

# A process-mining-based scenarios generation method for SOA application development

Jiang, L, Wang, J, Shah, N, Cai, H, Huang, C & Farmer, R

Author post-print (accepted) deposited by Coventry University's Repository

**Original citation & hyperlink:**

Jiang, L, Wang, J, Shah, N, Cai, H, Huang, C & Farmer, R 2015, 'A process-mining-based scenarios generation method for SOA application development' *Service Oriented Computing and Applications*, vol 10, no. 3, pp. 303-315

<https://dx.doi.org/10.1007/s11761-015-0188-8>

DOI 10.1007/s11761-015-0188-8

ISSN 1863-2386

ESSN 1863-2394

Publisher: Springer

*The final publication is available at Springer via <http://dx.doi.org/10.1007/s11761-015-0188-8>*

**Copyright © and Moral Rights are retained by the author(s) and/ or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This item cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder(s). The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holders.**

**This document is the author's post-print version, incorporating any revisions agreed during the peer-review process. Some differences between the published version and this version may remain and you are advised to consult the published version if you wish to cite from it.**

# A Process-mining-based Scenarios Generation Method for SOA Application Development

Lihong Jiang; Jianyi Wang;

School of Software  
Shanghai Jiao Tong University  
Shanghai, China

[jianglh@sjtu.edu.cn](mailto:jianglh@sjtu.edu.cn); [WangJianyi@sjtu.edu.cn](mailto:WangJianyi@sjtu.edu.cn)

Nazaraf Shah

Faculty of Engineering and Computing  
Coventry University  
Coventry, UK  
[aa0699@coventry.ac.uk](mailto:aa0699@coventry.ac.uk)

Hongming Cai; Chengxi Huang

School of Software  
Shanghai Jiao Tong University  
Shanghai, China  
[hmcai@sjtu.edu.cn](mailto:hmcai@sjtu.edu.cn); [hichens@sjtu.edu.cn](mailto:hichens@sjtu.edu.cn)

Ray Farmer

Faculty of Engineering and Computing  
Coventry University  
Coventry, UK  
[Csx254@coventry.ac.uk](mailto:Csx254@coventry.ac.uk)

**Abstract**—Business process models which are usually constructed by business designers from experience and analysis are the main guidelines for services composition in the SOA (Service-Oriented Architecture) applications development. However, due to the complexity of business models, it is a challenging task for business process designers to optimize the process models dynamically in accordance with changes in business environments. In this paper, a process-mining-based method is proposed to support business process designers to monitor efficiency or capture the changes of a business process. Firstly, we define a scenario model to depict business elements and their relationships which are critical to business process design. Based on the proposed scenario model, process mining algorithms, including control flow mining, roles mining and data flow mining are carried out in a certain sequence synthetically to extract business scenarios from event logs recorded by SOA application systems. Finally, we implement a prototype using a logistic scenario to illustrate the feasibility of our method in SOA applications development.

**Keywords**- SOA; business process model; process mining; scenario discovery

## I. INTRODUCTION

Business process management (BPM) systems provide a broader scope of perspectives to describe, configure and improve business processes to manage information systems as

opposed to workflow management (WFM) systems which focus mainly on processes automation for the organizations [1][24]. In the lifecycle of BPM, business process models play a significant role covering the stages of business processes design and system implementation. For business process management, two problems have attracted more attention of the researchers and practitioners. Firstly in many cases the process models could not represent all the possible required tasks and related control-flows in design stage. Secondly the process models are not adaptive enough to react to the dynamic environments at runtime. Therefore, it is important to propose methods to refine the business process models by monitoring process execution dynamically and realizing process optimization continuously.

In the BPM life cycle, there are two kinds of business process model refinement methods: these are model based and data based [33]. Model based methods often use simulation tools such as ARIS to verify the correctness or estimate the performance of the designed business processes before they are executed [34]. However the accuracy of the simulations usually depend heavily on the simulation environment configurations, which sometimes have deviations from the real environment. Comparatively, data based methods attracted more research attention because they better reflect the dynamic changes of the business processes at runtime as the event logs are collected and

analyzed[35]. In data based methods, the  $\alpha$ -series algorithms are designed to trace the business processes from event logs using Petri Nets, and the statistical algorithms are used to diagnose bottlenecks or resource contention during business process execution. Although the data based methods analyze business processes from multiple viewpoints, such as control flows, data flows and user roles, the weakness of these process-mining algorithms is that they are independent from each other and it is hard for users to analyze the performances or the changes of the whole business process synthetically.

This paper studies the methods of scenario generation from process logs perspective for the purpose of SOA system configuration, services governance and business process optimization. Instead of starting from the experiences of business experts of process modeling, we discover scenarios from the running environments (i.e., process instances, service and users) using process mining technology. We consider three types of business elements for scenario construction, which are the roles that carry out tasks in business processes, the tasks which are related to SOA service invocations and finally the business objects which are needed to fulfil the tasks requirements. In other words, we generate business scenarios from the executing environments not only considering the sequences of the tasks, but also the users and the business objects. Based on these elements, we propose a scenarios-based process mining method to construct comprehensive views of business scenarios to help improve the business processes. Furthermore, the scenarios from the executing business processes will help to understand the changes of the business processes more precisely.

The rest of the paper is organized as follows. In section 2, related work is reviewed. The motivation for our work is presented in Section 3. In Section 4, the detailed approach for mining scenarios from event logs is given. In section 5, we present a case study and a discussion to illustrate the scenario generation method in the area of logistic management. In Section 6, the conclusion and future work are described.

## II. RELATED WORK

With the application of SOA technology in the development of complex enterprise information systems [23][25], the analysis and modeling of complicated business processes pose challenges to the system analysts and developers. Various efforts have been made to model the operational business processes (e.g., Petri nets, BPMN, UML, and EPCs) [1]. In these methods, processes are described in terms of activities. However, due to the increasing complexity of the business models and the demands for the flexibility of enterprise information systems, additional factors, such as enterprise resources consumed by the business processes and the information delivering flow involved in the business processes, need to be considered in business process modeling and monitoring.

