Change visualisation

Low, W.Z.; van der Aalst, W.M.P.; ter Hofstede, A.H.M.; Wynn, M.T.; De Weerdt, J.

Published in:
Information Systems

DOI:
10.1016/j.is.2016.10.005

Published: 01/04/2017

Information Systems 65 (2017) 106–123

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/is

Change visualisation: Analysing the resource and timing differences between two event logs

W.Z. Low[a,*], W.M.P. van der Aalst[b,a], A.H.M. ter Hofstede[a,b], M.T. Wynn[a], J. De Weerdt[c]

[a] Queensland University of Technology (QUT), 2 George Street, Brisbane, QLD 4000, Australia
[b] Department of Mathematics and Computer Science, Technische Universiteit Eindhoven (TU/e), 5612 AZ Eindhoven, The Netherlands
[c] Department of Decision Sciences and Information Management, KU Leuven, Oude Markt 13, 3000 Leuven, Belgium

ARTICLE INFO

Keywords:
Event log comparison
Comparative process visualisation
Visual analytics

ABSTRACT

With organisations facing significant challenges to remain competitive, Business Process Improvement (BPI) initiatives are often conducted to improve the efficiency and effectiveness of their business processes, focussing on time, cost, and quality improvements. Event logs which contain a detailed record of business operations over a certain time period, recorded by an organisation's information systems, are the first step towards initiating evidence-based BPI activities. Given an (original) event log as a starting point, an approach to explore better ways to execute a business process was developed, resulting in an improved (perturbed) event log. Identifying the differences between the original event log and the perturbed event log can provide valuable insights, helping organisations to improve their processes. However, there is a lack of automated techniques and appropriate visualisations to detect the differences between two event logs. Therefore, this research aims to develop visualisation techniques to provide targeted analysis of resource reallocation and activity rescheduling. The differences between two event logs are first identified. The changes between the two event logs are conceptualised and realised with a number of visualisations. With the proposed visualisations, analysts are able to identify resource- and time-related changes that resulted in a cost reduction, and subsequently investigate and translate them into actionable items for BPI in practice. Ultimately, analysts can make use of this comparative information to initiate evidence-based BPI activities.

1. Introduction

Business Process Improvement (BPI) is concerned with identifying opportunities for business process redesign bearing in mind the potential impact that these redesign actions may have on different dimensions such as time, cost, quality and flexibility [1]. It is essential for organisations to constantly engage in ways to lower the cost and improve the efficiency of their business processes; hence the emphasis on BPI initiatives within organisations. However, the technical challenges of improving a business process are arduous. By utilising and analysing historical execution data, evidence-based approaches to back BPI initiatives are becoming more prevalent, with support from methodologies such as Six-Sigma [2], Lean Thinking [3], Kaizen [4], and others. Process mining techniques further promote BPI initiatives by providing support for analysis of business process execution trails recorded in event logs [5]. Because of the fact that these techniques make use of actual execution data, this support is evidence-based. Although a significant number of studies and implementations have been carried out in the areas of BPI and process mining [6], there are only a few studies that have looked into identifying business process inefficiencies from a cost perspective.

In an attempt to “improve the history”, Low et al. propose an approach to identifying potential efficiency gains in business operations by observing how they are carried out in the past, and then exploring better ways of executing them [7]. A generic cost structure assigns costs to the alternative execution scenarios, considering the trade-offs between time, cost, and resource utilisation. This approach takes the original execution history of a business process (event log) as an input, heuristically introduces cost-reducing perturbations, and produces an alternative execution scenario (perturbed event log).

Often the cost of the alternative execution scenario is significantly lower compared to the original event log [7]. In addition, the performance of the two execution scenarios varies. Analysts want to, with minimal effort, know what has changed, or what the differences between the two execution scenarios are, with the aim of attributing these execution discrepancies to changes in process performance.

