

# Business Process Mining for Service Oriented Architecture



Amna Al Shereiqli and Youcef Baghdadi

**Abstract** Service Oriented Architecture (SOA) is an architectural style that uses loosely coupled services, which have separate concerns, as the primary constituents to build software solutions as compositions of services. It emphasizes loosely coupling and interoperability of services, required for agility and flexibility of business processes (BPs), hence allowing current BPs as well as future BPs to be integrated. To design such SOA-compliant software that supports BPs, the service identification is a critical phase. Indeed, the form under which the service is located may impact the effectiveness of the SOA. More particularly, the granularity of the entire services is very crucial in attaining the flexibility. The main objective of this research is to mine BPs for SOA by introducing a new service identification approach. The approach first uses a BP mining discovery technique to discover the BP model. Then, it clusters the activities into tasks, and finally organizes tasks into an initial set of services. And checks the compliance of the resulted services against the service orientation principles. The proposed approach is validated by a case study. The approach helps large enterprises to mine their BPs, develop them, and identify services. Finally, we recommend an automation of the proposed service identification approach to making it more effective and useful BPs.

**Keywords** Business process · Business process mining · Service oriented architecture · Web service · Service orientation · Service identification

## 1 Introduction

An enterprise usually has a large and complex information system (IS) that consists of a huge number of BPs. Some are well modeled and others are not. Enterprises need to continuously adapt their BPs to cope up with business requirement changes,

---

A. Al Shereiqli · Y. Baghdadi (✉)  
Department of Computer Science, Sultan Qaboos University, Muscat, Oman  
e-mail: [ybaghdadi@squ.edu.om](mailto:ybaghdadi@squ.edu.om)

A. Al Shereiqli  
e-mail: [m084074@student.squ.edu.om](mailto:m084074@student.squ.edu.om)

© Springer Nature Switzerland AG 2020  
Y. Baghdadi et al. (eds.), *ICT for an Inclusive World*,  
Lecture Notes in Information Systems and Organisation 35,  
[https://doi.org/10.1007/978-3-030-34269-2\\_1](https://doi.org/10.1007/978-3-030-34269-2_1)

which requires more agility and flexibility of their BPs [1]. This agility is difficult to achieve because: (i) BP modeling considers that all the instances of the same BP run in the same way, and (ii) the tightly coupled components of the architecture of the legacy IS supporting the BP do not enable the loose coupling and interoperability required for flexibility and agility [2].

It is proven that SOA enables the agility of the enterprise through more flexibility of its BPs.

However, moving to SOA requires re-engineering the BPs in order to transform their activities into reusable, loosely coupled services [3].

One of the most used techniques toward reengineer running BPs is known as BP mining, which allows the extraction of the knowledge about BPs from the event logs of existing IS [4]. It is concerned with the discovery, monitoring, and improvement of the BPs by obtaining the wider knowledge from the event logs that is generated from existing ISs.

However, how to extract the activities from running BPs that are not modeled and present them as services is still an issue?

In this research, we propose a novel approach that will assist enterprises to mine their BPs in order to reengineer them in new technologies such as SOA. It is based on service identification methods. the new suggested service identification approach that mines BPs, clustering their activities into tasks, initialing a set of services, listing the services, and verifying the service candidates against the service orientation principles.

The remainder of this chapter is organized as follows. Section 2 presents the concepts related to this research. Section 3 presents some related works. Section 4 details the research methodology and the proposed approach. Section 5 applies the approach to a case study. Finally, a conclusion section provides some future work.

## 2 Background

In this chapter, key concepts related to the research are reviewed, including BP, BP mining, BP reengineering, SOA, and service identification techniques.

### *Business Process Management*

Business Process Management (BPM) concerns with seven phases as proposed by Mathiesen et al. [5]. These are: identification, modeling (as-is), analysis, improvement (to-be), implementation, execution (to-do), and monitoring/controlling.

### *Business Process Modeling*

BP modeling is defined as description of the activities in a diagrammatic manner. It shows the events and the variables from one place to another [6]. It is the most important step in BPM.

There are many techniques that are used by BPM such as UML diagrams, Business Process Modeling Notation (BPMN), data flow diagrams, flowchart technique, role activity diagrams, role interaction diagrams, Gantt charts, integrated definition for function modeling, colored petri-nets, object oriented methods, workflow technique, and simulation model.

### *Business Process Mining*

Business process mining is an extraction of the knowledge about the running BPs from the event logs of the running IS. The goal of BP mining is to discovering, monitoring, and improving real BPs by extracting them from event logs of running IS [4]. Indeed, more and more events are being recorded, thus, providing detailed information about the history of the running BPs, and there is a need to improve and support BPs in the competitive and rapidly changing environment.