Therefore, in order to fully interpret the business processes for SOA system development, researchers have considered other factors to describe the ways business processes are executed and enterprise resources interact with each other (such as roles, business rules, and priorities of resources manipulation) to enrich the business process models. For instance, multi-view models describe a system from multiple perspectives for the different participants involved in the modeling phase to avoid

confusion about the business logic, system architecture and implementation [2]. Multi-view methods provide more comprehensive understanding of business processes from different viewpoints, such as organizational structures, functions and data. Although the designers could gain an overall analysis using multi-model method, relationships among the different views have not been fully described in the multi-view models. As a result, the designers find it challenging to optimise enterprise resource utilities during business process modeling or business process running.

Business scenarios provide ways to characterize the uncertainty in decision making as well as in business process requirements analysis [3]. Researchers also use scenarios to reduce uncertainties involved in multi-objective decisions [4]. A scenario-driven approach is also used to capture the requirement of business processes while UML sequence diagrams may be used to define the scenarios for business process modeling [5].

In order to extract business requirements more precisely, process mining methods are adopted to discover business scenarios from event logs.

Both clustering and abstracting approaches have been widely used in process mining based on event log analysis from different viewpoints to analyze business processes [29][30]. Researchers in [31] use a clustering approach to identify both outliers and groups of normal instances to deduce a multi-perspective process model. Clustering methods could also be used to learn process variant adaptations during runtime [32]. Abstracting methods deal with the relations between events and activities to mine the control flows and data flows involved in business processes [27][28]. In [6], a process mining method is used to provide detailed clinical and hospital analysis of the delivery of health care. Some researchers use process mining methods to identify the difference between observed behaviors and pre-modeled behaviors [7]. It is also used to discover service behavior patterns during business process execution in enterprise system [8]. In SOA systems, activities carried out by services are recorded in event logs and business processes could be extracted from event logs. SOA execution logs are used to mine Web service composition patterns and the gaps between the discovered model and the initial composite model are used to re-engineer the initial Web service composition model [26].

However, more research needs to be done using scenario driven methods to improve the business process models to adapt to the changes of business environments.

In this paper, scenarios are built through mining event logs from multiple views including control flows, organization structure and data flows for business process optimization and service orchestration. Using the scenarios, we aim to depict business processes in a more detailed way so that the quality of services composition could be improved.

## III. SCENARIO BASED BUSINESS PROCESS ANALYSIS AND IMPROVEMENT IN SOA

### A. Research Framework

In SOA-based information systems, Web services are often composed in a loosely coupled style to implement flexible IT infrastructures to support changing business models [9]. In order

to improve the correctness and efficiency of services composition in SOA, services orchestration and choreography should not only consider the business functional requirements, but also consider the required resources such as users, data entities and involved control flows in business process execution.

In this paper, an extended scenario-based business process model is adopted to involve the business context in services composition. The business contexts would help the business designers to understand and optimize the business processes more precisely. Meanwhile, business context would provide hints to the services developers to adjust the parameters of the services to meet new business requirements. Therefore, the whole business process lifecycle would cover activities of business process modeling, services development, services composition, information systems deployment and business process optimization based on mining. The proposed framework is shown in Fig. 1.

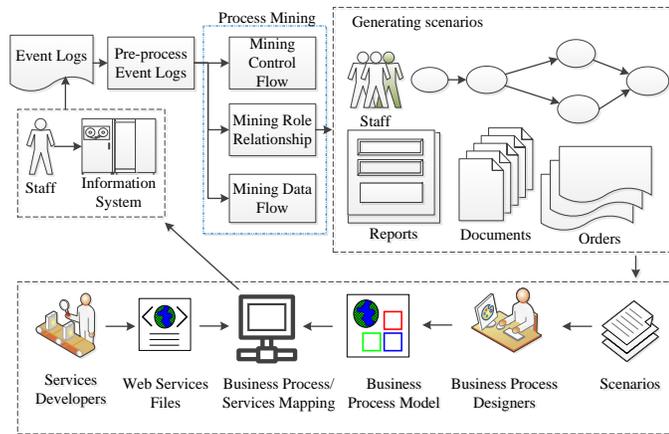


Figure 1. Research framework

In Fig.1, the proposed framework includes three main components, which are process mining module, scenario generation module, and services composition module. In the process mining module, the tools for control-flow mining, role-relationship mining and data flow mining are selected to analyze event logs of SOA services invocations. In the scenario generation module, business control flows are synergized with data flow and entity resources to provide business processes designers and services developers with a whole view of business processes containing business contexts.

### B. Business Scenario Model

In the scenario-based method of business process analysis and improvement in SOA shown in Fig.1, scenarios involving run-time contexts are crucial references for improving the business processes. We mine scenarios from event logs recorded in the SOA systems, and then these scenarios are defined as a composition of control flows, related data entities and task performers.

Control flows determine the logical sequences of tasks triggered and accomplished by SOA services invocation. Data entities are abstractions of conceptual and physical business objects related to business processes, such as product orders, manufacturing plans, and equipment states, etc., represented by

sets of properties [10]. Task performers are not limited to human beings such as customers, suppliers, and employees in a company. If some tasks are performed by a software component automatically without the direct interaction of people, such as services in SOA, these services should be regarded as task performers as well [11].

The aim of the design of our proposed business scenario model is to help business process analysts consider more critical business elements during business processes modification and optimization by capturing run-time business process changes reflected in SOA services invoking logs.

The following business elements are involved in the business scenarios.