* Corresponding author.
E-mail addresses: wz.low@hdr.qut.edu.au (W.Z. Low), w.m.p.v.d.aalst@tue.nl (W.M.P. van der Aalst), a.terhofstede@qut.edu.au (A.H.M. ter Hofstede), m.wynn@qut.edu.au (M.T. Wynn), jochen.deweerdt@kuleuven.be (J. De Weerdt).

http://dx.doi.org/10.1016/j.is.2016.10.005
Received 3 November 2015; Received in revised form 25 April 2016; Accepted 18 October 2016
Available online 21 October 2016

However, it is highly challenging to identify the differences between the original and the alternative execution scenarios. Current techniques only allow event logs to be analysed individually, whereas the analysis and visualisation of multiple event logs are done manually. Proper visualisation can promote the delivery and understanding of information [8], in this case, information concerning multiple event logs. However, to the best of our knowledge, there is a lack of studies that look into analysis of multiple event logs. Without a proper approach to identify and portray the differences between two event logs, potential process improvement opportunities may be missed.

Therefore, this paper aims to identify and visualise the differences between two event logs, providing insights into the changes in business process performance. Knowledge from the field of visual analytics is leveraged to propose visualisation environments that enable process analysts to identify and analyse the differences. The key differences, in this case, the resource allocations and the start and end times of activities, are illustrated using visualisations. The perceived usefulness and the user acceptance of the visualisations are assessed through an empirical user evaluation. With proper analysis and visualisation techniques in place, stakeholders are provided with insights in a more effective and efficient manner to better facilitate BPI initiatives.

The remainder of this paper is organised as follows. Related work is reviewed and discussed in Section 2. In Section 3, the fundamentals of this work, the concepts of resource reallocation and activity rescheduling, as well as the visualisation concepts, are introduced. Section 4 discusses the design principles, an illustrative example, and the realisations of the visualisation concepts. Section 5 discusses the user evaluation and its results. Section 6 concludes this paper and outlines potential future work.

2. Related work

Within the field of Business Process Management (BPM), the concepts of Business Process Redesign (BPR) and Business Process Improvement (BPI) are highly relevant to researchers and practitioners. A number of case studies have looked into creating a framework to list and classify best practices to facilitate BPR within organisations [9,10]. In [1], a number of BPR best practices and approaches are provided. BPR has also been applied and evaluated via case studies carried out in organisations from various fields [11,12]. An approach that uses performance measures to quantify the impact and trade-offs of BPR actions on all dimensions of workflow performance has also been developed [13].

Process mining facilitates continuous BPI and BPR activities by extracting knowledge from event logs [5,14,15]. An event log is a data store where a historical record of process execution is kept. Information such as events (offer, start, complete), data attributes, utilised resources and task durations can be extracted from an event log [16]. Process mining techniques can provide valuable insights into control flow dependencies, data usage, resource utilisation and various performance related statistics. Process mining is enabled through the facilitation of software. One of the leading tools for process mining is the ProM framework [5]. ProM provides a generic open-source framework for process mining and analysis tools in a standard environment, and plug-ins can be added to extend its functionalities [17].

A number of studies to improve business processes have utilised process mining techniques. Process mining facilitated the analysis of invoice processing from process, organisational, and case perspectives [5]. By analysing and learning from past business process executions, recommendations on possible next steps can be provided to the users considering specific optimisation goals [18]. Business process models can also be discovered by mining event logs [19]. Drawing inspiration from the well-known OLAP (Online Analytical Processing) data cubes, the notion of “process cubes” was introduced in [20]. Each cell in the process cube corresponds to a set of events and can be used to discover a process model, to check conformance with respect to some process model, or to discover bottlenecks [21]. A number of works revolve around comparing different representations of a business process. Techniques to compare two process models based on their observed control-flow behaviour have been proposed [22]. Process mining can also be used to measure the conformance between event logs and process models [23,24]. A generic framework for re-engineering event logs in a controlled manner has been introduced [25]. Yet, techniques to compare and analyse two event logs are still in their infancy.

