



## Business Process Management Journal

Mining business rules from business process model repositories

Jantima Polpinij, Aditya Ghose, Hoa Khanh Dam,

### Article information:

To cite this document:

Jantima Polpinij, Aditya Ghose, Hoa Khanh Dam, (2015) "Mining business rules from business process model repositories", Business Process Management Journal, Vol. 21 Issue: 4, pp.820-836,

<https://doi.org/10.1108/BPMJ-01-2014-0004>

Permanent link to this document:

<https://doi.org/10.1108/BPMJ-01-2014-0004>

Downloaded on: 17 August 2018, At: 12:33 (PT)

References: this document contains references to 30 other documents.

To copy this document: [permissions@emeraldinsight.com](mailto:permissions@emeraldinsight.com)

The fulltext of this document has been downloaded 676 times since 2015\*

### Users who downloaded this article also downloaded:

(2014), "Ten principles of good business process management", Business Process Management Journal, Vol. 20 Iss 4 pp. 530-548 <a href="https://doi.org/10.1108/BPMJ-06-2013-0074">https://doi.org/10.1108/BPMJ-06-2013-0074</a>

(2015), "Surveys in business process management – a literature review", Business Process Management Journal, Vol. 21 Iss 3 pp. 692-718 <a href="https://doi.org/10.1108/BPMJ-07-2014-0065">https://doi.org/10.1108/BPMJ-07-2014-0065</a>

Access to this document was granted through an Emerald subscription provided by emerald-srm:478405 []

### For Authors

If you would like to write for this, or any other Emerald publication, then please use our Emerald for Authors service information about how to choose which publication to write for and submission guidelines are available for all. Please visit [www.emeraldinsight.com/authors](http://www.emeraldinsight.com/authors) for more information.

### About Emerald [www.emeraldinsight.com](http://www.emeraldinsight.com)

Emerald is a global publisher linking research and practice to the benefit of society. The company manages a portfolio of more than 290 journals and over 2,350 books and book series volumes, as well as providing an extensive range of online products and additional customer resources and services.

Emerald is both COUNTER 4 and TRANSFER compliant. The organization is a partner of the Committee on Publication Ethics (COPE) and also works with Portico and the LOCKSS initiative for digital archive preservation.

\*Related content and download information correct at time of download.

# Mining business rules from business process model repositories

Jantima Polpinij

*Department of Computer Science, Faculty of Informatics,  
Mahasarakham University, Mahasarakham, Thailand, and*

Aditya Ghose and Hoa Khanh Dam

*School of Computer Science and Software Engineering,  
University of Wollongong, Wollongong, Australia*

## Abstract

**Purpose** – Business process has become the core assets of many organizations and it becomes increasing common for most medium to large organizations to have collections of hundreds or even thousands of business process models. The purpose of this paper is to explore an alternative dimension to process mining in which the objective is to extract process constraints (or business rules) as opposed to business process models. It also focusses on an alternative data set – process models as opposed to process instances (i.e. event logs).

**Design/methodology/approach** – The authors present a new method of knowledge discovery to find business activity sequential patterns embedded in process model repositories. The extracted sequential patterns are considered as business rules.

**Findings** – The authors find significant knowledge hidden in business processes model repositories. The hidden knowledge is considered as business rules. The business rules extracted from process models are significant and valid sequential correlations among business activities belonging to a particular organization. Such business rules represent business constraints that have been encoded in business process models. Experimental results have indicated the effectiveness and accuracy of the approach in extracting business rules from repositories of business process models.

**Social implications** – This research will assist organizations to extract business rules from their existing business process models. The discovered business rules are very important for any organization, where rules can be used to help organizations better achieve goals, remove obstacles to market growth, reduce costly mistakes, improve communication, comply with legal requirements, and increase customer loyalty.

**Originality/value** – There has very been little work in mining business process models as opposed to an increasing number of very large collections of business process models. This work has filled this gap with the focus on extracting business rules.

**Keywords** Information management, Process management, Software engineering, Business process redesign, Knowledge management, Process modelling

**Paper type** Research paper

## 1. Background

A business process is defined as consisting of a set of logically related activities, performed by their relevant roles or collaborators, to intentionally achieve the common business goals (Ye *et al.*, 2007). Business process has become the core assets of many organizations since they generate revenue and often represent a significant proportion of costs. Modeling and managing business processes is an important approach for managing organizations from an operational perspective. In fact, a recent study has shown that the business process management (BPM) software market reached nearly \$1.7 billion in total software revenue in 2006 and this number continues to grow (Hill *et al.*, 2007). It now becomes common for most medium to large to have collections



of hundreds or even thousands of business process models. For example, the BIT Process Library has 735 process models, the SAP Reference Model contains 604 process models, and there are 6,000+ process models in Suncorp's process model repository for insurance. Such large collections of process models form a valuable source of data which knowledge can be harvested.

In fact, there have been emerging interests in business intelligence which focusses on mining business-relevant data to extract meaningful and useful information for business purposes. For example, many techniques have been proposed to mine event logs (i.e. process mining) for the analysis, design, and enactment of business processes (van der Aalst *et al.*, 2008). Process mining help discover business process models which can be used as the basis for process improvement, understanding, communication, execution, analysis, and simulation (van der Aalst, 2011). In addition, process-centric business intelligence which focusses on the analysis of business-relevant data in the operational business process plays an important part in an organization's strive for competitiveness.

Business rules are defined and constrained as various aspects of the business (Ross, 1997). Simply speaking, business rules can be used as a guide for the operation of an organization. Despite its importance, discovering business rules is very costly in complex organizations. One method of finding business rules is process mining (van der Aalst *et al.*, 2008). Process mining mainly focusses on the derivation of process models from event logs (van der Aalst, 2011). However, there are settings in which process mining proves to be inadequate. If a process has never been performed, no event logs information is available, rendering the technique useless. In addition, learning/data mining techniques, such as those used in this approach to process, only return reliable results when there are sizeable data sets (in the case of process mining, event logs) available. If the size of the set of event logs is small (which is often the case with infrequently executed processes, such as those associated with disaster management, or with Greenfield applications), the reliability of the knowledge extracted cannot be guaranteed.

There has been, however, limited work in mining process models (alternatively referred to as process designs). In this paper, we present a new solution to find significant business rules hidden in business processes. We aim to explore an alternative dimension to process mining in which the objective is to extract process constraints (or business rules) as opposed to process models. It also focusses on an alternative data set – process models as opposed to process instances (i.e. event logs). This approach is valuable in the settings discussed above, in which process mining techniques do not return reliable results. The expected results of mining process model repositories are business rules, where a business rule defines one aspect of IT system that is intended to assert IT system structure or influence the behavior of IT system. Our business rules extracted from process models represent significant and valid sequential correlations among business activities belonging to a particular organization. Such business rules represent business constraints that have been encoded in business process models. Also, business rule is a critical component of a business process re-engineering (BPR) (Yu and Mylopoulos, 1994; Ross, 1997). The success of BPR projects usually have more to do with rethinking business rules, than with re-doing business process (or workflow). This is because business rules are more basic to the company (or enterprise, or organization) than business process (Ross, 1997).