#### Definition 1: Event

Events are the activities of task performing and SOA service invocation. Events are defined as tuples as follow.

$$\text{Event} = \langle \text{Case\_ID}, \text{Event\_ID}, \text{Timestamp}, \text{Performer}, \{BO\} \rangle$$

The *Case\_ID* is the serial number of the business process instance to be performed. The *Event\_ID* corresponds to the task or the SOA service name. *Timestamp* records the time the event occurs. *Performer* is the user carrying out the task. *{BO}* represents business objects involved in the event.

#### Definition 2: Execution Instance, Event log

Let  $E$  be the set of events,  $E^*$  be the set of all sequences composed of zero or more events from  $E$ . Then  $\sigma \in E^*$  denotes the execution instance while  $W \subseteq E^*$  denotes an event log.

#### Definition 3: Business Object (Bo)

As mentioned above, business objects are the abstractions of virtual and physical entities that are involved in the business processes.

$$Bo = \langle Bo\_ID, A \rangle;$$

$$A = \{a_1, a_2, \dots, a_m\} \quad (i = 1, 2, \dots, m);$$

*Bo\_ID* is the unique identification of the business object.  $A$  is the set of attributes of the business object.

#### Definition 4: Business Process (BP)

A business process is defined as the tuple of the unique identifier of the business process *BP\_ID*, a set of scenarios denoted as *SceS* and their relations denoted as *SceR*.

$$BP = \langle BP\_ID, SceS, SceR \rangle$$

#### Definition 5: Scenario

The main emphasis of business process reengineering should be put on the scenarios. A scenario is formed from a role, a set of ordered activities and a set of business objects. And the scenario should stipulate the correspondence between activities and the business objects. Thus, the scenario is defined as the following tuple:

$$\text{Scenario} = \langle \text{Sce\_ID}, \text{AS}, \text{AR}, \{BO\}, \text{ABO} \rangle$$

$Sce\_ID$  is the unique identifier of the scenario.  $AS$  names the activities included in the scenario and can be denoted as the following.

$$AS = \{Act_1, Act_2, Act_3, \dots\}$$

$$Act_i = \langle ActName \rangle$$

$AR$  defines the relations between the activities in  $AS$  so that the business process topology is defined by  $AR$ .

$$AR = \{ \langle Act_i, Act_j \rangle / Act_i, Act_j \in AS \text{ if } Act_i \text{ is related to } Act_j \}$$

$\{BO\}$  are the business objects involved in the scenario.

$ABO$  is a mapping function that maps an activity  $Act_i$  to a set of business objects used by the activity  $Act_i$  in this scenario.

$$ABO: AS \rightarrow 2^{|BS|}$$

$$ABO(Act_i) = \{Bo_1, Bo_2, Bo_3, \dots\};$$

**Definition 6: Role**

Usually in an organization employees are grouped into roles according to their assigned jobs, as well as the application systems. Different roles will have different allocated tasks and occupy different business objects.

$$Role = \langle Role\_ID, AS, \{BS\}, ABO \rangle;$$

The  $AS$  represents all activities carried on by this role in a business process.  $\{BS\}$  are business objects used by the role and  $ABO$  is the mapping function from one activity  $Act_i$  to a set of business objects used by the activity.

Based on above definitions, the business scenario model and the business elements in the model are shown in Fig. 2.

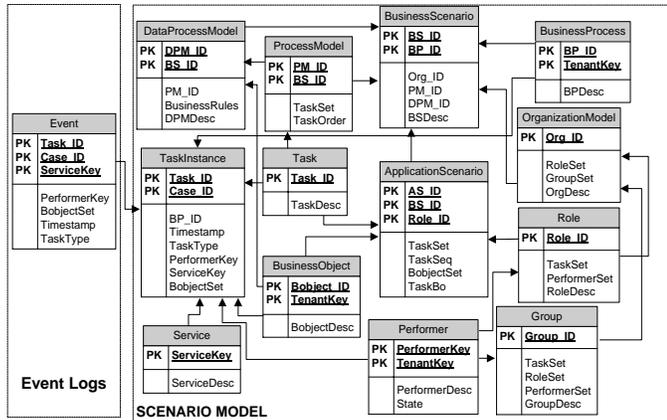


Figure 2. Business Scenario Model for Business Process Mining

As shown in Fig.2, in order to generate the scenarios from the event logs, we use process mining technology from the both organizational structure and the business process perspective. The organizational structure perspective focuses on business objects related to the tasks, such as who perform the tasks and how the task performers are related. The business process perspective focuses on the routines of the business process instances, for example, when manufacture orders are fulfilled, what kind of data flows would be involved.

#### IV. SCENARIO CONTEXT GENERATION USING PROCESS MINING

##### A. Event logs pre-processing

Currently, information systems such as ERP (Enterprise Resource Planning) systems, PDM (Product Data Management) systems, CRM (Customer Relation Management) systems, exist widely in various kinds of enterprises [12]. The event logs of these information systems record data about performed activities in the enterprises including time performed, actors and related business objects, as shown in Tab. 1.

TABLE I . FRAGMENT OF EVENT LOG

Case 1			
Event Id	Timestamp	Performer	Business objects
A	Case1Time1	P1	bo1, bo2
B	Case1Time2	P2	bo1
C	Case1Time3	P3	bo2
D	Case1Time4	P4	bo2
E	Case1Time5	P5	bo3

In SOA-based information systems, business process instances may be carried out by different services offered by different service developers. This brings challenge of understanding and analyzing business processes because event data are often scattered over different data sources and they are hard to be collected.

In order to address the above challenge, we use service invocation event logs as business process mining data sources, instead of using scattered running logs of information system. These SOA services invocation logs could be collected centrally on servers sides by recording the message exchanges [13] (e.g., SOAP messages) and read and write actions [14]. Since service-oriented applications are generally deployed on centralized runtime environments, such as Microsoft Azure Services Platform and Google App Engine, the adoption of such centralized runtime environments monitor the running of distributed services-oriented applications and obtain centralized service invocation event logs. These event logs can be transformed to a standard format, such as MXML (Mining eXtensible Markup Language) for further usage.