The analysis of event logs involves potentially a huge amount of data, which requires the support of specialised techniques. Visual analytics is a multidisciplinary field that aims to enable people or organisations to process and extract insights from vast amounts of data, and communicate results so that strategic actions can be taken [8]. Shneiderman first introduced “overview first, zoom and filter, then details-on-demand” as the visual information seeking mantra, to be used as a starting point for portraying information using advanced graphical user interfaces (visualisation) [26]. Keim et al. then improvised a visual design guideline of “analyse first, show the important, zoom, filter and analyse further, and details-on-demand”, bringing the focus towards visual analytics [8]. This mantra clearly stated that visual analytics is a field that combines both analytical approaches and visualisation techniques [8]. The decision-making performance is heavily dependent on what the problem is and how the problem is represented; it follows that the cognitive fit between the problem and its representation is extremely important [27,28]. Andrienko and Andrienko provided a general outline of how to explore, analyse and visualise spatial and temporal data [29]. More specifically, in terms of resource and timing information, social network graphs are commonly used to illustrate relationships [30,31], whereas charts are useful for describing time-related information [32,33].

Although a majority of the studies in the area of information visualisation do not focus on the business process perspective, visual analytics has been and is increasingly influencing the field of business process improvement and process mining [34–37]. As process mining techniques analyse huge amounts of business process information, the event data needs to be presented appropriately to be meaningful. Social Network Analysis (SNA), a collection of methods, techniques, and tools in sociometry for the analysis of social networks [38], has been used by several process mining techniques to facilitate their analysis of sociometry [5,39,40]. Dotted charts have also been used to facilitate log analysis, where each row corresponds to one of the cases, laid out on a time scale [36]. PPM (process of process modelling) chart, an extension of the dotted chart, was used to visualise the steps to create a process model [41]. A number of visualisations have also been proposed to analyse large numbers of event sequences [42–45]. MatrixWave [45] was proposed by Zhao et al. to visualise and compare traffic patterns of event sequences, however, the temporal perspective is not taken into account. LiteFlow [42], Outflow [43], and Frequence [44] were proposed to visualise and aggregate temporal event sequences, with the latter two visualisations accentuating event sequence outcomes as well. Nonetheless, only the temporal perspective is given priority, thus insights into the resource perspective are lacking. Moving away from event sequences, de Leoni et al. made use of both map and movie metaphors to visualise work items and resources assignment in a sophisticated manner [35]. In addition to visualising event data meaningfully, a number of studies looked into analysing and visualising the performance of business processes. An alignment-based framework with visualisation has facilitated the conformance checking of process models [34]. The performance evaluation of collaborating resources in a business process can be facilitated by visualisations as well [36]. The prediction of completion time of running instances using discovered process models has also been facilitated by a number of visualisations [37]. Several works have investigated visualisation approaches to highlight the differences and commonalities between business processes. Kriglstein et al. investigated visualization requirements for process changes [46], as well as a visualization approach to highlight

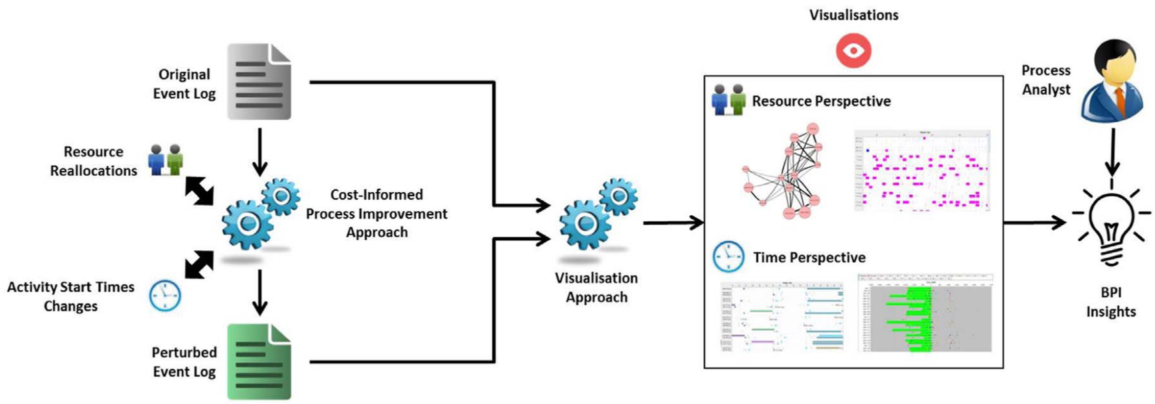


Fig. 1. Overview of the approach to explore less expensive business process execution scenarios, and gain insights by visualising the changes/differences between the two event logs.