## 2. Related work

Business rules (Ross, 2003; Morgan, 2002) are statements about business policies defining your business, where a business rule is often a constraint or a description of an

operation within your business. Also, a business rule can be associated to a class, to define common properties of instances, or to an operation, including defining of its effects. In general, the types of business rules can be classified into three different types: integrity constraints, derivation rules, and operation rules. In general, business rules may be informal or even unwritten (Sneed and Erdos, 1996; Shao *et al.*, 2002), and writing the rules down clearly and making sure that they do not conflict is a valuable activity (Ross, 2003). Integrity constraints are the business rules that can be found on the data of the application. Derivation rules are business rules that can be derived from other information. Operation rules are the business rules that describe the effects of an operation. Business rules are very important for any organization, where rules can be used to help the organization to better achieve goals, remove obstacles to market growth, reduce costly mistakes, improve communication, comply with legal requirements, and increase customer loyalty.

In general, when a business is changed in the marketplace, its applications are also required to change. Then, application development teams are expected to release changes and improvements in a short time. Therefore, when an organization wants to improve/modify its business policy, it needs to know current business rules used. Although business rules can be used to define modern business application, it is complicated to provide business rules that govern business decisions. This is because these rules are not easy to understand and verify accuracy. However, many organizations today have their business rules embedded in several existing artifacts such as business processes, applications, or databases (Sneed and Erdos, 1996; Shao *et al.*, 2002; Wang *et al.*, 2004). Therefore, it is possible to extract business rules from these existing applications. This technique can be called business rules mining.

A well-known solution of finding business rules can be done by process mining (van der Aalst *et al.*, 2008). Process mining provides the capability to discover, detect, control, organize, and monitor actual process execution by extracting useful knowledge from event logs (i.e. process instances) (van der Aalst and Weijter, 2004; van der Aalst *et al.*, 2008; van der Aalst, 2011). However, there has been limited work in process mining because it mainly focusses on the derivation of process models from event logs. If a process has never been performed, no event logs information is available, rendering the technique useless.

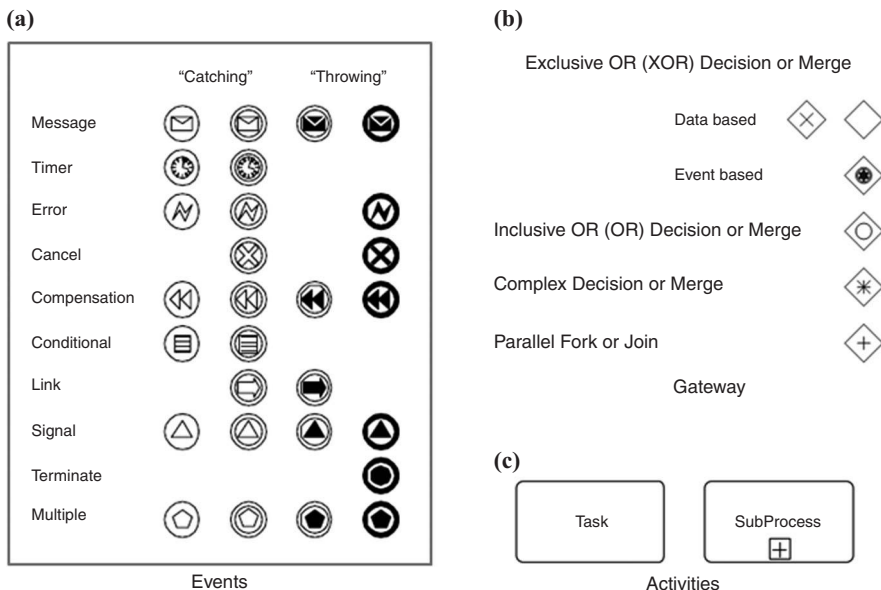
As above, some alternative solutions are required. Other solutions to obtain business rules is to extract them from existing applications where business rules can be found in the business documents/systems (Wang *et al.*, 2004; Shao *et al.*, 2002). Some of researchers argued that business rules are hidden in legacy systems (Wang *et al.*, 2004). Wang *et al.* (2001) proposed a tailored solution approach to the business rule extraction, which combines variable classifications, program slicing, and hierarchical abstraction among other maintenance techniques. However, it is very difficult to extract business rules from legacy systems because of the inconsistency of documentation (Wang *et al.*, 2004). This solution has been limited when they are applied to large complex legacy systems. In general, large legacy systems involve large amount of code, domain variables, synonym variables and business rules, which make it more difficult to extract business rules. As this, Wang *et al.* (2004) proposed a framework, which offers distinct advantages over normal extraction solutions for large legacy systems. This framework consists of five steps: slicing program, identifying domain variables, data analysis, presenting business rules, and business validation. Furthermore, business rules can be embedded in source code. Therefore, some proposed techniques are used to extract business rules from source code (Sneed and Erdos, 1996). They proposed a method for identifying and extracting

business rules by means of data output identification and program stripping. This method has been implemented in a reverse engineering tool SOFT-REDOC for COBOL programs. Finally, the results of this work were used to make more understanding in comprehending legacy programs.

In our earlier work (Polpinij *et al.*, 2010), we have proposed an alternative approach of process mining to find business rules from process models (alternatively referred to as process designs), where process designs are considered as process instances. We have proposed a preliminary technique for extracting business rules from repositories of business process designs or models, leveraging the well-known a priori algorithm. Such business rules are then used as a prior knowledge for further analyzing, verifying, and modifying process designs.

### 3. BPM and notation

Business Process Modeling Notation (BPMN) defines a business process diagram (BPD) for creating graphical models of business process operations (Miers and White, 2008; Muehlen and Recker, 2008). BPD is based on a flowcharting technique. A business process model is a network of graphical objects representing activities/works/tasks and the flow controls that defines their order of performances. There are four basic categories of elements: flow objects, connecting objects, swimlanes, and artifacts. Flow objects consist of three objects: event, activity, and gateway. The flow objects are connected together in a diagram by the connecting objects. The connecting objects consist of sequence flow, message flow, and association. They are two types of swimlanes: pool and lane. A pool is used to represent a participant in a process, while lane is used to organize and categorize activities. Artifacts can be used to provide the additional detail and context in a specific model. The core set of BPMN basics can be represented in Figure 1. Each of BPMN types can be extended to other elements. For example, all of the event elements can be shown in Figure 1. Consider an example of



**Figure 1.**  
Some of event  
elements

process model in Figure 2. It can be seen some workflows and flow controls that might be driven under some business rules. Simply speaking, some business rules may be hidden in the process model.