According to definition 1, event log files contain the triggering timestamps of the SOA services, names of the SOA services carrying on the activities, the ids of the business process instances including the activities, and the performers who initiate or perform the activities. We use MXML to represent these events logs, as shown in Fig.3.

```

<Process id="DEFAULT" description="Simulated process">
  <ProcessInstance id="1" description="order process instance">
    <AuditTrailEntry>
      <WorkflowModelElement>submit order</WorkflowModelElement>
      <EventType >start</EventType>
      <Timestamp>2006-01-01T00:00:00.000+01:00</Timestamp>
      <Performer>John</Performer>
      <BusinessObject>Commodity Order</BusinessObject>
    </AuditTrailEntry>
    ...
  </ProcessInstance>
</process>

```

Figure 3. MXML format for an event entry in log file

### B. Mining control flow

We use Petri net to represent the control flow of the SOA services invocations. We mine the sequential structure, parallel structure, and selective structure of the SOA services invocations using  $\alpha$  algorithm [15].

Using Petri nets, a SOA services invocation control flow could be defined as  $PM = \langle P, T, S, F \rangle$  where  $P$  is the finite set of places. The source place  $i$  and sink place  $o$  correspond to the start and end points in WF-net (WorkFlow-net).  $T$  stands for the finite set of transition in  $PM$ . Each transition corresponds to a single task in the business process.  $S$  is the set of resources of services used to perform the tasks in  $PM$ . We use  $si$  to denote the resource used to fulfil the  $i$ th task.  $F$  describes the relations between the places and the transitions denoted as  $F \subseteq (P * T) \cup (T * P)$ .

WF-net and JPDL (JBoss jBPM Process Definition Language) are therefore used to display the Petri net result of SOA services invocation control flows mining by  $\alpha$  algorithm, which is shown in Fig.4.

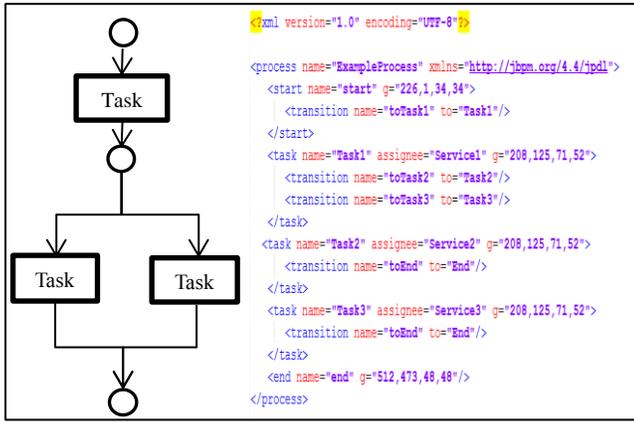


Figure 4. Control flow of SOA services invocation using WF-net/JPDL

A WF-net and its implementation using JPDL are shown in Fig.4. In JPDL, label `<process></process>` stands for an entire process model. Label `<task></task>` stands for the transition, e.g. the tasks in the business processes. Its attribute `assignee` stands for the service resource related to the task. `<start></start>` stands for starting a task, while `<end></end>` stands for end of a task. `<transition></transition>` is the input/output function of the transition.

### C. Mining the relations between roles

Roles are the actors who perform the tasks in the business processes. Usually business processes are performed by different roles cooperatively. The relations between the roles often reflect relations between the activities. Therefore mining the relations between the roles could help to monitor or discover business process changes from the view point of performer relations in addition to the control flows of SOA services invocations. The performers, whether people or software agents, are kinds of enterprise resources; the results of roles mining could help managers to optimize the allocation of human and services resources more efficiently.

In order to establish the role relations of the users, metrics of *Performer-Activity* are used to discover the relations between the activities and their performers [6]. As users belonging to the same role often conduct similar activities rather than totally different activities, the frequencies of the performers conducting the activities are counted and used to measure the distance between two performers, as shown in Table II, where  $X_{i,j}$  represents the frequency of performer  $i$  performing activity  $j$ .

TABLE II. THE PERFORMER-ACTIVITY MATRIX

	A	B	C	D	E
Performer P1	$X_{1,1}$	0	0	0	0
Performer P2	0	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	0
Performer P3	0	0	$X_{3,3}$	$X_{3,4}$	0
Performer P4	0	0	$X_{4,3}$	$X_{4,4}$	0
Performer P5	0	0	0	0	$X_{5,5}$

Here, we use Pearson's correlation coefficient to count the distance between two roles. The formula (1) is as follows.

$$PCC(p_1, p_2) = \frac{\sum_{j=1}^n ((x_{1j} - \frac{1}{n} \sum_{j=1}^n x_{1j}) * (x_{2j} - \frac{1}{n} \sum_{j=1}^n x_{2j}))}{\sqrt{(\sum_{j=1}^n (x_{1j} - \frac{1}{n} \sum_{j=1}^n x_{1j})^2 + \sum_{j=1}^n (x_{2j} - \frac{1}{n} \sum_{j=1}^n x_{2j})^2)}} \quad (1)$$

Using the above formula, the distances of each two roles is calculated to classify them into different work groups. The pseudocode of the algorithm is shown in Fig. 5.

```

Role Classification Algorithm
Input: P, TV // Input P is the set of performers, TV is the threshold
Output: G = (P; Arc; W) // Output G is the graph of performers distances,
for each p in P
  pi = P;
  for each elseP in P
    pj = elseP;
    If (pi ≠ pj) then
      Weight = CalculatePCC(pi, pj);
      //count the correlation coefficient from pi to pj
      If (Weight > TV) then
        generateArc(pi, pj);
      end if;
    end if;
  end for;
end for;
return G = (P; Arc; W);

```

Figure 5. The pseudocode of the roles classification algorithm

Using the classification algorithm, the roles shown in Table II could be divided into three groups, shown in Fig. 6.