the instance traffic of two process models [47]. The behaviours of organisations were also compared and illustrated using an alignment matrix visualisation [48]. Yet, there is a lack of studies on how visualisations can be utilised to compare two event logs and provide meaningful insights.

3. Change visualisation approach

The proposed process improvement approach begins with changing and improving the history of a process (via cost-reduction), and subsequently visualising the differences between the original event log and the perturbed event log. Fig. 1 illustrates the overall process improvement approach.

A hybrid genetic algorithm-based approach was developed to explore less expensive ways (in terms of a cost function, that includes various quality dimensions) to execute a business process [7]. By utilising an event log, a number of key characteristics of the process are kept the same (such as the activities performed and their durations, and the arrival times of cases), while other elements within the event log (such as resource allocations and start times of activities) are perturbed in order to explore different execution scenarios, taking into account the working hours of resources. The notion of cost is applied to time and resource efficiency measures. For example, cases that finish earlier and use resources efficiently are desired, hence less costly. These cost notions are captured by a robust cost structure, which is then used as an objective function to determine the fitness of the execution scenarios in terms of process-related cost. We take into consideration cost-informed trade-offs between multiple aspects such as case duration, activity execution, and resource utilisation. Table 1 shows examples of such cost functions. The hybrid genetic algorithm-based approach explores different execution scenarios by perturbing the resource allocations and activity start times. This results in a perturbed event log, where the process has been optimised. We refer the reader to [7] for further details.

The emphasis of this paper is on how to visualise the differences between the original event log and the perturbed event log (that is, change visualisation). With both event logs on hand, this work analyses certain process characteristics by comparing and contrasting two event logs. The concepts of *resource swap* and *time shift* are proposed to analyse the differences between two event logs. In addition, a number of visualisations are introduced and implemented to better portray the two concepts. These visualisations take into account some common design principles from the field of visual analytics [8]. Categories are depicted using appropriate colour schemes and shapes. The magnitude of differences is represented by the size, weight and length of the elements. The positioning of elements, as well as the usage of bar charts with timeline axis, may illustrate spatial and temporal data. The design principles are discussed in detail in Section 4.

In the remainder of this section, the fundamentals are introduced and formalised in Section 3.1. The changes from a resource perspective, namely the reallocation of resources, the change in resource utilisations, and the changes in busy and idle times for resources, are analysed in Section 3.2. Lastly, Section 3.3 discusses the changes from a time perspective, including effects of activity (task instances) rescheduling on cases and tasks, as well as the changes in busy and idle times for cases.

3.1. Fundamentals

The starting point of this approach is an event log. An event log consists of a collection of process instances (*cases*). For each case, there is a sequence of *activities*. An activity is an instance of a *task*. We require that each activity has a *start time* and an *end time*. Each activity is executed by a *resource*. Table 2 illustrates a fragment of an event log.

Definition 1 (***Event Log***). Given $TS = \mathbb{R}$ is the set of possible timestamps, and $Dur = \mathbb{R}_1^+$ is the set of durations, $L = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ is an event log where:

Table 1
Example of cost functions defined in the cost structure used by the cost-informed process improvement approach.

Cost Type	Property	Value	Cost Function
Case	Damage Type & Case Duration Case Duration	Windscreen [SLA Breach]	$40 per hour $1500 if it takes more than 5 days, and $500 for every subsequent day after that
Activity	Activity & Resource Activity & Resource	CAR & A1 [Over-qualified Resources]	$100 per invocation $1000 per invocation
Resource	Resource Utilisation Resource Utilisation Resource Utilisation	Between 0 and 0.15 (under-utilised) Between 0.75 and 0.85 (optimum utilisation) Higher than 0.9 (over-utilised)	$45 per minute $1 per minute $20 per minute

Table 2
A possible fragment of the car insurance claim event log in chronological order, where each line is an activity instance. (CAR=Create Assessment Report; RAR=Review Assessment Report).