The main components of BP mining are event logs and process model. An event log is defined as “a collection of events used as input for process mining. Events do not need to be stored in a separate log file” [7]. Event logs comprise information about the start and completion of process steps together with related context data [8]. Every log entry holds an identifier of the case, an activity name, a timestamp, a user identifier, a booking code and a notification [9]. It is used as starting point in mining [10]. The process model is generated from the event logs [11].

### *Tools for BP Mining*

BP mining are performed by various software tools. There are about fifteen commercial process mining tools. This research uses Disco because it is free for academia (by using academic credentials). In addition, it provides an automatic process discovery modeling as the event logs are imported into the system. This allows a simply filtering and examining the attributes in parallel with the exposed fuzzy model. Furthermore, it provides a filtering mechanism that is more transparent and obviously introduced in comparison to the other tools. Moreover, it imports log size capacity up to five million events. It is supported by standalone desktop version platform and it is easy to use. It enables animation of processes, which makes it easy to discover bottlenecks on the flow of processes and the animation of processes can be stored as video format. This approach uses a Disco to mine BP, as it introduces a large number of cases [12]. Disco uses as input an event log in CSV, MXML, XLS, XES or FXL format and produces as-is BP model as output. In addition, it produces the frequency of each activity and the frequency of each edge in the BP model. Disco imports the event log which fulfills the minimum requirement for analysis, including case ID, activity, and timestamp.

### *Business Process Reengineering*

Business Process Reengineering (BPR) was defined early in 1997 by Blyth as “the fundamental re-thinking and radical redesign of BP to achieve dramatic improvement in critical, contemporary measures of performance, such as cost, quality, service, and speed” [13]. The goal of BPR is to radically improve the BPs. It needs the organizational restructuring with the aid of simplification and standardization and IT. Hammer [14] defined seven principles for BPR.

### *Service Oriented Architecture*

SOA is defined as “an architectural paradigm for developing and integrating heterogeneous ISs with strict message-driven communication paradigm” [15]. It provides a set of guidelines, principles, and techniques in which BPs, information, and enterprise assets can be effectively re-organized and re-deployed to support and enable strategic plans and productivity levels that are required by competitive business environments [16].

The improvement of flexibility, interoperability, and abstraction level of software components are the major aim of SOA. Loose coupling and discoverability are the two key principles of SOA [17].

## **3 Related Work**

Several service identification methods (SIMs) are proposed. These methods use different inputs, strategies, and techniques, and produce different types of output. The artifacts used as input include BP, application domain, legacy system, mix, data, feature, or use cases. They generally produce one of the six types of services: business process service, data service, composite service, IT service, web service, and partner service.

In this research, we are interested in the SIMs that use the BP as input artifact. Wang et al. proposed a method to identify the normalized service from the BP model. This method has three stages: (1) identify some normalized services from BPs and design the containment relationship between service and business activities, (2) design service ports in the business aspect, and (3) design the component set that constitutes a service and message mapping between service ports and component interfaces in the technical aspect. It is based on BP decomposition and algorithm techniques. However, it has not been validated by a case study [18]. Inaganti and Behara [19] concentrated on recognizable proof of big business level services and built up the choreography of the services through BP. They use BP decomposition strategy and some guidelines techniques to identify services. Their technique for service identification is restricted to the service identification procedure. Although the proposed method is clear for service identification, the measures for identification of business activities are not introduced. Besides, it is not approved with any useful case study and does not propose standard displaying documentations for process modeling and SOA. Amsden [20] focused on the description of service elements

that should be located in service form of the specification phase. The method uses BP decomposition strategy and analysis technique to determine services. It has been validated by real example. Mani et al. [21] presented a method that focuses on the design specification of the user interface. It captures the unified user interface design specification from the business procedure and the information display. It uses the BP decomposition and user interface strategies to identify services. A shopping cart of Amazon case study has been used to validate this method. Jamshidi et al. [22] proposed a method that consists of four steps: (1) modeling of BP, (2) identification or the location of service model elements, (3) grouping or the categorization of services, (and 4) proper documentation of service. It is uses BP decomposition and business entity strategies and algorithm technique to determine services. This method is evaluated based on its use, users' form of analysis, and strength or power over the existing methods. Dwivedi and Kulkarni [23] introduced a method that uses some heuristics for identifying service candidates along with model driven development. It is uses BP model (with UML) as input. As techniques, it is uses BP decomposition strategy and algorithm technique to identify a list of services. It has been validated by a real example. Bianchini et al. [24] presented a method that focuses on identifying services from a collaborative BP. It is divided into four phases: (1) semantic form of the process annotation, (2) proper identification of candidate services, (3) evaluation and analysis of service strong cohesion/coupling, and (4) refinement of the entire process decomposition. It uses BP decomposition strategy and an ontology technique to determine candidate services. It has been validated by a case study. Yousef et al. [25] proposed a method called BPAOntoSOA that uses ontology technique and BP decomposition strategy to identify services. It is based on BP understanding and analysis, taking into account functional and non-functional requirements. It is uses clustering algorithm to improve the correctness of mapping business functions to services in the resulting service oriented model. It is validated by a case study of healthcare sector. Azevedo et al. [26] introduced a method that applies heuristics to characterize services from the semantic investigation of process components and from a syntactic examination of process models. It consists of three stages: (1) selection of activities, (2) identification and classification of candidate services, and (3) consolidation of candidate services. Kim and Doh [27] proposed a method that uses the concept of graph clustering. It considers activities as service identification and bunches activities with high cooperation to maximize the cohesion of local tasks. It uses BP decomposition strategy and algorithm technique to determine services. It is validated by case study. Ren and Wang [28] proposed a method that uses a BP as input and produces a service model as output. It uses a clustering algorithm. It is validated by a case study. Nikravesh et al. [29] presented a method called 2PSIM. It uses as a BP as input to generate a service model. It identifies services by applying graphic form of partitioning algorithms. Kazemi et al. [30] exhibited an automated technique to generate business services by using a BP decomposition strategy. Jamshidi et al. [31] introduced an automated method that uses the best practices and the core principles of the form of model driven software development. It uses as starting input an enterprise business model and produces a service model. It determines services by implementing a meta-heuristic algorithm. It was evaluated against enterprise scale