However, after the data set of process models is considered carefully, it can be found that directly extracting business rules from process models represented on graphical diagram such BPMN is a difficult task. Therefore, the collection of process models will be transformed into a textual format from which it is easier to extract knowledge. This work uses the XML Metadata Interchange (XMI)[1] standard, which is an Object Management Group (OMG) standard for exchanging metadata information through Extensible Markup Language (XML). An example of business process is drawn through BPMN can be presented in Figure 2.

4. Transforming process models into activity sequences

Our approach has two main steps. In the first step, we transform a business process model into a number of activity sequences, each of which represents an alternative workflow. Details of the first step are provided in this section. The outcome of the first step, a repository of activity sequences, is mined to extract business rules. This is done in the second step which will be described in detail in the next section.

A business process model can be transformed into a number of activity sequences. Each activity sequence is denoted as  $S = \langle s_1, s_2, \dots, s_i \rangle$ , where  $s_i$  can be an activity set. An activity set is a set of activities and/or activity-sequences that are performed together (i.e. in parallel). For example,  $\{a_1, a_2\}$  denotes activity  $a_1$  and  $a_2$  performed in parallel whilst  $\{a_1, \langle \{b_1\}, \{b_2\} \rangle\}$  denotes activity  $a_1$  and activity sequence  $\langle \{b_1\}, \{b_2\} \rangle$  performed in parallel. A activity sequence is sometimes denoted as  $\langle s_1, s_2, \dots, s_n \rangle_{cond}$ , meaning that  $s_1, s_2, \dots, s_n$  would be performed (in sequence) under the condition  $cond$  is true.

Table I shows four basic scenarios where a BPMN business process model is translated into activity sequences. In case (a), since there is no gateway in the business process, the transformation is straightforward: the business process is translated into one activity sequence of three single activity-sets  $\{A\}$ ,  $\{B\}$ , and  $\{C\}$ . In case (b), the AND gateway indicates that activity sequence  $\langle \{X\}, \{M\} \rangle$  is performed in parallel with activity sequence  $\langle \{Y\} \rangle$ . Consequently, both of them together form an activity set  $\{ \langle \{X\}, \{M\} \rangle, \langle \{Y\} \rangle \}$  which is preceded by  $\{A\}$  in the activity sequence generated from the process (b).

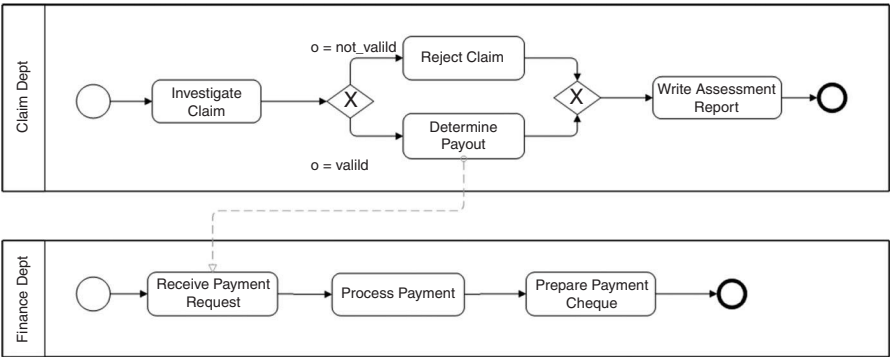
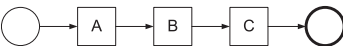
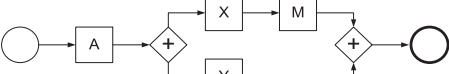
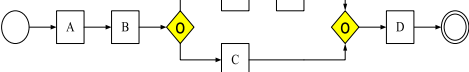
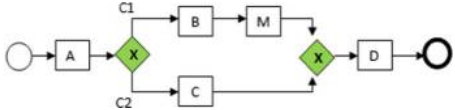


Figure 2.  
An example  
of process  
model drawn  
through BPMN

Case	Original process models	Gateway	Activity sequences
(a)		None	S1: $\langle \{A\}, \{B\}, \{C\} \rangle$
(b)		AND	S1: $\langle \{A\}, \{ \langle \{X\}, \{M\} \rangle, \langle \{Y\} \rangle \rangle$
(c)		OR	S1: $\langle \{A\}, \{B\}, \langle \{E\}, \{F\} \rangle, \{D\} \rangle$ S2: $\langle \{A\}, \{B\}, \langle \{C\} \rangle, \{D\} \rangle$ S3: $\langle \{A\}, \{B\}, \{ \langle \{E\}, \{F\} \rangle, \langle \{C\} \rangle \rangle, \{D\} \rangle$
(d)		XOR	S1: $\langle \{A\}, \langle \{B\}, \{M\} \rangle_{C1}, \{D\} \rangle$ S2: $\langle \{A\}, \langle \{C\} \rangle_{C2}, \{D\} \rangle$

**Table I.**  
Examples of  
generating sequences  
from process models

Case (c) shows an example where the process has an OR gateway and is thus translated into three different activity sequences which correspond to three alternative scenarios: only activities *E* and *F* are performed; only activity *C* is performed; and activities *E* and *F*, and activity *C* are performed in parallel, denoting as activity-set  $\{ \langle \{E\}, \{F\} \rangle, \langle \{C\} \rangle \}$ . Finally, the process in case (d) which has an XOR gateway is translated into two activity sequences, each of which contain a conditional activity sequence corresponding to each branch of the XOR gateway:  $\langle \{B\}, \{M\} \rangle_{C1}$  corresponding to the branch where *C1* is true, whereas  $\langle \{C\} \rangle_{C2}$  corresponding to the branch where *C2* is true. Table II show all the activity sequences that are generated from a repository of seven business processes that we will use as an example in this paper.