Case ID	Activity	Start Time	End Time	Resource	...
1	CAR	10/06/13 09:31:00	10/06/13 09:39:00	A5	...
1	RAR	10/06/13 09:42:00	10/06/13 10:00:00	IS2	...
2	CAR	10/06/13 09:45:00	10/06/13 09:50:00	A1	...
2	RAR	10/06/13 09:55:00	10/06/13 10:10:00	IS1	...
...	...	...	...	...	...

- C is a set of cases,
- $\mathcal{A}$ is a set of activities (task instances),
- $\mathcal{T}$ is a set of tasks,
- $\mathcal{R}$ is a set of resources,
- the sets C, $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{R}$ are pairwise disjoint and finite,
- $case \in \mathcal{A} \to C$ is a surjective function mapping activities to cases,
- $task \in \mathcal{A} \to \mathcal{T}$ is a surjective function mapping activities to tasks,
- $res \in \mathcal{A} \to \mathcal{R}$ is a surjective function mapping activities to resources,
- $art \in C \to TS$ is a function specifying the arrival time of cases such that,

$$art(c) = \min_{\substack{a \in \mathcal{A},\\ case(a)=c}} st(a), for\ all\ c \in C,$$

- $st \in \mathcal{A} \to TS$ assigns a start time to each activity and $et \in \mathcal{A} \to TS$ assigns an end time to each activity such that:
  - $\forall_{a \in \mathcal{A}}\ st(a) \geq art(case(a))$, and
  - $\forall_{a \in \mathcal{A}}\ et(a) > st(a)$.

  For convenience, we define the activity sets:

- $\mathcal{A}^c = \{a \in \mathcal{A} | case(a) = c\}$, for all $c \in C$,
- $\mathcal{A}^t = \{a \in \mathcal{A} | task(a) = t\}$, for all $t \in \mathcal{T}$, and
- $\mathcal{A}^r = \{a \in \mathcal{A} | res(a) = r\}$, for all $r \in \mathcal{R}$.

In reality, most, if not all resources involved in business processes are not available at all times. Working hours, annual leave, sick leave, and so on will affect the availability of the resources. In order to accurately calculate the time taken by a resource to execute an activity, the working hours of resources are taken into consideration.

Definition 2 (***Interval***). An interval $Intv = \{(z, z') \in TS \times TS | z < z'\}$ consists of a pair of timestamps, with z denoting the interval's start time, and z' denoting the interval's end time. The end time of an interval must be later than its start time.

Definition 3 (***Working Hours***). Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, \text{task}, \text{res}, art, st, et)$ be an event log, WH: $\mathcal{R} \to \mathcal{P}(Intv)$ is a set of intervals denoting the valid working hours of resources (where resources are allowed to execute activities), such that:

- $stw \in Intv \to TS$ denotes the starting time of the working hour, i.e. $stw(z, z') = z$, and $etw \in Intv \to TS$ denotes the end time of the working hour, i.e. $etw(z, z') = z'$,
- for all $r \in \mathcal{R}$, and all $p_1, p_2 \in WH(r)$, $p_1 = p_2$, or $stw(p_2) > etw(p_1)$, or $stw(p_1) > etw(p_2)$ (there should be no overlapping of working hours for each resource).

Additional process-related information, such as activity durations, case end times, and case throughput times, can be derived from information present in an event log. The temporal functions are defined as follows.