study. Soltani and Benslimane [32] developed a method called an Automatic Model-Driven Service Identification (AMSI) that uses a high level of BP model as input to specify service model artifacts. It identifies services by applying multi-objective evolutionary. Birkmeier et al. [33] proposed a method to determine services from BP models. The general impacts of this approach are to illustrate how the form of the business architecture can be utilized to drive the organization of the IS architecture and to make a proper confirmation of the alignment with business needs. El Amine and Benslimane [34] exhibited a type of a mechanized approach to generate business services by actualizing a few outline measurements based on process decomposition strategy. It takes a BP as input information and produces a list of business services by applying a multi-objective combinatorial particle swarm optimization algorithm. Bianchini et al. [35] acquainted a Process with Service approach (P2S). It is a computer-aided methodology to permit the recognizable proof of services that consolidates a collaborative BP. Mohamed et al. [36] proposed an automated method. It identifies services from BPs. It applies for bunching a hybrid particle swarm optimization algorithm and several design metrics for delivering reusable services with appropriate granularity and a satisfactory level of coupling and cohesion. Leopold et al. [37] proposed a mechanized service identification technique that distinguishes a list of positioned service candidates from BP models. Specifically, it creates a list of atomic services, composite services, and inheritance hierarchy services.

All of the above-mentioned SIMs use as input a business process model. Since, it plays an important part wrapping the phase's form of the BP design and system implementation. However, in most cases, the BP model could not introduce all possible essential tasks and interrelated control flows in the design phase. Moreover, the BP model is not sufficiently adaptive to respond at a runtime to the dynamic environments [38].

## 4 The Proposed Approach

This section presents the proposed approach along with the used research methodology. The research methodology mainly consists of two steps: (1) study the BP mining approaches, and (2) selection of a suitable SIM. The proposed approach is based on the service identification phase of a service oriented development.

### 4.1 The Research Methodology

This section details the steps of the research methodology that is used to design the proposed approach. As shown in Fig. 1, the methodology consists mainly in surveying the BP mining techniques and the existing SIMs to select a suitable SIM,

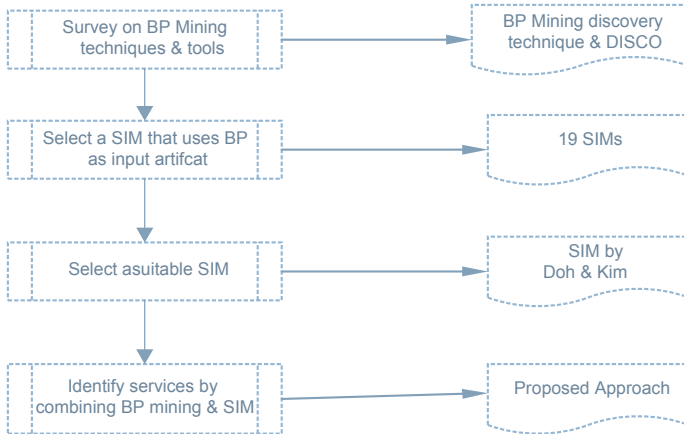


Fig. 1 Steps and output of the research methodology

then combining the BP mining techniques with the SIM method to identify services from the running BPs.

*Step 1: Study the BP mining*

BP mining focuses on the analysis of the BP by using an event log. It aims at discovering, monitoring, and improving the BPs by extracting knowledge from the event logs that are generated automatically in the IS [4]. It consists of three types of techniques: discovery, conformance checking, and enhancement.