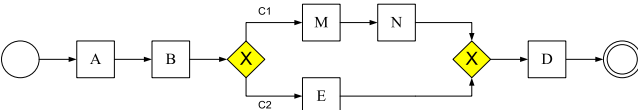
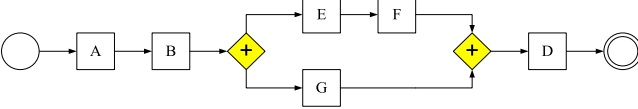
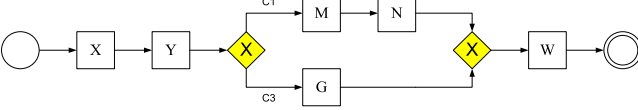
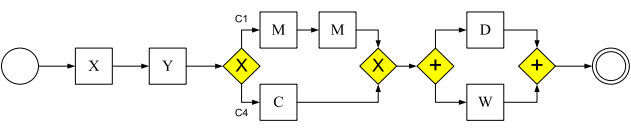
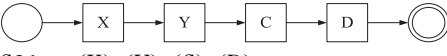
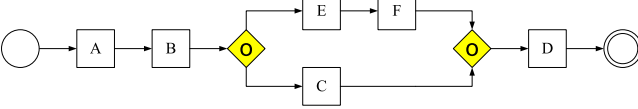
## 5. Extracting activity sequences

Given a large collection of activity sequences, the problem of mining business rules is to find all sequences whose support is greater than the user-specified minimum support. Such activity sequences represent sequential patterns (also called frequent sequences) and are considered in our context as business rules.

Before explaining our approach, let us define the concept of subsequence. Sequence  $\langle a_1, a_2, \dots, a_n \rangle$  is a subsequence of sequence  $\langle b_1, b_2, \dots, b_m \rangle$  if there exists integer  $i_1 < i_2 < \dots < i_n$  such as  $a_j$  is a subset or subsequence of  $b_{ij}$  (depending on  $b_{ij}$  is an activity set or an activity sequence) for all  $j = 1, \dots, n$ . For example, sequence  $\langle \{B\}, \{E\}, \{D\} \rangle$  is a subsequence of sequence  $\langle \{A\}, \{B\}, \langle \{E\}, \{F\} \rangle, \{D\} \rangle$ , whereas sequence  $\langle \{C\}, \{D\} \rangle$  is not.

The support count for a sequence is defined as the number of activity sequences in the total activity sequences that contain this sequence. The number of elements in activity sequence  $p$  is called the length of the sequence ( $L$ ), denoted as  $|p|$ . An activity sequence with length  $k$  is called  $k$ -sequence. The  $i$ th element in the activity sequence  $p$  is represented by  $p_i$ . The empty activity sequence is denoted by  $\langle \rangle$ . The result of the concatenation of two activity sequences  $a$  and  $b$  is a new activity sequence  $p$ , denoted as  $p = ab$ .

**Table II.**  
Examples of  
generating sequences  
from process models

Process models	Generating activity sequences from process models
<i>P1</i> Original form Activity sequences	 $S_{1,1}: \langle \{A\}, \{B\}, \{C\}, \{D\} \rangle$
<i>P2</i> Original form Activity sequences	 $S_{2,1}: \nu \langle \{A\}, \{B\}, \langle \{M\}, \{N\} \rangle_{C1}, \{D\} \rangle$ $S_{2,2}: \langle \{A\}, \{B\}, \langle \{E\} \rangle_{C2}, \{D\} \rangle$
<i>P3</i> Original form Activity sequences	 $S_{3,1}: \langle \{A\}, \{B\}, \{ \langle \{E\}, \{F\} \rangle, \langle \{G\} \rangle, \{D\} \rangle$
<i>P4</i> Original form Activity sequences	 $S_{4,1}: \langle \{X\}, \{Y\}, \langle \{M\}, \{N\} \rangle_{C1}, \{W\} \rangle$ $S_{4,2}: \langle \{X\}, \{Y\}, \langle \{G\} \rangle_{C3}, \{W\} \rangle$
<i>P5</i> Original form Activity sequences	 $S_{5,1}: \langle \{X\}, \{Y\}, \langle \{M\}, \{N\} \rangle_{C1}, \{ \langle \{D\} \rangle, \langle \{W\} \rangle \} \rangle$ $S_{5,2}: \langle \{X\}, \{Y\}, \langle \{C\} \rangle_{C4}, \{ \langle \{D\} \rangle, \langle \{W\} \rangle \} \rangle$
<i>P6</i> Original form Activity sequences	 $S_{6,1}: \langle \{X\}, \{Y\}, \{C\}, \{D\} \rangle$
<i>P7</i> Original form Activity sequences	 $S_{7,1}: \langle \{A\}, \{B\}, \langle \{E\}, \{F\} \rangle, \{D\} \rangle$ $S_{7,2}: \langle \{A\}, \{B\}, \langle \{C\} \rangle, \{D\} \rangle$ $S_{7,3}: \langle \{A\}, \{B\}, \{ \langle \{E\}, \{F\} \rangle, \langle \{C\} \rangle, \{D\} \rangle$

Our approach uses mining sequence techniques that are widely used in data mining. Specifically, we adapted the Generalized Sequential Pattern Mining algorithm (Agrawal and Srikant, 1995). The algorithm makes multiple passes over the repository of activity sequences. In the first pass, we try to identify the support for each individual activity, i.e. the number of activity sequences that include the activity. Activities that have minimum



support are identified as frequent activities, each of which gives one-element frequent sequence consisting of that activity. In each subsequent pass, we begin with a seed set which contains the frequent sequences identified in the previous pass. We use the seed set to generate candidate sequences, which are new potentially frequent sequences, each of which should have one more activity than a seed sequence. We then determine the support for all candidate sequences by doing a pass over activity sequence repository and those that have minimum support become frequent sequences and are used as the seed for the subsequent pass. The algorithm terminates in two situations: no candidate sequences are generated; and no frequent sequences are found at the end of a pass.

The algorithm is summarized as follows:

1. Scan activity sequence repository and take sequences as length-1 ( $L_1$ ) candidate with min-support.
2. For each level (e.g. sequence of length- $k$ ) do
  - 2.1 Scan process repository to collect support count for each candidate sequence
  - 2.3 Generate candidate length  $(k+1)$  sequence from length- $k$  frequent sequences using Apriori through the consideration of constraints C5
3. Repeat until no frequent sequence / no candidate can be found
4. Generate sequential rules that satisfy the support and confidence thresholds

Candidate generation algorithm:

**Input:** Frequent Set  $k-1$ , denoted as  $F(k-1)$

**Output:** Candidate Set, denoted as  $C$

1. Join  $F(k-1)$  with  $F(k-1)$
2. Get rid of infrequent sequence (prune)

**Note:** Order of items matter

Let us now illustrate how the algorithm works using the repository of activity sequences corresponding to the seven process models in Table II. We assume the min-support is set to 2. The process of how  $L_1$ -sequences are generated is shown in Figure 3. As can be seen, the table on the left shows all the candidate sequences of length 1 and their support count. Candidate sequences whose support count is less than the min-support (i.e. 2) are pruned. The table on the right in Figure 3 shows the frequent  $L_1$ -sequence, i.e. candidate sequences whose support count is greater or equal to the min-support.

Similarly, the subsequent passes over the repository of activity sequences give us  $L_2$ -sequence,  $L_3$ -sequence, and  $L_4$ -sequence candidates as shown in Figure 4.