Definition 4 (***Temporal Functions***). Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, \text{task}, \text{res}, art, st, et)$ be an event log and WH be the set of valid working hours of resources, the following are some derivable temporal functions from an event log where:

- $dur_L^{WH} \in \mathcal{A} \to Dur$ is a function mapping activities to durations, taking into account the working hours of resources:
  - $dur_L^{WH}(a)=$

$$\sum_{p \in WH(res(a))} dur(st(a), et(a), stw(p), etw(p)),$$

where $dur(st(a), et(a), stw(p), etw(p))=$
  - 0, if $et(a) \leq stw(p) \vee etw(p) \leq st(a)$,
  - $et(a) - st(a)$, if $st(a) \geq stw(p) \wedge et(a) < etw(p)$,
  - $etw(p) - st(a)$, if $etw(p) \leq et(a) \wedge stw(p) < st(a) \wedge etw(p) > st(a)$,
  - $et(a) - stw(p)$, if $stw(p) \geq st(a) \wedge etw(p) > et(a) \wedge stw(p) < et(a)$, and
  - $etw(p) - stw(p)$, if $stw(p) > st(a) \wedge etw(p) < et(a)$,

  for all $a \in \mathcal{A}$,
- $cet_L \in C \to TS$ is a function specifying the end time of a case, where $cet_L(c) = \max_{a \in \mathcal{A}^c} et(a)$, for all $c \in C$,
- $tpt_L \in C \to Dur$ is a function assigning throughput times to cases, where $tpt_L(c) = cet_L(c) - art(c)$, for all $c \in C$.

The actual duration of an activity is calculated based on the amount of time it is being executed by a resource during its working hours. A case is deemed as finished once the last activity within the case is completed. The throughput time of a case is the duration between the case arrival and end time.

A perturbed event log is an event log where its resource allocations, and activity start and end times are modified, while everything else remains unchanged. A perturbed event log is defined as follows.

Definition 5 (***Perturbed Event Log***). Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, $\overline{L} = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ is a perturbation of L where st', et', and res' are modified, and $\overline{L}$ is an event log.In addition, we define:

- $\overline{\mathcal{A}}^r = \{a \in \mathcal{A} | res'(a) = r\}$, for all $r \in \mathcal{R}$ (resource allocation for activities in the perturbed event log), and
- $\mathcal{A}^\Delta = \{a \in \mathcal{A} | res(a) \neq res'(a)\}$ (activities where their resources were reallocated and replaced).

3.2. Changes from the resource perspective

A *resource swap* is characterised by the reallocation of activities from one resource to another. An activity that has been reallocated from one resource to another is called a *swapped* activity. Likewise, an activity that was not reallocated is called a *stable* activity. Resource swaps might influence the utilisation of resources, the execution cost, and the performance of a business process. While improvements in utilisation and cost may be obvious, the actual perturbations of resource allocations that produce these results are not evident. Inspired by the research studies in the area of BPI [1,10,11,13,49,50] and reinforced by our research team's experience in the area of BPM[1], we propose to consider resource-related questions such as:

- How many resource swaps have occurred?
- Which activities had their resources reallocated?
- Which resources got swapped the most?
- Which cases and tasks are involved, and how many resource swaps have occurred for a particular case or task?
- Which activities are the most stable (the least resource swaps occurred)?
- What are the utilisation levels of the resources, and are there any changes in resource utilisation?
- Are there any changes to the times when resources are busy or free?

Business process improvement actions can be initiated from potential insights gathered by interpreting the answers to the questions

[1] http://bpm-research-group.org/research/rss-publications/

above. For instance, sub-optimal resource allocations can be identified by investigating the resource swap frequency and patterns, allowing stakeholders to devise better resource allocation strategies. Knowledge gained from studying idle patterns of resources can also instigate innovative process redesign activities to complement the organisation's resource behaviours.

To answer these questions, the three main areas that could be affected by a resource swap are formalised and represented via visualisation concepts. These areas are 1) the resources, tasks, and cases that are involved in the reallocation of resources; 2) the differences in resource utilisations; and 3) the change in resource busy and idle times.

3.2.1. *Change in resource allocations*

The direct result of resource swaps is the change in resource allocations for activities. Resource swaps can be analysed from a number of *perspectives*. It is interesting to not only analyse 1) the reallocation of activities from one resource to another, but also 2) the reallocation of activities from one resource to another within a case, and 3) the reallocation of activities of different tasks from one resource to another.