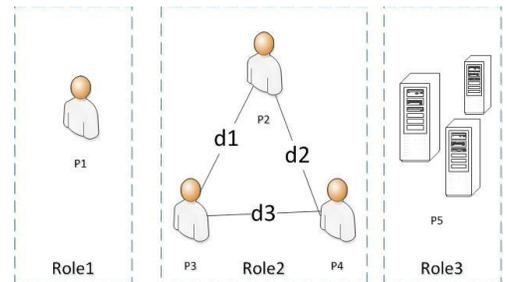


Figure 6. Role classifications derived from event logs

The result shows that  $P1$  can be assigned to  $Role1$ ,  $P2$ ,  $P3$  and  $P4$  can be put into the same group  $Role2$ , and  $P5$  could be assigned to  $Role3$ .

The relations between the work-groups are determined by the control flow structures of the activities. We have three types of control flow structures. They are sequential, parallel and selective. If two work-groups perform sequential activities, their relations are defined as *Hierarchical*. If two work-groups perform parallel activities, their relations are defined as *Cooperative*. If two work-groups perform selective activities, their relations are defined as *Alternative*.

#### D. Mining data flow

Data flows are transmission sequences of data between activities in business processes. They reflect the logical relations of the business activities performed by different roles. In many situations, the values of the parameters transmitted in the data flow determine the directions of the business process control and govern the choices of certain cases taking particular business paths [7]. For example, if an order is accepted, the customer should get a notification message sent automatically. These decisions could be made depending on the values assigned to business objects. Therefore, we can analyze the event logs to identify the possible data constraints affecting the structures of business processes, and obtain business rules used for further control of business processes.

We use the Daikon system proposed in [16] to detect the relations between the variables and the assigned values to deduce data constraints affecting the structures of the business processes. These data flows are helpful for the services developers to analyze or reengineer the business processes.

#### E. Scenario generation

According to control flows, role relations and data flows, scenarios are generated automatically from log data. The scenario could be generated using the algorithm, shown in Fig. 7.

```

Algorithm: GENERATE SCENARIOS
Input: control flow model (ordering of activities)
Scenario set SceS
  embed business objects set and branch conditions relevant in each activity
  put first activity into Scenario generated according to current role
  then call recursion(SceS, first activity's successor, current role)

recursion(Scenario Sce, Activity Act, Role preRole)
Input: Scenario Sce, current activity Act, previous role preRole
  if(Act's role ≠ preRole)
    Sce ← new scenario Sce
    put Sce into SceS
  end if
  put Act into Sce
  for each Act's successor ActSucc
    recursion(Sce, ActSucc, Act's role)
  end for
end recursion
end

```

Figure 7. Scenario generation

Fig. 7 shows that the generation of the scenarios starts by focusing on one of the roles in the scenarios. Then activities performed by this role are pulled into the same scenarios. After that, related activities according to the control flows are pulled into the scenarios to expand the range of the business processes. The algorithm will recur till all the users are pulled in.

## V. CASE STUDY AND DISCUSSION

### A. Case study

In this section, we use a prototype to illustrate our approach of scenario generation based on process mining.

It is believed that cloud computing technologies could improve the efficiency and the performance of resources allocation and many SOA applications are deployed on the cloud computing platforms nowadays [20-22]. The prototype is part of a SOA-based logistics management platform developed for the transportation bureau in one of the provinces in China. In the transportation bureau, logistics companies are registered to convey varieties of logistics transportations. In order to efficiently use the computational resources, the transportation bureau puts forward the project to build a cloud platform so that logistic companies could tenant ( or utilise) SOA services from the platform to deploy their logistic management systems.

The architecture of the cloud logistic management platform is shown as Fig.8.

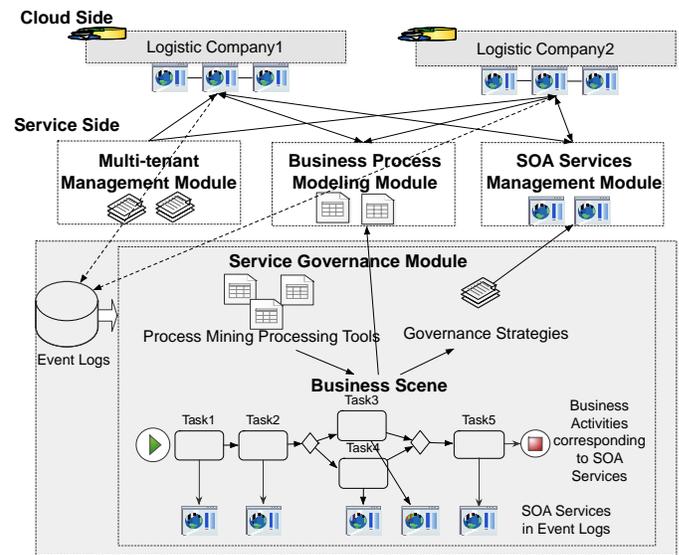


Figure 8. Architecture of the cloud logistic management platform

As shown in Fig.8, in the cloud side, the logistic companies use the business modeling tools provided by the platform to construct their business process models. On service side, SOA services could be selected to choreograph corresponding logistic information systems for the tenants according to the business process models. During the execution of the logistics information systems, event logs are collected on the service side. In this prototype system, the process-mining-based scenario generation method is used to analyze the event logs generated by the tenant companies. It could be seen in Fig.8 that the mined scenarios may help SOA service providers to build SOA governance strategies. Also the scenarios could be references to business process designers.

In the prototype, an open-source tool ProM is used to analyze the event logs [17] as shown in the following steps.

