

Managing Massive Business Process Models and Instances with Process Space

Shuhao Wang, Cheng Lv, Lijie Wen, and Jianmin Wang

School of Software, Tsinghua University, Beijing 100084, P.R. China
shudiwsh2009@gmail.com, lvcheng1031@qq.com,
wenlj@tsinghua.edu.cn, jimwang@tsinghua.edu.cn

Abstract. BPM techniques are becoming more widely used, and there are more and more business process models and instances emerging. In this demonstration, we show how to manage large scale of process models and instances with Process Space. Creating, importing, storing, indexing and querying of models and instances will be exhibited. Since online tools for managing massive process instances are very rare, we focus on showing our useful tool of exploring process instances.

1 Introduction

With the technology of business process management being more widely used, there are more and more business process models and instances accumulated in enterprises. For example, there are more than 8,000 models in CMCC, a Chinese state-owned telecommunication company. On the other hand, in recent years, process data (especially event log) is increasing sharply and becoming a typical type of big data. Behavioral analysis on big process data is in urgent need. For example, 90,000 construction machineries in Sany (officially Sany Heavy Industry Co., Ltd.), a Chinese multinational heavy machinery manufacturing company, generated more than 60 billion working status records in 2012. The size of Wal-Mart's RFID data recorded every three days is equivalent to the entire collections of The Library of Congress, and even common RFID applications generate log data of more than 1 Gigabyte every day.

Based on these models and instances, enterprises can get a clear view of their processes' running states, which plays an important role in making decisions. Therefore, managing a large number of models and instances efficiently is challenging and can bring huge values for enterprises.

Our contributions can be summarized as follows. Firstly, an online process model and instance management tool is fully designed and implemented for big process data, including modeling, importing, storing, analysis and querying. Secondly, it's an open platform which supports user-defined modules and functions.

The remainder of this paper is structured as follows. In Section 2 we study related work, before we introduce the implementation of our work in Section 3. The management of process instances is presented in Section 4. In Section 5, we show our demonstration and conclude the paper.

2 Related Work

BeehiveZ [2] BeehiveZ¹ is a business process model management system developed by Tsinghua University. It focuses on the kernel algorithms for model query, index, generation, simulation, similarity measure and the evaluation of process mining algorithms, etc. Four types of business process model indexes and queries mentioned in [7], including: (1) exact query based on structure, (2) similarity query based on structure, (3) exact query based on behavior, (4) similarity query based on behavior, are all supported by BeehiveZ.

Nearly all the functions in BeehiveZ have been integrated and extended into our new online tool Process Space.

Oryx Oryx² is a web-based editor for modeling business processes in various languages like BPMN, EPC and Petri net. It is an open platform for developments regarding process modeling. Oryx is a project of the University of Potsdam. Some source codes of Oryx are modified and imported into Process Space.

Disco Disco³ is a stand-alone applications designed and developed by Fluxicon for process mining and analysis on event logs. In the design of process execution analysis module of Process Space, we get a lot of inspiration from chart display methods in Disco. Compared to Disco, Process Space is an open web application which can handle large scale event logs and supporting third-party plugins.

Business process model query There have already been several research prototypes on business process model query. BPMN-Q [3] is a graph-based query language. WISE [5] is a workflow information search engine, where workflow models are represented hierarchically. VisTrails [4] allows users to query workflows by example. BP-QL [1] is based on an abstraction of the BPEL standard for distributed environment and supports query by example. Yan [8] uses feature-based similarity estimation to improve the efficiency of similarity search.

3 Implementation of Process Space

Process Space is a Browser/Server application implemented in Java. As shown in Figure 1, it includes four major modules, which are process model analyzer, process monitor, process instance analyzer and process data repository. For space limitation, we omit the detail description.

¹ <https://code.google.com/p/beehivez/>

² <https://code.google.com/p/oryx-editor/>

³ <http://fluxicon.com/disco/>

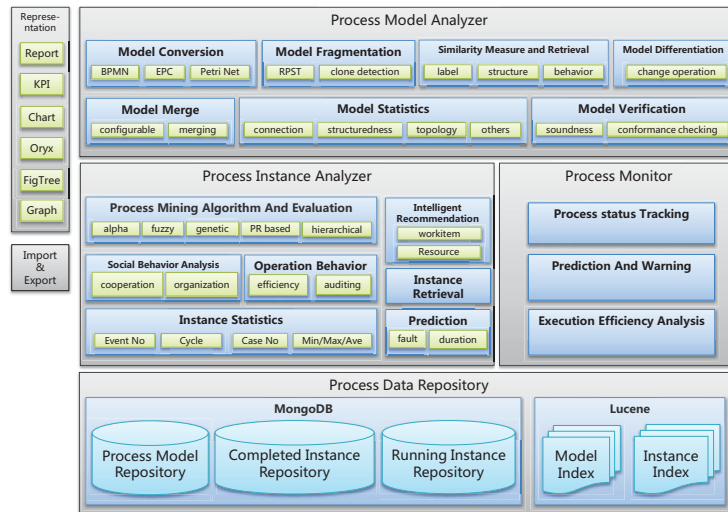


Fig. 1. System architecture of Process Space

All the functions in process model analyzer are borrowed from BeehiveZ and are redesigned accordingly. BeehiveZ uses Derby, a memory database, while Process Space chooses MongoDB⁴ which is an open-source document database. The demo of BeehiveZ was shown in [2]. Therefore, our demonstration will focus on the management of process instances, including importing, view, analysis and statistics. Details will be presented in Section 4.