However, the proposed approach is limited to the discovery technique. A discovery technique uses an event log as input and produces a BP model without using any prior information. Figure 2 depicts the BP mining discovery technique process. It starts

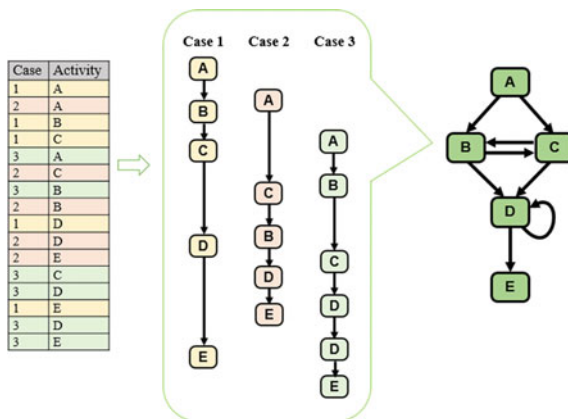
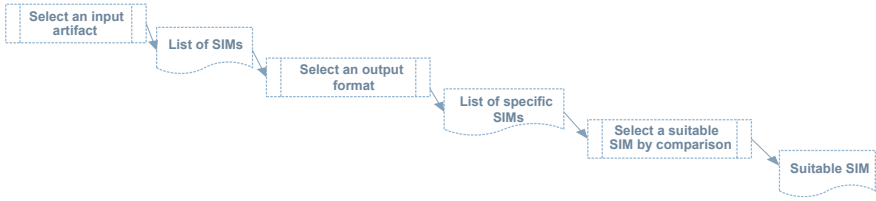


Fig. 2 The process of BP mining discovery technique



**Fig. 3** Methodology of selected SIMs

from the event log, then extract each case or process instance, and finally combine them to produce the BP model.

*Step 2: Selecting a suitable SIM*

This step consists mainly in searching for existing SIMs that use as input a BP to select a suitable SIM by comparing the existing, as shown in Fig. 3.

SIMs are divided into three scenarios: top down, bottom up, and hybrid. This research focuses on the top-down scenario and starts from the input-output matrix introduced by Gu and Lago [39] to select SIMs. According to this research, there are about thirty SIMs based on the top-down scenario. They use various inputs artifacts such as BP, application domain, legacy system, data, feature model, use case, and/or a mix of them. These SIMs use strategies such as BP decomposition, business functions, business entity, ownership and responsibility, goal driven, component based, existing supply, front office application usage analysis, infrastructure, non-functional requirement, and user interface. They are based on techniques used such as algorithm, guideline, analysis, ontology, pattern, and information manipulation. Finally, these methods produce output in various formats such as informal service specification, service model, formal service specification, service implementation, or a list of services.

This research focuses only on the SIMs that use as input a BP. Table 1 depicts an example of comparison between SIMs that use BP as input: authors, publication year, type of input, strategy used to understand the type of input, type of output, technique, and the validation.

**Table 1** SIMs that use a BP as input to get a service as output

Author	Year	Type of input	Strategy	Type of output	Technique	Validation
Mani et al.	2008	Business process	Business process decomposition and user interface	Service implementation	Algorithm	Case study
Kim & Doh	2009	Business process	Business process decomposition	Service implementation	Algorithm	Evaluated



**Table 2** Comparison between Mani et al. method and Kim and Doh method

Service identification method criteria	Mani et al. [21]	Kim and Doh [27]
Scenario	Top-down	Top-down
Strategy	BP decomposition and user interface	BP decomposition
Input	BP	BP
Input format	BP model and data model	Model
Service classification scheme	Two service hierarchies: Service and human task	Three service hierarchies: Service, task, and activity
Covering of SOA design phases	Service identification, specification, and realization	Service identification and specification
Characteristics	WSDL description and case study	WSDL description and case study
Application of process models for service identification	Process model and data model are decomposed to generate user interface model than maps to service candidates	Process model divided into service candidates

This research compares between two SIMs that use a BP as input and produce services as output. It uses different criteria introduced by Kloose et al. [40] and Kim and Doh [27] as shown in Table 2. These criteria are: scenario, strategy, input, input format, service classification scheme, covering of SOA design phases, characteristics, and application of process models for service identification.

Both methods are analyzed in the top-down scenario that starts with the high-level artifacts as input to identify services. While, both of them start with BP as input, the format of an input differs in each method. Mani et al. method starts with the BP model and data model, but Kim and Doh method starts with the BP model only. In addition, both methods produce as output service implementation written in Web Service Description Language (WSDL). The other comparison criteria are approximately similar. Thus, the proposed approach will be based on Kim and Doh method, because BP model is produced by only a BP only mining software tool.

The SIM introduced by Kim and Doh in 2009 has three activities: clustering of activities into tasks, organizing tasks into initial set of services, and describing the services.

## 4.2 The Proposed Approach Described

The proposed approach uses as input an event log and produces a list of services as output.