Step1: Control flow mining

We choose a plug-in  $\alpha$  algorithm of ProM to discover the control flow of the SOA services calling as shown in Fig.9.

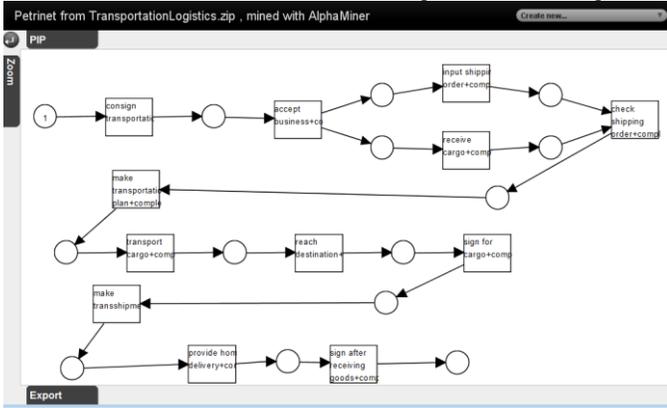


Figure 9. Control flow of intra-provincial logistics process

In Fig.9, the control flow of the business process of intra-provincial logistics is displayed.

### Step 2 Role relations mining

Users involved in the business processes are classified and divided into different roles according to the *Performer-Activity* metrics.

In this step, we use plug-in called Social Network tool in ProM to calculate the distances between two performers. It could be seen in Fig.10 that in this case study the performers could be divided into three types of roles, named drivers, delivers and logistics assistants, belonging to two work groups, named intra-provincial work group and extra-provincial work group.

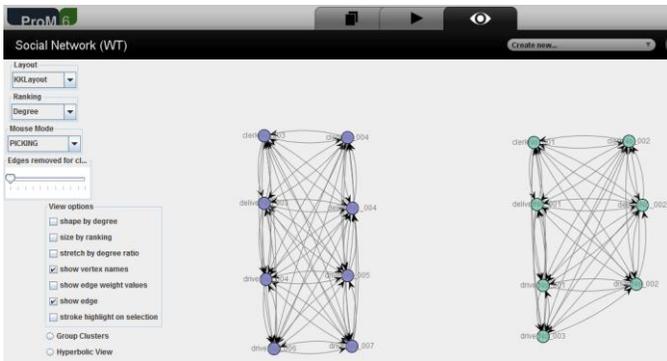


Figure 10. Generated work groups

Because the logistics company in this case study has two types of customers running intra-provincial and extra-provincial business separately, the users are divided into two work groups. It could be seen that in this case study the mining results are consistent with the real situation of the analyzed company. Therefore the mining results could help the business process designers better understand the business processes of the companies.

### Step 3: Data flow mining

In data flows, the constrain conditions which trigger the activities could be obtained by counting the values of the business objects involved in the business processes. In this case study, we use decision-tree algorithm implemented in ProM [7]

to calculate the values of the business objects and the results show that two business objects *TransportPlan* and *ShippingOrder* are critical in this business process, as demonstrated in Fig.11. They are the attributes related to shipping activities in the logistics process. The result shows that the task “check shipping order” will check the shipping destination and current location recorded in *TransportPlan* and *ShippingOrder* respectively. If *TransportPlan.Destination*  $\neq$  *ShippingOrder.CurrentLocation*, “transit shipment” will be carried out, otherwise, “make transportation plan” will be carried out.

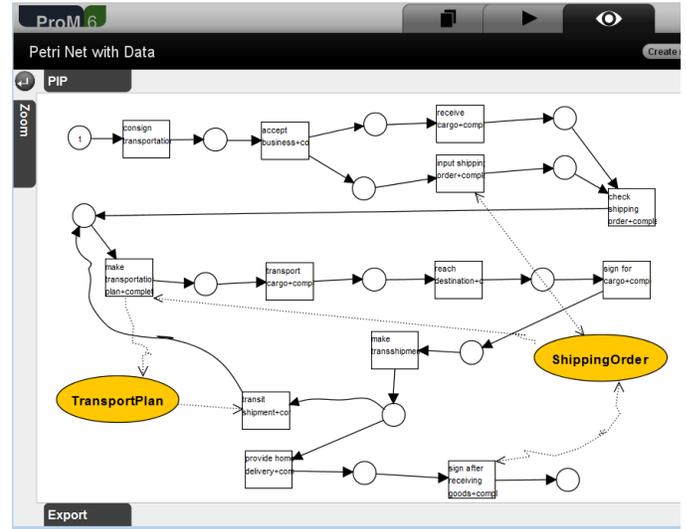


Figure 11. Data flow mining

### Step 4: Scenario generation

According to the results obtained in Step 2 and 3, the control flow shown in Fig.9 could be divided into several scenarios based on the algorithm shown in Fig. 6. Scenarios for the extra-provincial businesses are synthesized into one comprehensive business processes, as shown in Fig.12.

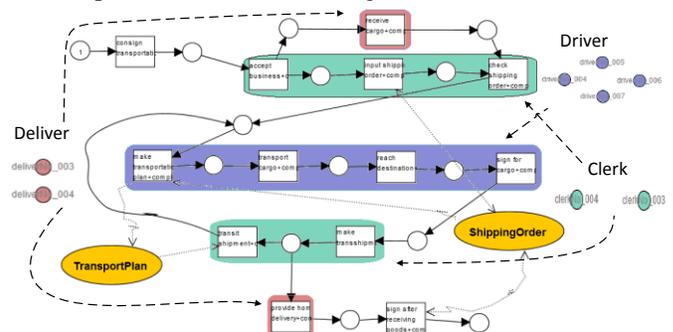


Figure 12. Scenario generation

In Fig.12, three scenarios are generated, each scenario corresponds to one type of roles, which are *Driver*, *Delivery-man*, and *Clerk* respectively. In these scenarios, business objects, tasks and roles are closely related to show the business processes more clearly to the business processes designers and SOA services developers.