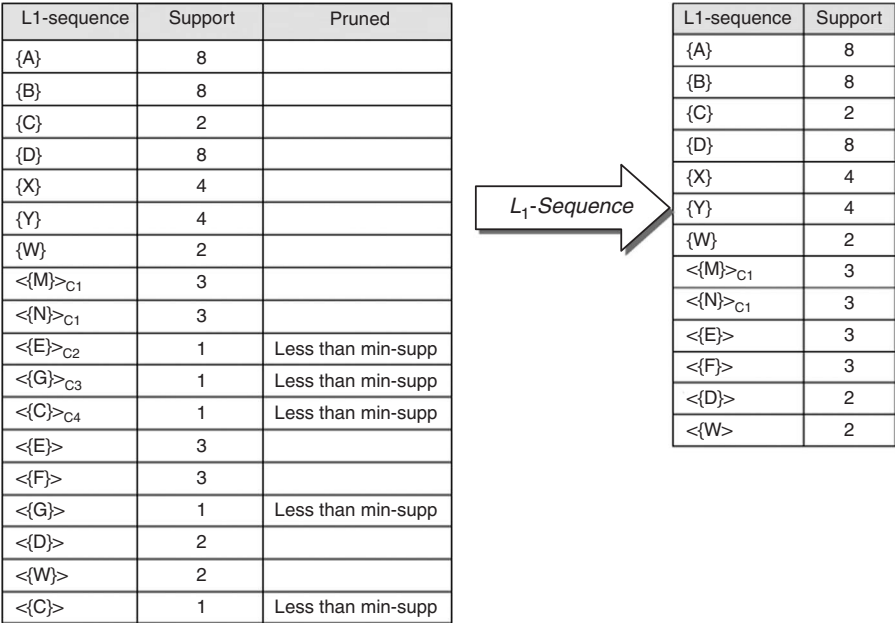
## 6. Empirical validation

In this section, we report on our experiments to validate the effectiveness of our approach. The experiments were done on two data sets. The first data set consists of the real-world process models, whereas the second data set has a large number of synthesized process models generated by a random process model generator. The details of the experiments are elaborated as follows.

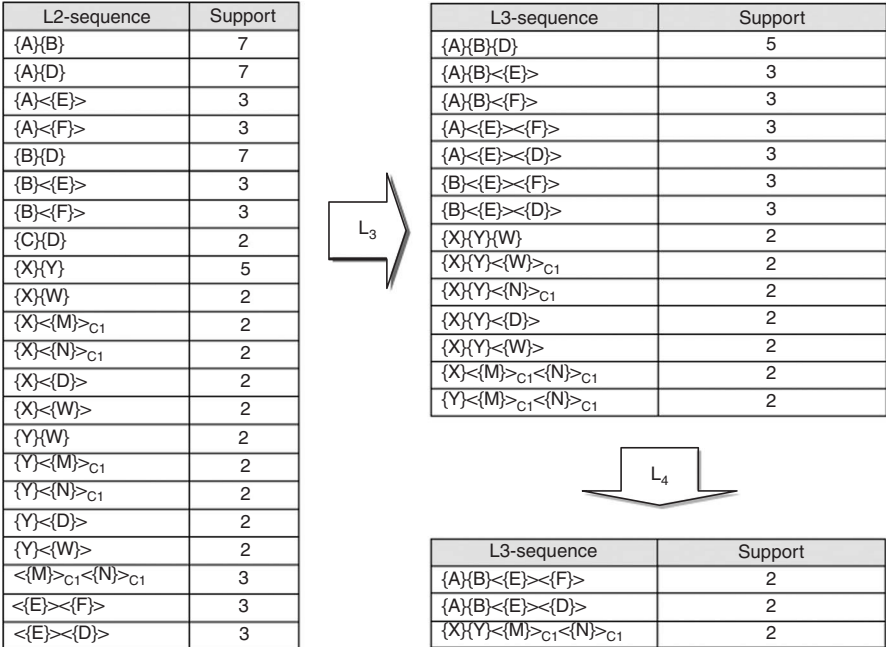
### 6.1 Experiments with large data sets of synthesized business process models

One of the most important questions that need to be addressed in the evaluation is: how can we assess whether the business rules we extracted are actually correct? This is

**Figure 3.**  
Generating  
 $L_1$ -sequence  
candidates



**Figure 4.**  
Generating  $L_2$ -  
sequence,  
 $L_3$ -sequence, and  
 $L_4$ -sequence  
candidates



important because it reflects the correctness/effectiveness of your approach. Although the previous experiment was done with real-life business process models, we were not able to answer this question due to the unavailability of business rules prior to our validation. Therefore, we have performed further experiments on large data sets of synthesized process models. Specifically, the design of our experiment is described as follows:

- (1) We developed a random process generator and use it to generate a very large number of process models. The generator will accept as parameters the number of tasks and the number of gateways which we are needed in each process model.
- (2) We will randomly generate between 20 business rules using the same vocabulary used for the process model. These business rules form the so-called ExpectedSet.
- (3) We will filter the process models generated in step 1 using the business rules generated in step 2 to obtain only those process models which comply with these rules.
- (4) We will then use our technique to mine these filtered process models and obtain business rules. The outcome of this step is the ActualSet of business rules.

Ideally, a perfect mining technique would produce an ActualSet which is equal to the ExpectedSet, meaning that the mined business rules are correct and complete (i.e. all possible business rules have been extracted). However, in practice the ActualSet may contain some rules that are not in the ExpectedSet and vice versa. Therefore, we use prevision/recall analysis to measure the performance of our approach. Recall and precision are used to measure how many of the mined rules are correct and how many of the correct rules are not mined. For example, let assume that the ExpectedSet contains two rules: { < a1→a4→a7→a8 > , < a10→a17→a100→a93→a20 > } and the ActualSet is { < a1→a4→a7→a8 > , < a10→a17 > }. Our approach gives one correct rule, one incorrect rule, and misses one rule:

$$\text{Recall} = \frac{\text{ExpectedSet} \cap \text{ActualSet}}{\text{Total\_of\_ExpectedSet}}$$

$$\text{Precision} = \frac{\text{ExpectedSet} \cap \text{ActualSet}}{|\text{ActualSet}|}$$

We have conducted the experiments with different sizes of data set: 400, 800, and 1,200 process models. For each data set, we have also tested with two different values of min-support (2 and 4). The results are summarized in Table III.

As can be seen from Table III, all the business rules extracted by our technique are expected (precision is 1 for all cases) and recalls are the same for both min-support of 2 and 4. These results have demonstrated the effectiveness of our approach in terms of extracting the correct business rules.