In Process Space, two different repositories are provided to manage massive process models and instances separately, both including creating, storing, viewing, and querying. We use MongoDB as data storage layer, Spring Data for MongoDB⁵ as data access layer, Spring Framework⁶ as business logic layer and JQuery⁷ as web presentation layer. An important design idea in Process Space is its extensibility. Importing new modules into Process Space is facilitating. Users are encouraged to upload new algorithms of mining, indexing and querying implemented in Java under specific interfaces. We intend to build a tool with an open service concept and attract more people to contribute in BPM field.

4 Instance Management in Process Space

Before managing process instances, event logs must be imported and resolved. Several methods are implemented in order to import formatted text files, complex CSV and MS Excel files as well as ProM's MXML format [6]. After analysis

⁴ <http://www.mongodb.org/>

⁵ <http://projects.spring.io/spring-data-mongodb/>

⁶ <http://projects.spring.io/spring-framework/>

⁷ <http://jquery.com/>

on event log, instance data is reconstructed and saved into database. We can easily view and move event logs with a friendly interface.

Querying instances Common instance querying examples are mainly based on length of activities, time consuming of activities, activity categories and adjacent activities. These four types of queries are all supported in Process Space.

- *Length of activities* refers to the number of activities in a process instance. When a user wants to query instances whose length is equal to, or larger than, or between some given values, this type of query would be used.
- *Time consuming of activities* refers to the cost of time from the execution of first activity to last activity in an instance. This is helpful in situations like querying instances completed within a given period of time.
- *Activity category* index is designed to solve problems when a user is interesting in instances containing specific types of activities.
- *Adjacent activities* refers to two activities whose execution is in sequential order in a process instance. This type of query is utilized to search instances with a given sequential structure.

In order to meet the requirement of user-defined instance index, we have designed third-party interfaces. Users can implement their own indexing algorithms and upload them onto Process Space. Once uploaded algorithms have been checked and granted, they will come into operation in the system.

Process Space uses Lucene⁸, a Java-based indexing and search technology developed by Apache, as indexing engine. Moreover, we use mongo-lucene⁹, a MongoDB-backed lucene directory for a scalable real-time search, to save indexing data into MongoDB and manage them with Lucene at the same time.

We use large scale of auto-generated models and instances to test the performance of index module. With the scale of models and instances enlarging, the time of inserting, indexing and querying keeps increasing and maintains a linear relationship with the scale, as shown in Figure 2. Experimental results show that the index module has an excellent performance.

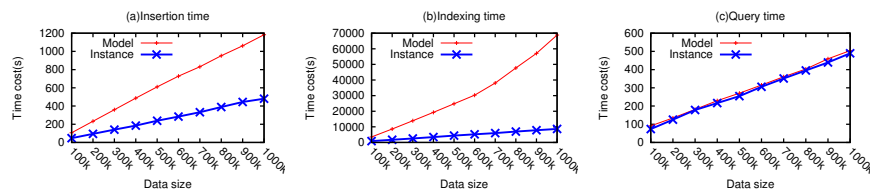


Fig. 2. Experimental results on performance of index module

⁸ <http://lucene.apache.org/>

⁹ <https://github.com/rstiller/mongo-lucene>

Process execution analysis and visualization display Like Disco, Process Space implements three major types of process execution analysis, instance overview, activity statistics and resource statistics. Line charts and pie charts are the two types of methods to represent the analysis results.

5 Demonstration

What will be shown in the demo? We will demonstrate the usage of Process Space and focus on the management of process instances, including importing event log, instance overview, indexing and querying, and analysis. We will use industrial event logs from different enterprises and massive artificial event logs as examples. Through the demonstration on the example event logs, we can see the management of instances is very friendly and efficient and the query performance is good with use of indexes.

Significance in BPM area. This demonstration shows a process model and instance management tool called Process Space and how to manage process instances efficiently and effectively with it. It is helpful for efficient management and analysis of business process data.

Acknowledgements. The work is supported by the National Science Foundation of China (No.61472207 & No.61003099), the Ministry of Education & China Mobile Research Foundation (MCM20123011) and the special fund for innovation of Shandong, China No. 2013CXC30001.

References

1. Catriel Beeri, Anat Eyal, Simon Kamenkovich, and Tova Milo. Querying business processes. In *Proceedings of the 32nd international conference on Very large data bases*, pages 343–354. VLDB Endowment, 2006.
2. Tao Jin, Jianmin Wang, and Lijie Wen. Efficiently querying business process models with beehivez. In *BPM (Demos)*, 2011.
3. Sherif Sakr and Ahmed Awad. A framework for querying graph-based business process models. In *Proceedings of the 19th international conference on World wide web*, pages 1297–1300. ACM, 2010.
4. Carlos E Scheidegger, Huy T Vo, David Koop, and et al. Querying and re-using workflows with vstrails. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1251–1254. ACM, 2008.
5. Qihong Shao, Peng Sun, and Yi Chen. Wise: A workflow information search engine. In *Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on*, pages 1491–1494. IEEE, 2009.
6. Boudewijn F van Dongen, Ana Karla A de Medeiros, HMW Verbeek, and et al. The prom framework: A new era in process mining tool support. In *Applications and Theory of Petri Nets 2005*, pages 444–454. Springer, 2005.
7. Jianmin Wang, Tao Jin, Raymond K Wong, and Lijie Wen. Querying business process model repositories. *World Wide Web*, pages 1–28, 2013.
8. Zhiqiang Yan, Remco Dijkman, and Paul Grefen. Fast business process similarity search with feature-based similarity estimation. In *On the Move to Meaningful Internet Systems: OTM 2010*, pages 60–77. Springer, 2010.