In addition, analysing resource swaps using different *resource allocation views* could yield interesting insights. The different resource allocation views include 1) resource allocation in the original event log; 2) resource allocation in the perturbed event log; 3) stable activities in both original and perturbed event logs; 4) all resource allocations in both original and perturbed event logs; 5) swapped activities in the original event log; and 6) swapped activities in the perturbed event log. To realise the analysis of resource swaps from different perspectives and viewpoints, a weighted digraph (weighted directed graph) is chosen because its graphical properties can be used to characterise relationships between multiple entities [30].

Three resource swap weighted digraphs are conceptualised in accordance with the three perspectives (resource, case, and task), where the representation of their graphical elements will change depending on the graph perspective and the resource allocation view. Fig. 2 shows a resource swap weighted digraph illustrating the reallocation of activities from one resource to another. Fig. 3 conceptualises the resource swaps from a case perspective where an activity within a case has been reallocated from one resource to another, whereas Fig. 4 shows the resource swaps from a task perspective where an activity of a certain task has been reallocated from one resource to another. The graph elements are described in detail as follows:

- node – represents activities executed by a resource (i.e., a 'resource' node) or activities executed within a case (i.e., a 'case' node) or activities of a task (i.e., a 'task' node). The nodes are labelled with respective identifiers for resources, cases, or tasks;
- node size – represents the number of activities associated with a given node. The size of a resource node can be configured to show one of six different resource allocation views;
- pie-in-node – categorises the activities represented within a resource-node into their corresponding tasks, illustrating the proportion of tasks executed by a resource;
- directed edge – depending on the source and target nodes:
  - a directed edge between two resource nodes represents the reallocation of activities from one (source) resource to another (target resource);
  - a directed edge from a resource node to a case node represents the activities in the (target) case had reallocated from another resource to the (source) resource;
  - a directed edge from a case node to a resource node represents the activities in the (source) case had reallocated from the (target) resource to another resource;
  - a directed edge from a resource node to a task node represents the activity instances of the (target) task had reallocated from another resource to the (source) resource;
  - a directed edge from a task node to a resource node represents the activity instances of the (source) task had reallocated from the (target) resource to another resource; and

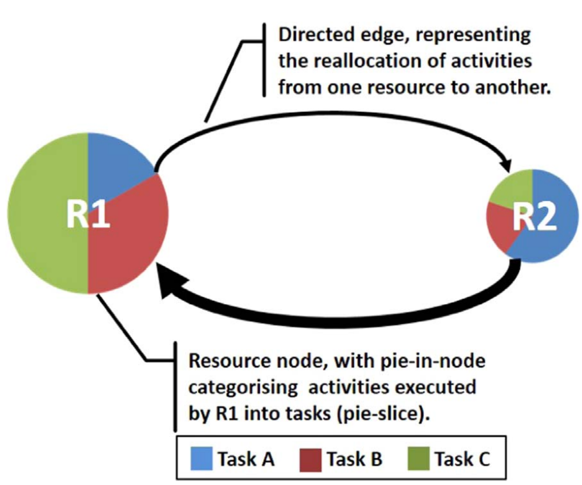


Fig. 2. A graph illustrating the reallocation of activities from one resource to another.

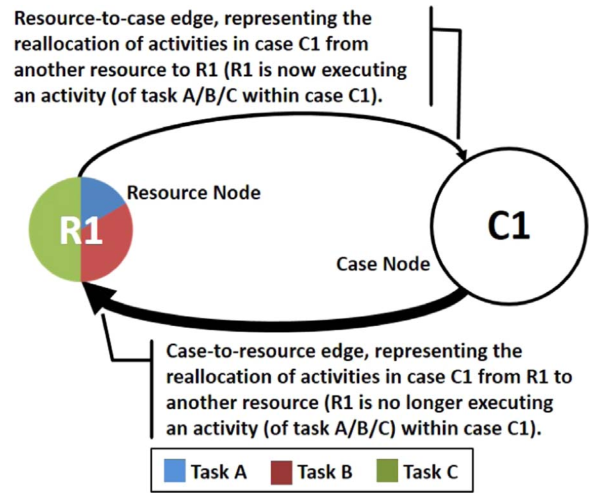


Fig. 3. A graph illustrating the reallocation of activities from one resource to another within a case.