### 6.2 Experiments with a samll data sets of real business process models

The process models used in this experiment capture “as-is” work-flows. Each process was modeled based on a series of interviews with various actors within a financial organization. Furthermore, each model has been created in text as well as at various abstraction layers using BPMN. There are 31 process models in the collection. First, we transformed all the 31 process models into XMI format and then generated activity-sequences corresponding to those process models.

In the real world, it is possible that a set of process models can be produced by many process designers. Sometimes, given name of tasks/activities may be different although these tasks indicate to the same thing. It can be illustrated as follows. Consider two process models in Figure 5.

In Figure 5, the name of task “Generate a letter” in *P1* and the name of task “Generating a letter” in *P2* are different, if they are considered by an automatic analysis. However, these activities will be considered as the same activity by using the VL-ontology (Polpinij and Ghose, 2008).

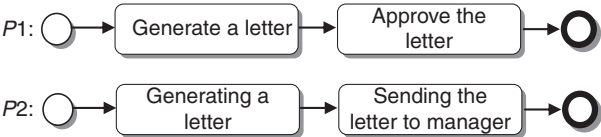
Many of the 30 business process models in our data set has sub-process, which give in total 325 (sub-)process models. It is noted that the min-support used in this experiment is 2. An experiment result can be shown as Figure 4. This figure shows some of activities and conditions (based on gateways) and some of the activity-sequences that are generated from the original process models. In addition, it shows the longest sequence that can be extracted from the collection. This sequence satisfies the support and confidence thresholds and it is considered as a business rule.

If we consider only business rules that contain at least four activities, we were able to obtain eight business rules (see Table IV). It is noted that there is no exact rule, since there is not any business rule having the confidence threshold as 1.00. In this context, the extracted business rules will be evaluated by considering the impact value. Each business rule will be used to retrieve and count the number of process models that contain this business rule. Figures 6 and 7.

The business rules extracted can be used for further analyzing, verifying (e.g. conformance checking), and improving process designs. More specifically, we can identify a set of key activities which are important for driving in process success as a set of business rules, this can lead to efficiencies in various ways such as reducing time of verification and improvement of process designs. For example, we found that the sequence of activities < DMU.Receive Change of Details form in mail > , < DMU.Process Form (Closed System) > , < DMU.Scan form into Staffware > , < DMU.Receive output letter from software system > are critical to the success of the process of dealing with new members. This is also correctly reflected the fact that business rule 1 (refer to Table I)

**Table III.**  
The experimental  
results of process  
model retrieval  
by using the  
actual rules

Min-support	Number of random process models	Recall (%)	Precision (%)	Time used (seconds)
2	400	71.64	100	50.76
	800	71.37	100	110.20
	1,200	72.29	100	190.20
4	400	71.64	100	60.16
	800	71.37	100	124.32
	1,200	72.29	100	200.32
Average of performance		71.77	100	122.66



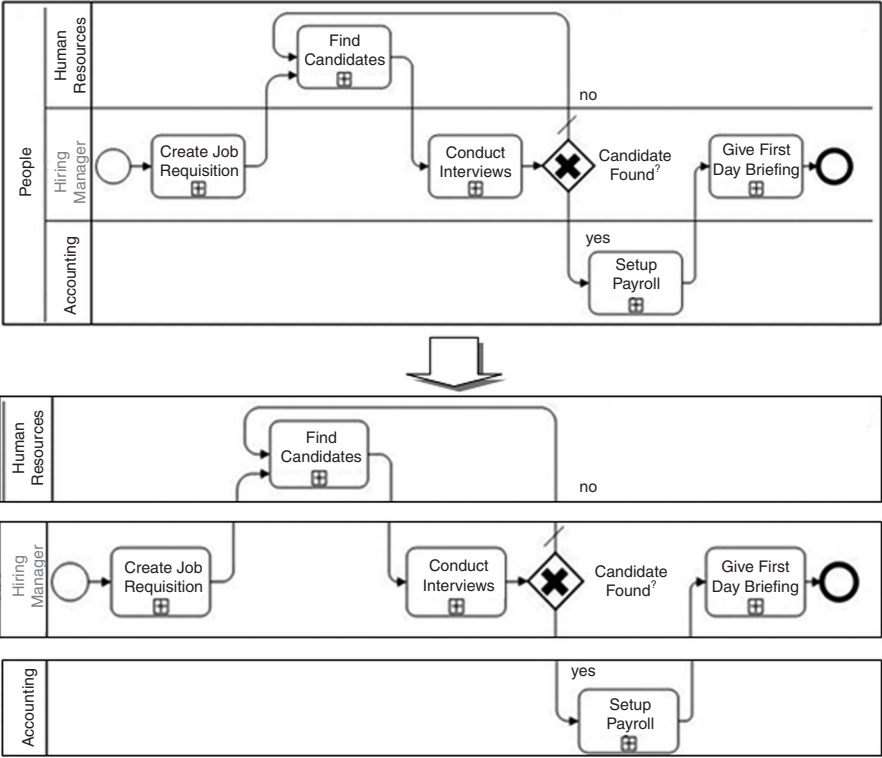
**Figure 5.**  
Activity with  
different name but  
the same semantic

Rule	Detail
Rule 1	< DMU.Receive Change of Details form in mail > < DMU.Process Form (Closed System) > < DMU.Scan form into Staffware > < DMU.Receive output letter from software system >
Rule 2	< Software.General Workflow.Form scanned into workflow system > < Software.General Workflow.Work flow case created and allocated > < Software.General Workflow.Staff memmer opens case in workflow system > < Software.Member Processing Workflow.Open Workflow case from member index workflow queue > < Software.Member Processing Workflow.Begin manual processing > < Software.Member Processing Workflow.Update case details >
Rule 3	< Processing Team.Open Case from workflow engine > < Processing Team.Verify that form is a change of personal details form > < Processing Team.Lookup form personal details in DBMS >
Rule 4	< Processing Team.Update the DBMS > < Processing Team.Updates case details with updates that have been made > < Processing Team.Save workflow case >
Rule 5	< Process.DMU Receives letter and scans into Staffware > < Process.Form is index to working queue in staffware > < Process.Workflow case is opened from employee queue > < Exclusive-gateway > < Process.Processing member opens form > < Exclusive-gateway (YES) > < Process.Employer profile created in DBMS > < Process.White new employer number on the top of the form > < Process.Update case detail in staffware and save > < Exclusive-gateway (YES) > < Process.Return form to cash receiving for processing >
Rule 6	< Process.DMU Receives letter and scans into Staffware > < Process.Form is index to working queue in staffware > < Process.Workflow case is opened from employee queue > < Exclusive-gateway > < Process.Processing member opens form > < Exclusive-gateway (NO) > < Process.Manual Letter sent to DMU to mail to employer >
Rule 7	< Processing Team.Ensure certified evidence has been attached > < Processing Team.Exclusive gateway(Was certified evidence attached? - YES) > < Processing Team.Open verified form and support documentation >
Rule 8	< Processing Team.Ensure certified evidence has been attached > < Processing Team.Exclusive gateway(Was certified evidence attached? - YES) > < Processing Team.Open verified form and support documentation > < Processing Team.Search DBMS for relevance employer information > < Processing Team.Exclusive gateway(Was the member found? - NO) > < Processing Team.Update details in DBMS >