First, a BP mining discovery technique allows the extraction of the BP model from the event log of running IS. Next, the obtained BP model is used as an input in the clustering activity, whereby, each cluster is mapped into a task. These tasks are then organized into an initial set of services by allocating the activities to the corresponding task and defining operations for each task. Then, each task becomes a service and is inserted into its corresponding list. Finally, the resulted services are verified to comply with the principles SOA.

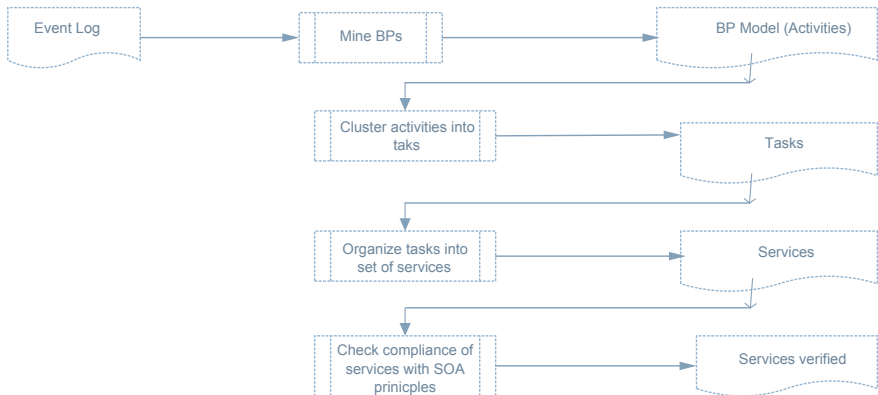
The approach consists of seven activities: collect an event log of the BP, convert an event log file into a suitable format that is understood by a BP mining software tool, use the BP mining discovery technique tool and produce a BP, cluster the activities of the BP model into tasks, organize tasks into initial a set of services, check the compliance of the resulting services with SOA principles. These activities are performed sequentially, as shown in Fig. 4.

*Step 1. Collecting an event log of the BP*

An event log contains information about events referring to an activity and a case of the running BP. Each event represents an activity name and is related to the particular case (case identifier). In addition, an event log introduces more information about events such as resource executing or initiating the activity, the timestamp of event and data elements recorded with the event [41].

Event logs can be automatically generated from different business systems such as ERP systems, CRM or Workflow Management systems. The two common formats of event logs are Mining eXtensible Markup Language (MXML) and eXtensible Event Stream (XES) [42].

Since, most legacy systems of the large enterprises are not documented or explicitly presented, they have very limited information about what is actually happening in their organization. In practice, there is often a significant gap between what is



**Fig. 4** The proposed approach illustrated

prescribed or supposed to happen, and what actually happens. The event log contains more valuable and embedded knowledge. It is used as a starting point of this approach, i.e. as an input artifact for BP mining.

*Step 2. Converting an event log file into a suitable format that is understood by a BP mining software tool*

Each software tool uses different formats of event logs. For instance, CSV, XLS, MXML, XES, and FXL formats are understood by Disco.

*Step 3: Using the BP mining discovery technique tool to produce a BP model*

In this approach, we use Disco as the BP mining discovery technique software tool. The following steps depict how to use it:

*Step 4: Clustering the activities into tasks*

The BP model generated by Disco will be the starting input in this step with the frequency of each activity and the frequency of each edge in the model. Each activity in the BP model maps into a task. It clusters tasks which have edge with high frequency into a new task through hierarchical clustering algorithm to minimize the coupling of tasks and to maximize the cohesion of tasks. Then, the following rules depict how to cluster tasks and how to calculate the frequency of new tasks:

Rule 1: When two adjacent tasks are merged, one of their incident edges having the highest frequency is selected first.

Rule 2: Once an edge is selected, the edge cannot be selected again in each repetition.

Rule 3: When tasks are merged, a new task is generated and its frequency is calculated.

Rule 4: When adjacent two tasks are combined into a new task, the frequency of new task is the maximum frequency of two tasks.

Rule 1 and Rule 2 are used to determine which tasks are merged. On the other hand, the model is reconstructed by adjusting edges based on Rule 3 and the frequencies of new tasks are assigned based on Rule 4.

The edges linked to two tasks, which are merged into a new cluster, are connected to a new task. Tasks are then repeatedly combined in the same manner.

The termination of the clustering of tasks is based on the preferred size of services and the reasonable number of services desired for business domain. Since, the number of services directly affects the performance and network overheads.

*Step 5. Organizing tasks into an initial set of services*

The objective of this step is to organize tasks into services, where the cohesion within the service is maximized and the coupling between services is minimized. Service specification is derived by organizing identified tasks into services. Graph clustering provides a good way for grouping the vertices in the graph according to their connections. A structure of tasks is generated and activities in the task are suggested to be included in a service. This method provides a potential design with different levels of inter-service coupling, where activities are organized into services and smaller services are successively integrated into bigger ones.