In this case study, it could be seen that business processes are traced in the event logs of SOA service invocations. In our

method, we use  $\alpha$  algorithm to mine the control flow of the business activities. After that, role relation and data-flow mining algorithms are used sequentially at each point of activities along the generated control flow to obtain information about the required human resources and data entities during business process performing. These detailed business contexts may help designers to monitor and optimize the business process more clearly.

### B. Discussion

In the above case study, the prototype of the cloud logistics management platform was developed according to the real requirements of the logistic companies. We have simulated the running of the platform in the laboratory and generated an extra-provincial business scenario using event logs to show the effectiveness and applicability of our proposed method. Event logs are collected after the platform is deployed to conduct the comparison between our method and the others, we compare the scenario-based business process modeling method with multi-view business modeling methods such as ARIS [18] and Artifact [19]. Our comparison is based on provision of various functionalities in the proposed method rather than on comparison of logs collected as a result of running platform.

TABLE III. COMPARISON BETWEEN BUSINESS PROCESS MODELING METHODS

Features	Artifact	ARIS	Our approach
<b>Business Process Modeling</b>			
Oriented Users	Software Engineers	Business Process Designers	Software Engineers and Business Process Designers
Process Modeling Supporting Tools	Manually Modeling Supported by Visual Tools	Manually Modeling by Supported by Visual Tools	Automatically Generation by Process Mining Tools
Modeling Methodology	Abstract Expertise of IT engineers	Abstract Expertise of Domain Experts	Analyze Event Logs
Model Execution	Support by BPEL or BPMN	None	Support by BPEL after Model Mapping
Model Views	Single View of Processes	Multi Views of Processes, Roles and Data	Process-centric Multi Views of Tasks, Data, and Roles added in related Relations
<b>Business Process Optimization</b>			
Process Optimization Support	None	Yes	Yes
Optimization Method	None	Simulation Methods	Process Mining Algorithms
Business Process Monitoring	None	None	Yes
Adaption to Dynamic Business Process Changes	Low	Low	High

Table III demonstrates that the scenario-based business process model has the following advantages.

(1) More flexible for business modeling. Since the scenarios could be regarded as a process-centric artifact, the scenario could be easily encapsulated as business model and transformed for different disposing purposes. Business process management systems may be achieved by the scenarios conveniently and

modified through recombining these scenarios or adding new scenarios. For instance, if the business process in the case study should have a new part of asking help from external experts having existed in the scenario repository, it could be directly inserted into the process in accordance with the requirements.

(2) A comprehensive view for process optimization or refinement. Since the scenarios contain not only process model but also data objects and roles, we could refine the business processes based on more detailed contexts of run time. This could increase the efficiency of business process modeling.

## VI. CONCLUSION AND FUTURE WORK

In this paper, we emphasize three factors in business process analysis: the roles which invoke the SOA services, the business objects required in the business processes, and the control flows of the activities. We adopt the concept of business scenario to analyze and optimize the business process models based on the data of event logs. Two main contributions are summarized as follows:

(1) A method for scenario discovery using three perspectives of process mining is proposed to construct comprehensive business process models. In this scenario generation method, roles are centric. The users are classified into roles and work groups according to their distances calculated on the metrics of *Performer-Activity*. Activities related to the users are added into the scenarios to expand the covering range of the business process. The control flow structures are used to determine the relations between the roles. The scenarios are enriched by the constrain conditions calculated from the data flows.

(2) The scenario model is proposed for analysis and improvement of business process modeling, which is highlighted by its flexible composition and overall comprehensibility for technical or non-technical people. And the usability of scenario-based process improvement is discussed, including the scenario reconstruction and modification.

Although scenario-based technologies used in our research have shown advantages in capturing users' requirements for SOA systems development, the proposed method of scenario generation has limitations. The services invoked in the SOA systems should be well defined and the running traces should be recorded in the log files so that elements such as roles, activities and control flows could be extracted from the log data. In addition, the accuracy of the business process models built on the scenarios in our approach varies depending on the choices of the process mining algorithms.

Our future research will focus on following issues.

1. Algorithms need to be designed for mining more complex relations among the users involved in the business processes. Only simple relations such as clusters and rough hierarchical relations could be discovered in this paper. Algorithms need to be proposed to obtain more accurate sequential and hierarchical relations widely existed in the organization structure from the log files of SOA systems.

2. Methods of merging distributed scenarios need to be studied in future. In cloud computing, services are deployed in

distributed environment. In this paper, we use SOA services invoking logs centralized on service sides to analyze the business scenarios. The scattered scenarios are merged according to the control flows mined from the centralized event logs of SOA service invoking. Further research is needed to deal with the situation where event logs are scattered.

#### ACKNOWLEDGMENT

This research is supported by the National Natural Science Foundation of China under No. 61373030, 71171132.

