business process.

A critical feature of behavioral comparison operations is the interpretability of the identified differences between logs, models or between a model and a log. In this regard, a set of techniques for conformance checking, deviance mining and process model

comparison have been proposed in [1, 2], [3] and [4, 5], respectively. These techniques use event structures [6], a well-known model of concurrency, as unified behavioral abstractions for both logs and models [7]. This comparison approach is *independent of the input-format*, i.e. process models, event logs or both. The differences are reported to the user in the same way regardless of the input type, using a set of sentences in natural language, without requiring in-depth knowledge of formal modeling languages like Petri nets. Consequently, our tool is suitable for use by business analysts and, as opposed to other approaches, it does not require process modeling or mining experts to interpret the results. Figure 1 depicts an overview of the implemented operations.

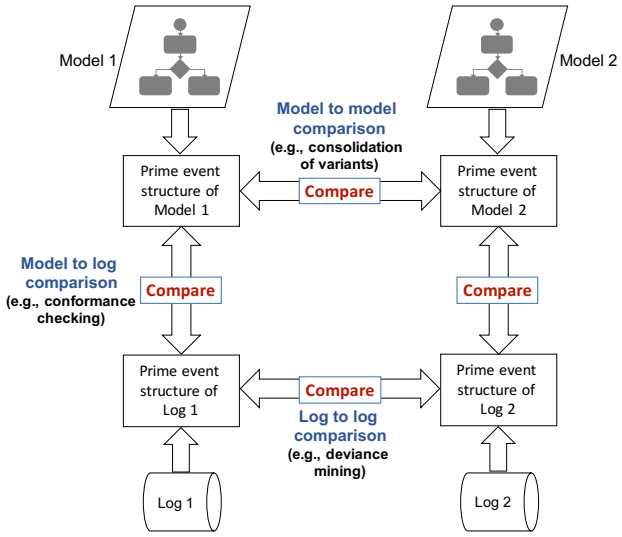

Fig. 1: Overview of the implemented comparison operations.

This paper presents the integration of the techniques proposed in [1, 2], [3] and [4, 5] into the Apromore process model repository [2]. Apromore is an open-source and extensible online repository of state-of-the-art capabilities for managing large process model collections. The operations for conformance checking, deviance mining and (behavior-based) process model comparison complement the wide range of existing capabilities, e.g., process model merging, simulation, restructuring, querying and similarity search, provided by Apromore. The three techniques are wrapped into a pairwise *Compare* operation that will automatically execute the appropriate comparison operation depending on the type of selected artifacts as input, two models, one model and one log, two logs. The modeling languages supported are all those available in Apromore, namely BPMN, EPCs, Petri nets and YAWL. The log format can be XES or MXML.

[2] http://apromore.qut.edu.au

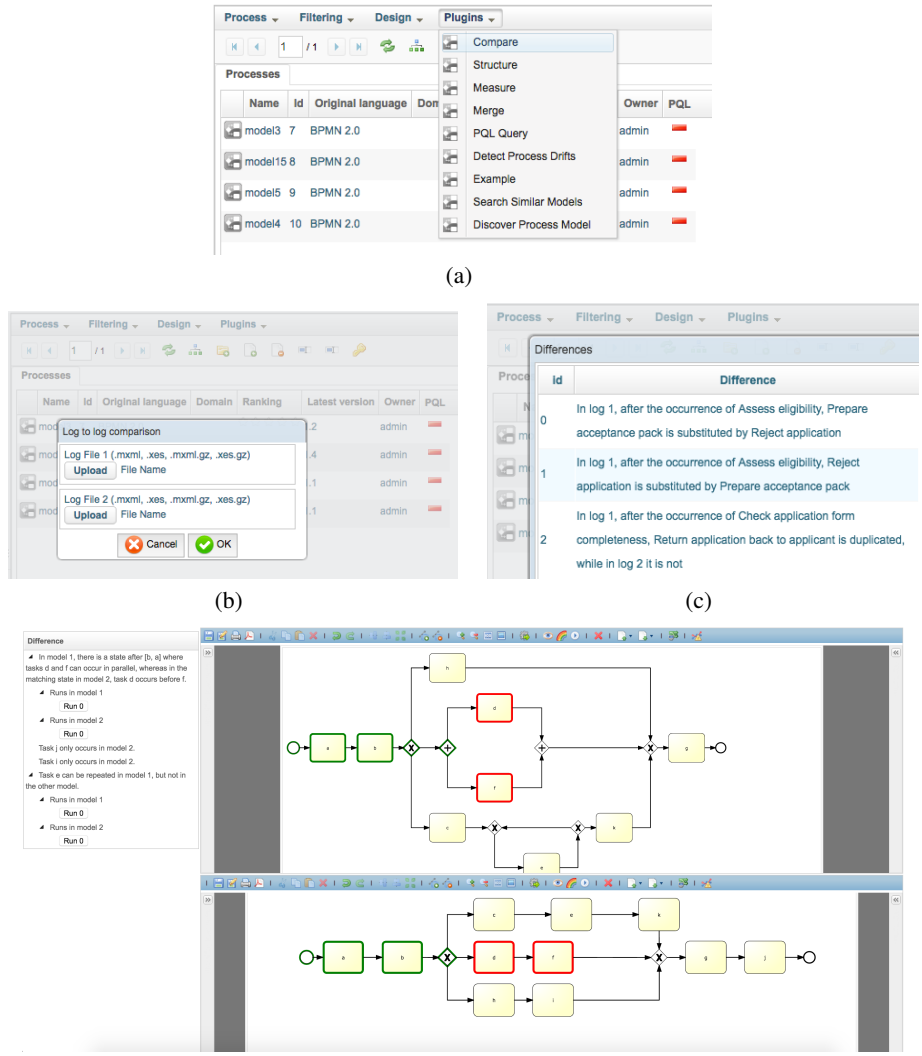

Fig. 2: Apromore Compare interface.