### *Step 6. Listing of services*

For an identified service, a set of attributes to describe and document the capability of each service is defined. The rich description of the capabilities can be passed to development teams who can use the information to help select the appropriate implementation technologies, hosts, deployment topologies. The list of services is easily derived from tasks and activities are provided by the BP model.

### *Step 7. Verifying service candidates against the service orientation principles*

The identified services need to be evaluated against service orientation principles, while performing service identification phase.

## **5 Case Study**

### **5.1 Presentation**

The Road Traffic Fine Management sector was formed due to a great urge for a professional interface in between the motoring public and municipalities who gives out traffic fines. An Italian Local Police Force presents an IS that handles Road Traffic Fine Management process.

The Road Traffic Fine Management process has eleven activities: create fine, send fine, insert fine notification, insert date appeal to the prefecture, appeal to judge, add a penalty, send for credit collection, send the appeal to the prefecture, receive appeal result from the prefecture, notify result appeal to an offender, and payment.

This research uses the event log that was recorded by the IS that manages the Road Traffic Fine Management process, as a case study. The event log contains about 150,370 cases and 561,470 events.

The Road Traffic Fine Management process requires implementation in SOA to be more efficient, effective, and flexible.

### **5.2 Application**

The event log has been collected from the internet, as a real event log of road traffic fine management process. It consists of 150,370 events. It contains all information of the road traffic fine management process for instance case ID, activity, complete timestamp, resource, total payment, notification type, expense, vehicle class. However, the needed fields are case ID, activity, and timestamp (Fig. 5).

The event log format has been converted to CSV (Comma-Spread Values) format to be acceptable.

In this step, Disco takes an event log with the various format such as CSV, XLS, MXML, XES, and FXL as input and produces a BP model as output. Moreover, it

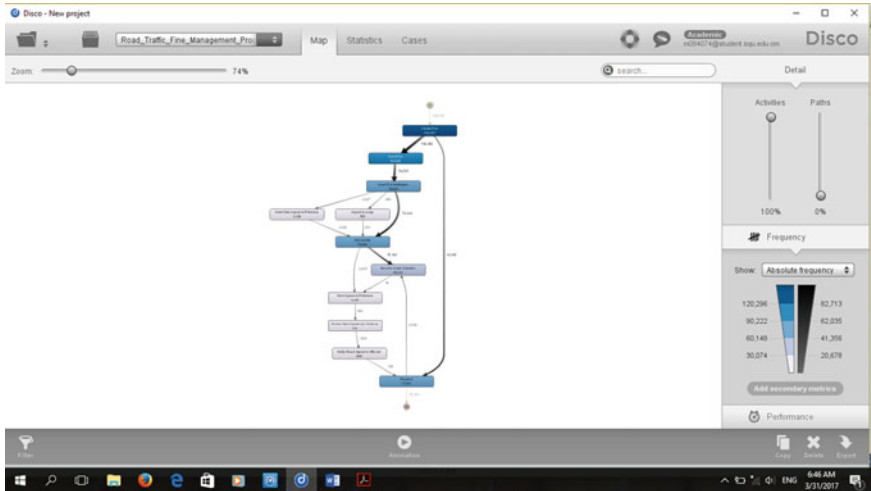


Fig. 5 A road traffic fine management process model

presents the frequency of each activity and edge in the model. The format of Road Traffic Fine Management process event log is CSV.

Firstly, all activities are converted into tasks. Secondly, the edge between two tasks with the highest frequency is selected. The new task is generated by merging those two tasks, where the frequency of this new task is the maximum frequency of two tasks. The connections to new task are updated.

The generated tasks are mapped into an initial set of services, as shown in Table 3.

Table 3 Set of services, tasks and activities

Service	Task	Activity
Service 1	Task 7	Create fine Receive fine Insert fine notification Add penalty Send for credit collection Payment Insert date appeal to prefecture Send appeal to prefecture
Service 2	Task 8	Receive result appeal from prefecture Notify result appeal to offender
Service 3	Appeal to judge	Appeal to judge

## 6 Conclusion

### 6.1 Summary

The research aimed at mining BPs in order to reengineer them in SOA. It concentrated on service identification based on running BPs in order to improve their flexibility with SOA.

The literature review has shown that the existing SIMs do not use event logs of the running BPs as input, though they contain valuable information of the running BPs.

The project has proposed a new top-down approach to identify service candidates by using the running BPs. The proposed approach uses as input an event log of a running BP to produce a list of services. It is a step-wise process that consists of: (1) collecting the event log of a running BP, (2) converting the event log file into a suitable format that is understood by a BP mining software tool, (3) using the BP mining discovery technique tool to produce a BP model, (4) clustering of activities into tasks, (5) organizing tasks into an initial set of services, (6) listing the services, and (7) verifying the resulting service against service orientation principles.