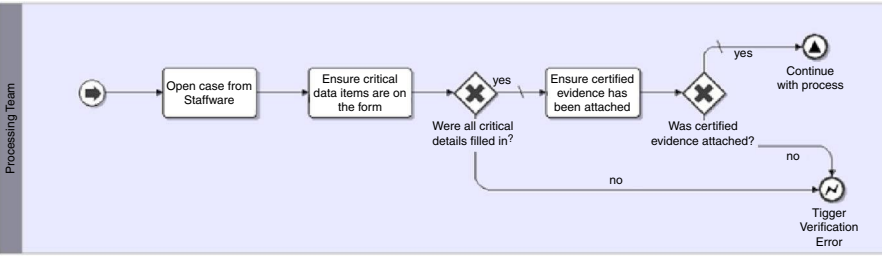
**Table IV.**  
The set of business  
rules extracted from  
our case study

was extracted from that process. In fact, discovered business rules can be used to verify against the intended policies and/or compliance requirements. For example, we could annotate each activity in a business rule with semantic effects and use them to check against compliance requirements.

In addition, in order to improve existing business processes, the business process designer should have adequate understanding of the relevant business rules. The discovered business rules can be used to classify process designs. Process designs enforcing the same business rules can be grouped in the same class. In our case study, we were able to classify 30 process models in the organization into four groups using the



**Figure 6.**  
Some process models  
used in our data set



**Figure 7.**  
An example of the  
original process  
model generated  
into the form of  
activity-sequences

P1 : {open case from Staffware}, {Ensure critical data items are on the from}, <{Ensure certified evidence has been attached}>  
Were all critical details filled in?=yes

P2 : {open case from Staffware}, {Ensure critical data items are on the from}, <{no activity}>  
critical details filled in?=no Were all

extracted business rules. For instance, the first group has five process models, all of which have business rules 1-4. Discovering and understanding the similarities among business processes can offer many benefits to organizations. The five business processes in the first group of our case study can be managed together, which facilitates business process maintenance.

In addition, business process models are essential knowledge assets for an organization (with hundreds and thousands of business process models ) to manage its business processes in terms of documenting and implementing procedures, control their execution, analyze their performance, and improve them (i.e. business process management). Organizations face a range of challenges in managing change in the context of large and complex collections of business processes. Making changes to large, complex repositories of business process models is a highly challenging task. This is mainly due to the ripple effect caused by a change. First, changes in business rules (due to changed policies and/or compliance requirements) need to be implemented into business processes. Since our approach is able to discover business rules from process models, it is also able to help identify exactly which business processes need changing when their associated business rules change. Second, a change made to one business process can potentially affect a range of other processes that are related to the process being changed. For example, changes initially made to a sub-process (e.g. adding a new activity) may lead to secondary, additional changes made to the processes that contain this sub-process. Such changes made to those processes may lead to further changes in other related processes. In a large repository of hundreds or even thousands of business process models, it becomes critical to determine the impact of a change. Using the business rules extracted from existing business process models, we are able to form inter-process relationships (e.g. processes need to enforce the same business rules) which are then used to support change impact analysis. Change impact analysis plays a major part in planning and establishing the feasibility of a change in terms of predicting the cost and complexity of the change (before implementing it). This helps reduce the risks associated with making changes that have unintended, expensive, or even disastrous effects on existing business operations.

## 7. Conclusion

The emphasis of BPR is to improve existing systems with a greater return of investment (ROI) than could be obtained through a new development. This is because BPR is the systematic transformation of an existing system into a new form to realize quality improvements in operation, system capability, functionality, performance, or evolvability at a lower cost, schedule, or risk to the customer. Many methods of BPR are proposed but these methods are driven on the concept of modification of business process or workflow (Wastell *et al.*, 1994; van der Aalst and Hee, 1995; Jarzabek and Ling, 1996; Muthu *et al.*, 1999; Thong *et al.*, 2003; Tavana *et al.*, 2012). However, business rule is considered as a critical component of BPR, since business rules define the active policy of an organization, an ability to identify, understand and leverage these rules is a key prerequisite to evolving an organization's IT infrastructure. The success of BPR projects usually have more to do with rethinking business rules, than with re-doing business process (or workflow). This is because business rules are more basic to the company (or enterprise, or organization) than business process. However, Business process can be improved everyday basis this is what staff and management should be aiming to do. In many organization, the head of staff might bring the staff together to discuss yesterday's problem/performance, including asking for suggestions for improvement. Therefore, if we use the business rules as the key component for BPR, a solution to quickly

archive the business rules may be required. As above, this paper present a new method of business rule extraction from business processes. Business rules that are automatically extracted by our approach represent significant and valid sequential correlations among business activities belonging to a particular organization.

Our work supports organizations in harvesting business rules from their existing business process models. There are many benefits from discovering business rules from business processes and make them explicitly available to relevant stakeholders in an organization. For example, the availability of business rules will allow us to have a better understanding of the operational process, hence making the maintenance and evolution of business processes easier. In addition, the discovered business rules can be used to develop new rules or modify the existing rules. Furthermore, the analysis of business rules may help an organization to better achieve their business goals and reduce costly mistakes. Individuals within an organization also benefit from well-documented business rules in terms of improving communication and comply with legal requirements.

The lack of any explicit modelling of business rules poses several major problems for an organization. Typically, the first implementation of business rules into business process design aligns well with the intended policies. However, the ever-changing business environment (due to various reasons such as new customer requirements, global competition pressures, new regulations, new IT solutions, economic down turn, etc.) demands organizations to constantly consider changing their business activities in order to remain competitive and sustainable in the long term. Hence, due to such frequent changes, there may be a gap between the actual execution of a process and what is being described in the process model. This phenomenon creates an inconsistent enforcement of business rules where the executed business rules tend to deviate from the intended one. As a result, business rules extracted from business process models (as done in our approach) may not necessarily correspond to the business rules that are currently enforced in business process execution. Future work involves improving the recall of our approach and investigating how to extract other useful information from repositories of business process models.