Given a pair of process artifacts, the steps for the three types of comparison are: 1. computation of event structures out of event logs and/or models, 2. comparison of event structures, and 3. verbalization of the identified differences. In the current implementation the type of event structures used is *prime event structures* [6], the comparison is performed using a so called *partial synchronized product* [4], and the verbalization of the differences is aligned with the corresponding papers [1, 3–5].

Figure 2 shows the different user interfaces in Apromore for the comparison of two logs and two models. Figure 2(a) shows the menu in Apromore where the *Compare*

operation is located, Figure 2(b) depicts the input dialog to upload a pair of event logs and Figure 2(c) shows the popup window displaying the statements in natural language explaining the identified differences. Finally, Figure 2(d) depicts the representation of the differences resulting from the comparison of two models. In this case, the differences are represented in two ways: 1. as statements in natural language (left) and 2. as overlaid graphics on the process models (right).

2 Significance and Maturity

Our toolchain exhibits a novel approach for identifying root causes of deviant process executions (via event log comparison), identifying differences between process executions and normative process specifications (conformance checking of an event log with a process model), and identifying differences between different versions or specifications of process models (i.e. comparing two different business process models).

The approach can be applied both in intra-organizational and cross-organizational settings. For instance, different process variants and executions in public organizations can be identified and analyzed to obtain a set of generic models (e.g. including best practices) and a set of additional organization-specific features. In addition, process executions in different organizational branches can be analyzed to identify causes for performance differences.

This approach provides for the first time a tool that abstracts from the representation (i.e. process model or event log) and is capable of reporting a complete set of differences via natural language statements. The set of produced statements are compact and interpretable to allow for a clear overview of differences between process models and/or logs of process executions. The reporting of differences is specifically intended to inform business analysts, and comparison of complex models and logs is, as such, no longer limited to technical users only. Consequently, users who are directly involved in the business process under investigation are now able to interpret the results of the comparison. Furthermore, the results provided are complete (i.e. all behavioral differences are identified), more compact and precise than existing approaches and provide, therefore, a much better assessment of differences between process models and logs.

The Apromore features used in this approach have been evaluated extensively with respect to accuracy, scalability and advantages over existing approaches. The evaluations comprised large collections of both artificial and real-life process models and event logs. The qualitative evaluation showed that the presented toolchain produces a more compact and much more understandable diagnosis than existing techniques. Furthermore, the tool exposes differences that are difficult or impossible to identify otherwise. The quantitative evaluation involved over 700 real-life process models and showed that the proposed approach has reasonable execution times (within seconds). Even in extreme cases with a high number of differences between the process model and the event log (with the event log containing more than 8,000 event occurrences, considering distinct traces only), the execution time is still within a few minutes. The detailed results of these evaluations are reported in [3] (log2log), [1] (model2log) and [4, 5] (model2model).

Apromore features an OSGi plugin framework to support dynamic enabling/disabling of plugin bundles and multiple bundle versions. These capabilities include presentation capabilities with respect to process model restructuring, filtering of models based on process-related aspects, searching and querying for specific process patterns, advanced design of process models, including configuring and merging if existing models, and evaluation capabilities to assess the quality and correctness of models, as well as simulation and conformance checking techniques for benchmarking. Furthermore, Apromore provides full import and export functionality to a large variety of business process modeling languages and log formats, such as BPMN, XPDL, EPML, ARIS, YAWL, PNML, XES and MXML.

Apromore is the result of over six years of ongoing development and is currently in version 3.4. The platform is implemented via a service-oriented architecture and deployed as a Software as a Service. The technologies used in Apromore combine Spring as the Java development framework, Maven as the dependency manager, OSGi as the plugin architecture, EclipseVirgo as the OSGi-based application server, and ZK as the AJAX front end. The chosen technologies allow Apromore to be an extensible framework, where new plugins can be easily added to an ecosystem of advanced capabilities for managing process model collections.

3 Screencast

A screencast of Apromore's compare feature can be found at `http://goo.gl/JB1EDv`. The public release of Apromore is available at `http://apromore.qut.edu.au` and its source code can be downloaded under the GNU LGPL license version 3.0 from `https://github.com/apromore/ApromoreCode`.

References

1. García-Bañuelos, L., van Beest, N.R.T.P., Dumas, M., La Rosa, M.: Complete and interpretable conformance checking of business processes. QUT ePrints, Technical report #91552, http://eprints.qut.edu.au/91552/ (December 2015)
2. García-Bañuelos, L., van Beest, N.R.T.P., Dumas, M., La Rosa, M.: Business process conformance checking based on event structures. In: Proc. of NWPT'2015, Reykjavik University (2015) 3 pages.
3. van Beest, N.R.T.P., Dumas, M., García-Bañuelos, L., La Rosa, M.: Log delta analysis: Interpretable differencing of business process event logs. In: Proc. of BPM 2015, Springer (2015) 386–405
4. Armas-Cervantes, A., Baldan, P., Dumas, M., García-Bañuelos, L.: Diagnosing behavioral differences between business process models: An approach based on event structures. Information Systems **56** (2016) 304–325
5. Armas, A., Baldan, P., Dumas, M., García-Bañuelos, L.: Behavioral comparison of process models based on canonically reduced event structures. In: BPM 2014, Springer (2014) 267–282
6. Nielsen, M., Plotkin, G.D., Winskel, G.: Petri Nets, Event Structures and Domains, Part I. TCS **13** (1981) 85–108
7. Dumas, M., García-Bañuelos, L. In: Process Mining Reloaded: Event Structures as a Unified Representation of Process Models and Event Logs. Springer (2015) 33–48