The proposed approach has been applied to a real BP, where the event log for Road Traffic Fine Management process was used as input to generate a set of services that would support the process in an SOA.

This research has theoretical and practical impacts. From a theoretical perspective, the research contributes to service technology by introducing new service identification approach. From a practical perspective, the proposed approach can assist the enterprises to mine their BPs in order to model them with services with respect to SOA and make them more flexible.

This approach still needs additional improvement, as it does not categorize the resulting services into various types, such as business services, IT services, data services, or composite services, based on the nature of the logic these services encapsulate and the manner the services are usually used within SOA.

## 6.2 Future Work

The proposed approach can be further:

- Refined in order to identify high quality services.
- Extended to cover all phases of SOA software development.
- Automated by using a tool in order to perform the service identification process faster and easier.

## References

1. Baghdadi, Y.: A business model for B2B integration through Web services. In IEEE International Conference on e-Commerce Technology, pp. 187–194, IEEE (2004)
2. Baghdadi, Y.: Modelling business process with services: towards agile enterprises. *Int. J. Bus. Inf. Syst.* **15**(4), 410–433 (2014)
3. Baghdadi, Y., Al-Bulushi, W.: A guidance process to modernize legacy applications for SOA. *Serv. Oriented Comput. Appl.* **9**(1):41–58 (2015)
4. Van Der Aalst, W.: Process mining: overview and opportunities. *ACM Trans. Manage. Inf. Syst. (TMIS)* **3**(2), 7 (2012)
5. Mathiesen, P., Watson, J., Bandara, W., Rosemann, M.: Applying social technology to business process lifecycle management. In: International Conference on Business Process Management, pp. 231–241. Springer, Heidelberg (2011, August)
6. Chapman, A.: Business Process Modeling. Accessed in March 2016. Retrieved from: <http://www.businessballs.com/business-process-modelling.htm>
7. Daniel, F., Barkaoui, K., Dustdar, S.: Business Process Management Workshops: BPM 2011 International ..., Part 1 (2012)
8. Mans, R.S., Schonenberg, M.H., Song, M., van der Aalst, W.M., Bakker, P.J.: Application of process mining in healthcare—a case study in a Dutch hospital. In: International Joint Conference on Biomedical Engineering Systems and Technologies, pp. 425–438. Springer, Heidelberg (2008, January)
9. Rubin, V.A., Mitsyuk, A.A., Lomazova, I.A., van der Aalst, W.M.: Process mining can be applied to software too! In: Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, p. 57 (2014, September)
10. Van Dongen, B.F., de Medeiros, A.K.A., Verbeek, H.M.W., Weijters, A.J.M.M., Van Der Aalst, W.M.: The ProM framework: a new era in process mining tool support. In: International Conference on Application and Theory of Petri Nets, pp. 444–454. Springer, Heidelberg (2005, June)
11. Kalenkova, A.A., van der Aalst, W.M., Lomazova, I.A., Rubin, V.A.: Process mining using BPMN: relating event logs and process models. *Software & Systems Modeling*, 1–30 (2015)
12. Kebede, M.: Comparative Evaluation of Process Mining Tools (2015)
13. Blyth, A.: Business process re-engineering. *ACM SIGGROUP Bulletin* **18**(1), 4–6 (1997)
14. Hammer, M., Champy, J.: Re-engineering the Corporation: A Manifesto for Business Revolution. Harper Business, New York (1993)
15. Fung, B.C., Trojer, T., Hung, P.C., Xiong, L., Al-Hussaeni, K., Dssouli, R.: Service-oriented architecture for high-dimensional private data mashup. *IEEE Trans. Serv. Comput.* **5**(3), 373–386 (2012)
16. Papazoglou, M.P., Van Den Heuvel, W.J.: Service-oriented design and development methodology. *Int. J. Web Eng. Technol.* **2**(4), 412–442 (2006)