#### REFERENCES

- [1] Van Der Aalst Wil MP (2013) Business process management: a comprehensive survey. ISRN Software Engineering:Article ID 507984. <http://dx.doi.org/10.1155/2013/507984>
- [2] Wang Q, Yung KL, Ip WH(2004) A hierarchical multi-view modeling for Networked Joint Manufacturing System. *Computers in Industry* 53(1):59-73
- [3] Kwakkel JH, Auping WL, Pruyt E (2013) Dynamic scenario discovery under deep uncertainty: the future of copper. *Technological Forecasting and Social Change* 80(4):789-800
- [4] Maghouli P, Hosseini SH, Oloomi Buygi M, Shahidepour M (2011) A scenario-based multi-objective model for multi-stage transmission expansion planning. *IEEE Transactions on Power Systems* 26(1):470-478
- [5] Ruokonen A, Kokko T, Systa T (2012) Scenario-driven approach for business process development. *International Journal of Business Process Integration and Management* 6(1):77-96
- [6] Partington A, Wynn M, Suriadi S, Ouyang C, Karnon J (2015) Process Mining for Clinical Processes: A Comparative Analysis of Four Australian Hospitals. *ACM Transactions on Management Information System* 5(4):Article 19
- [7] De Leoni M, Van Der Aalst Wil MP (2013) Data-aware process mining: discovering decisions in processes using alignments. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*:1454-1461
- [8] Aalst W van der (2013) Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior. *IEEE Transactions on Services Computing* 6(4):525-535
- [9] Chesbrough H, Spohrer J (2006) A research manifesto for services science. *Communications of the ACM* 49(7):35-40.
- [10] Grabowik C, Knosala R (2003) The method of knowledge representation for a CAPP system. *Journal of Materials Processing Technology* 133(1):90-98
- [11] Yu H, Zhu C, Cai H, Xu B(2009) Role-Centric RESTful Services Description and Composition for E-Business Applications. In *Proceedings of IEEE International Conference on e-Business Engineering*:103-110
- [12] Gunther CW, Rinderle-Ma S, Reichert M, van der Aalst Wil M.P., Recker J (2008) Using process mining to learn from process changes in evolutionary systems. *International Journal of Business Process Integration and Management* 3(1):61-78
- [13] Van Der Aalst Wil MP, Dumas M, Ouyang C, Rozinat A, Verbeek E (2008) Conformance checking of service behavior. *ACM Transactions on Internet Technology* 8(3):Article No.13
- [14] Diniz PC, Ferreira DR (2008) Automatic extraction of process control flow from I/O operations. in *Proceedings of 6th International Conference Business Process Management*:342-357
- [15] Van der Aalst W, Weijters T, Maruster L (2004) Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering* 16(9):1128-1142
- [16] Ernst MD, Cockrell J, Griswold, WG, Notkin D (2001) Dynamically discovering likely program invariants to support program evolution. *IEEE Transactions on Software Engineering* 27(2):99-123
- [17] Verbeek HMW, Buijs Joos CAM, Van Dongen BF, Van Der Aalst Wil MP (2011) Xes, xesame, and prom 6. *Information Systems Evolution. Lecture Notes in Business Information Processing* 72:60-75
- [18] Repa V, Železník O (2014) Methodological Limitations of Modeling Languages BPMN and ARIS. In the proceedings of 15th International Carpathian Control Conference: 507-512
- [19] Mens T, D'Hondt T(2000) Automating support for software evolution in UML. *Automated Software Engineering* 7(1):39-59
- [20] M. Reza Rahimi, et al.(2014) Mobile Cloud Computing: A Survey, State of Art and Future Directions. *MONET* 19(2): 133-143
- [21] M. Reza Rahimi, et al.(2013) MuSIC: Mobility-Aware Optimal Service Allocation in Mobile Cloud Computing. *IEEE CLOUD* 2013. 75- 82
- [22] Guiyi Wei, et al.(2010) A game-theoretic method of fair resource allocation for cloud computing services. *The Journal of Supercomputing*.54(2): 252-269
- [23] Qiang Duan,et al.(2012) A Survey on Service-Oriented Network Virtualization Toward Convergence of Networking and Cloud Computing. *IEEE Transactions on Network and Service Management*.9(4): 373-392
- [24] Q.Z Sheng, et al. (2014) Web services composition: A decade's overview. *Information Sciences* 280, 218-238
- [25] J.T Yao,et al.(2013) Granular computing: perspectives and challenges. *IEEE Transactions on Cybernetics*, 43 (6), 1977-1989
- [26] Walid Gaaloul, Karim Baïna, Claude Godart(2008) Log-based mining techniques applied to Web service composition reengineering. *SOCA (2008)*:93-110, DOI 10.1007/s11761-008-0023-6
- [27] Thomas Baier, Jan Mendling, Mathias Weske(2014) Bridging abstraction layers in process mining. *Information Systems*. 46 (2014):123-139
- [28] Ying Liu, Hui Zhang, Chungping Li, Roger Jianxin Jiao(2012) Workflow simulation for operational decision support using event graph through process mining. *Decision Support Systems*. 52 (2012):685–697
- [29] Jochen De Weerd, Annelies Schupp(2013) An Vanderloock, Bart Baesens, Process Mining for the multi-faceted analysis of business processes-A case study in a financial services organization. *Computers in Industry*. 64 (2013):57–67
- [30] Marcella Rovani, Fabrizio M. Maggi, Massimiliano de Leoni, Wil M.P. van der Aalst, Declarative process mining in healthcare. *Expert Systems With Applications*. 42 (2015):9236–9251
- [31] Francesco Folino, Gianluigi Greco, Antonella Guzzo, Luigi Pontieri(2011) Mining usage scenarios in business processes: Outlier-aware discovery and run-time prediction. *Data & Knowledge Engineering*. 70(2011):1005–1029
- [32] Chen Li, Manfred Reichert, Andreas Wombacher, Mining business process variants: Challenges, scenarios, algorithms. *Data & Knowledge Engineering*. 70(2011):409–434
- [33] Wil M. P. van der Aalst, *Business Process Management: A Comprehensive Survey*, ISRN Software Engineering, Vol. 2013, Article ID 507984, <http://dx.doi.org/10.1155/2013/507984>
- [34] Jimin Ling, Qi Feng, Li Zhang, A business process simulation method supporting resource evolution, *Proceedings of the 2014 International Conference on Software and System Process*,169-177
- [35] Wil Van Der Aalst, J. Leon Zhao, Harry Jiannan Wang, Editorial: Business Process Intelligence: Connecting Data and Processes, *ACM Transactions on Management Information Systems (TMIS)*, Volume 5 Issue 4, March 2015:18e:1-18e:5