#### Note

1. [www.omg.org/technology/documents/format/xml.htm](http://www.omg.org/technology/documents/format/xml.htm)

#### References

- Aalst, W.M.P.V.D. (2011), *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, Springer Publishing Company, Springer-Verlag Berlin Heidelberg.
- Aalst, W.M.P.V.D. and Hee, K.M.V. (1995), "Framework for business process redesign", in Callahan, J.R. (Ed.), *Proceedings of the 4th Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Berkeley Springs, West Virginia.
- Agrawal, R. and Srikant, A. (1995), "Mining sequential patterns", *Proceedings of the Eleventh International Conference on Data Engineering (ICDE)*, 6-10 March, Taipei.
- Hill, J.B., Cantara, M., Deitert, E. and Kerremans, M. (2007), "Magic quadrant for business process management suites", Tech. rep., Gartner Research.
- Jarzabek, S. and Ling, T.W. (1996), "Model-based support for business re-engineering", *Information and Software Technology*, Vol. 38 No. 5, pp. 355-374.
- Larose, D.T. (2005), *Discovering Knowledge in Data: An Introduction to Data Mining*, Wiley-Interscience, London.
- Miers, D. and White, S.A. (2008), *BPMN Modeling and Reference Guide*, Future Strategies Inc, FL.



- Morgan, T. (2002), *Business Rules and Information Systems: Aligning IT with Business Goals*, Addison Wesley Professional, London.
- Muthu, S., Whitman, L. and Cheraghi, S.H. (1999), "Business process reengineering: a consolidated methodology", *Proceedings of The 4th Annual International Conference on Industrial Engineering Theory, Applications, and Practice*.
- Polpinij, J., Ghose, A. and Dam, H.K. (2010), "Business Rules Discovery from Process Design Repositories", *6th World Congress on Services*.
- Polpinij, J. and Ghose, A.K. (2008), "An ontology-based sentiment classification methodology for online consumer reviews", *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pp. 518-524.
- Ross, R.G. (1997), "Business rule re-engineering", *Data Base Newsletter*, Vol. 25 No. 2, pp. 25-32.
- Ross, R.G. (2003), *Principles of the Business Rule Approach*, Addison Wesley Professional, Boston, MA.
- Shao, J., Embury, S.M., Fu, G., Liu, X. and Gray, W.A. (2002), "Unlocking business rules for maintaining information systems", *The Proceedings of the 1st International Workshop on Database Maintenance and Re-engineering (DBMR)*, pp. 17-30.
- Sneed, H.M. and Erdos, K. (1996), "Extracting business rules from source code", *The Proceedings of IEEE 4th International Workshop on Program Comprehension (IWPC)*, pp. 240-247.
- Tavana, M., Sisti, A.F. and Trevisani, D.A. (2012), "An automated workforce clustering method for business process reengineering in research and development organizations", *International Journal of Information Technology Project Management (IJITPM)*, Vol. 3 No. 4, pp. 1-20.
- Thong, J.Y.L., Yap, C.S. and Seah, K.L. (2003), "A consolidated methodology for business process reengineering", *International Journal of Computer Applications in Technology*, Vol. 17 No. 1, pp. 1-15.
- Van Der Aalst, W.M.P., Rubin, V., Verbeek, H.M.W., Van Dongen, B.F., Kindler, E. and Günther, C.W. (2008), "Process mining: a two-step approach to balance between underfitting and overfitting", *Software and Systems Modeling*, Vol. 9 No. 2, pp. 87-111.
- Van Der Aalst, W.M.P. and Weijters, A.J.M.M. (2004), "Process mining: a research agenda", *Computers in Industry*, Vol. 53 No. 3, pp. 231-244.
- Wang, X., Sun, J., Yang, X. and He, Z. (2004), "Business rules extraction from large legacy systems", *Proceedings of Eighth European Conference on Software Maintenance and Reengineering (CSMR)*, 24-26 March, Tampere.
- Wastell, D.G., White, P. and Kawalek, P. (1994), "A methodology for business process redesign: experiences and issues", *The Journal of Strategic Information Systems*, Vol. 3 No. 1, pp. 23-40.
- Ye, J., Coyle, L., Dobson, S. and Nixon, P. (2007), "Ontology-based models in pervasive computing systems", *The Knowledge Engineering Review*, Vol. 22 No. 4, pp. 315-347.
- Yu, F. and Mylopoulos, J. (1994), "Using goals, rules, and methods to support reasoning in business process reengineering", *Proceeding of 27th Hawaii International Conference on System Science*, 4-7 January, Maui, Hawaii.

### Further reading

- Barothy, T., Peterhans, M. and Bauknecht, K. (1995), "Business process reengineering: emergence of a new research field", *SIGOIS*, Vol. 16 No. 1, pp. 3-10.
- Damij, T. (2003), "Using an object-oriented methodology called tad in business process reengineering", *Journal of Object Technology*, Vol. 2 No. 2, pp. 151-168.

- Hammer, M. and Champy, J. (1993), *Reengineering the Corporation*, Harper Collins, New York, NY.
- Johnson, S.C. (1967), "Hierarchical clustering schemes", *Psychometrika*, Vol. 2 No. 3, pp. 241-254.
- Lang, P. and Obermair, W. (1997), "Medeling business rules with situation/activation diagrams", *The Proceedings of 13th International Conference on Data Engineering*, 7-11 April, Bermingham.
- May, M., Berendt, B., Cornuéjols, A., Gama, J., Giannotti, F., Hotho, A., Malerba, D., Menesalvas, E., Morik, K., Pedersen, R., Saitta, L., Saygin, Y., Schuster, A. and Vanhoof, K. (2008), *Research Challenges in Ubiquitous Knowledge Discover*, Chapman & Hall, London.
- Olalla, M.F. (1999), "Information technology in business process reengineering", *The 47th International Atlantic Economic Conference*, March, Vienna.

#### About the authors

Jantima Polpinij is currently a Lecturer working for the Department of Computer Science, Faculty of Informatics, Mahasarakham University. Her research skills are data mining and knowledge discovery in database, information retrieval, natural language processioning, and business process management. Jantima Polpinij is the corresponding author and can be contacted at: jantima.polpinij@gmail.com

Professor Aditya Ghose received the PhD Degree in Computer Science from the University of Alberta, Canada. He is currently a Professor at the School of Computer Science and Software Engineering, University of Wollongong. His research interests include services, software engineering, supply chain management, and business process management.

Dr Hoa Khanh Dam received the PhD Degree in Computer Science from the RMIT University in Australia. He is currently a Lecturer at the School of Computer Science and Software Engineering, University of Wollongong. His research interests include software engineering, service-oriented computing, and business process management.

**This article has been cited by:**

1. Marie-Sophie Denner, Maximilian Röglinger, Theresa Schmiedel, Katharina Stelzl, Charlotte Wehking. How Context-Aware Are Extant BPM Methods? - Development of an Assessment Scheme 480-495. [[Crossref](#)]