17. Dai, W., Vyatkin, V., Christensen, J.H., Dubinin, V.N.: Bridging service-oriented architecture and IEC 61499 for flexibility and interoperability. *IEEE Trans. Ind. Inform.* **11**(3), 771–781 (2015)
18. Wang, Z., Xu, X., Zhan, D.: Normal forms and normalized design method for business service. In: *IEEE International Conference on e-Business Engineering, ICEBE 2005*, pp. 79–86 (2005, October)
19. Inaganti, S., Behara, G.K.: Service Identification: BPM and SOA Handshake. *BPTrends* **3**, 1–12 (2007)
20. Amsden, J.: Modeling SOA: Part 1. Service specification. *IBM Dev. Works* (2007)
21. Mani, S., Sinha, V.S., Sukaviriya, N., Ramachandra, T.: Using user interface design to enhance service identification. In: *IEEE International Conference on Web Services, ICWS'08*, pp. 78–87 (2008, September)
22. Jamshidi, P., Sharifi, M., Mansour, S.: To establish enterprise service model from enterprise business model. In: *IEEE International Conference on Services Computing, SCC'08*, vol. 1, pp. 93–100. *IEEE* (2008, July)
23. Dwivedi, V., Kulkarni, N.: A model driven service identification approach for process centric systems. In: *Congress on Services Part II, 2008. SERVICES-2*. *IEEE*, pp. 65–72 (2008, September)
24. Bianchini, D., Cappiello, C., De Antonellis, V., Pernici, B.: P2S: a methodology to enable inter-organizational process design through web services. In *International Conference on Advanced Information Systems Engineering*, pp. 334–348. Springer, Heidelberg (2009, June)
25. Yousef, R., Odeh, M., Coward, D., Sharieh, A.: BPAOntoSOA: a generic framework to derive software service oriented models from business process architectures. In: *Second International Conference on the Applications of Digital Information and Web Technologies, ICADIWT'09*, pp. 50–55 (2009, August)
26. Azevedo, L.G., Santoro, F., Baião, F., Souza, J., Revoredo, K., Pereira, V., Herlain, I.: A method for service identification from business process models in a SOA approach. In: *Enterprise, Business-Process and Information Systems Modeling*, pp. 99–112. Springer, Heidelberg (2009)
27. Kim, Y., Doh, K.G.: Formal identification of right-grained services for service-oriented modeling. In: *International Conference on Web Information Systems Engineering*, pp. 261–273. Springer, Heidelberg (2009, October)
28. Ren, M., Wang, Y.: Rule based business service identification using UML analysis. In: *2nd IEEE International Conference on Information Management and Engineering (ICIME)*, pp. 199–204 (2010, April)
29. Nikraves, A., Shams, F., Farokhi, S., Ghaffari, A.: 2PSIM: two phase service identifying method. In: *On the Move to Meaningful Internet Systems: OTM 2011*, pp. 625–634 (2011)
30. Kazemi, A., Rostampour, A., Azizkandi, A.N., Haghighi, H., Shams, F.: A metric suite for measuring service modularity. In: *Proceedings of the 2011 CSI International Symposium on Computer Science and Software Engineering (CSSE 2011)*. *IEEE, USA*, pp. 95–102 (2011)
31. Jamshidi, P., Mansour, S., Sedighiani, K., Jamshidi, S., Shams, F.: An Automated Service Identification Method, p. 2. Technical Report, TR-ASER-2012-01, Automated Software Engineering Research Group, Shahid Beheshti University (2012)
32. Soltani, M., Benslimane, S.M.: From a High Level Business Process Model to Service Model Artifacts-A Model-Driven Approach. In: *ICEIS*, vol. 3, pp. 265–268 (2012)
33. Birkmeier, D.Q., Gehlert, A., Overhage, S., Schlauderer, S.: Alignment of business and its architectures in the german federal government: a systematic method to identify services from business processes. In: *2013 46th Hawaii International Conference on System Sciences (HICSS)*, pp. 3848–3857 (2013, January)
34. El Amine, C.M., Benslimane, S.M.: Using combinatorial particle swarm optimization to automatic service identification. In: *13th International Arab Conference on Information Technology ACIT*, pp. 17–19 (2013)
35. Bianchini, D., Cappiello, C., De Antonellis, V., Pernici, B.: Service identification in interorganizational process design. *IEEE Trans. Serv. Comput.* **7**(2), 265–278 (2014)



36. Mohamed, M., Mohamed, B.S., Chergui, M.E.A.: A hybrid particle swarm optimization for service identification from business process. In: 2014 Second World Conference on Complex Systems (WCCS), pp. 122–127 (2014, November)
37. Leopold, H., Pittke, F., Mendling, J.: Automatic service derivation from business process model repositories via semantic technology. *J. Syst. Softw.* **108**, 134–147 (2015)
38. Jiang, L., Wang, J., Shah, N., Cai, H., Huang, C., Farmer, R.: A process-mining-based scenarios generation method for SOA application development. *Serv. Oriented Comput. Appl.* **10**(3), 303–315 (2016)
39. Gu, Q., Lago, P.: Service identification methods: a systematic literature review. In: European Conference on a Service-Based Internet, pp. 37–50. Springer, Berlin, Heidelberg (2010)
40. Klose, K., Knackstedt, R., Beverungen, D.: Identification of Services-A Stakeholder-Based Approach to SOA Development and its Application in the Area of Production Planning. In: ECIS, Vol. 7, pp. 1802–1814
41. Van der Aalst, W.M.P.: *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer, Heidelberg (2011)
42. AlShathry, O.: Process mining as a business process discovery technique. *Computer Engineering & Information Technology* (2016)
43. Paszkiewicz, Z.: Process mining techniques in conformance testing of inventory processes: an industrial application. In: International Conference on Business Information Systems, pp. 302–313. Springer, Heidelberg (2013, June)