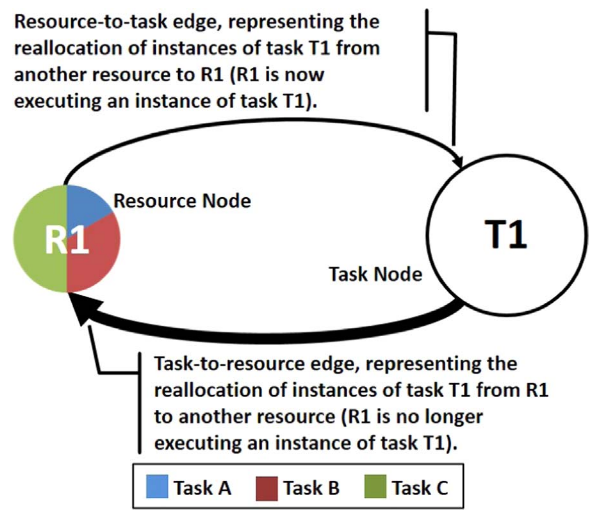


Fig. 4. A graph illustrating the reallocation of activities of different tasks from one resource to another.

- edge weight (thickness) – illustrates the number of reallocations as represented by the directed edge.

With the support of appropriate design rationales and interactivity, potential observations and insights that could be gained from these visualisation types include:

- the set of tasks that a particular resource or role is authorised to execute (the resource node's pie slices, using a segmented colour scale for ease of task identification);
- the stable and swapped activities (the nodes and the edges, where their magnitude can be derived by observing the node size and edge weight);
- resource behaviours, where, for example, a particular activity (or task or case) has a higher retention rate with a particular resource (the proportion of the pie slices of a resource node); and
- reallocation patterns between resources, cases, and tasks.

To facilitate the definition of a resource swap weighted digraph, the graph perspectives and resource allocation views are characterised using two configuration parameters:

- $\mathcal{G}$ = {'res', 'case', 'task'} denotes the three graph perspectives that could be used to analyse the resource swaps; and
- $\mathcal{V}$ = {'old', 'new', 'stable', 'all', 'disappeared', 'emerged'} denotes the six different resource allocation views to represent the resource reallocations between two event logs.

Definition 6 **(*Activity Sets*)**. Let $r \in \mathcal{R}$ be a resource. For each $v \in \mathcal{V}$ there is an $\mathcal{X}_v^r$ that maps the activities associated with resource r for resource allocation view v, defined as follows:

- $\mathcal{X}_v^r = \mathcal{A}^r$, if v = 'old',
- $\mathcal{X}_v^r = \overline{\mathcal{A}}^r$, if v = 'new',
- $\mathcal{X}_v^r = \mathcal{A}^r \cap \overline{\mathcal{A}}^r$, if v = 'stable',
- $\mathcal{X}_v^r = \mathcal{A}^r \cup \overline{\mathcal{A}}^r$, if v = 'all',
- $\mathcal{X}_v^r = \mathcal{A}^r \setminus \overline{\mathcal{A}}^r$, if v = 'disappeared', and
- $\mathcal{X}_v^r = \overline{\mathcal{A}}^r \setminus \mathcal{A}^r$, if v = 'emerged'.

For convenience, we use the notation $\mathcal{X}_v^y = \mathcal{A}^y$, for $y \in \mathcal{C} \cup \mathcal{T}$, even though there is no dependency on v.

Definition 7 **(*Resource Swap Weighted Digraph*)**. Let $L = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, $\overline{L} = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L, $g \in \mathcal{G}$ be a graph perspective, and $v \in \mathcal{V}$ be a resource allocation view. $RS_{L,\overline{L}}^{g,v} = (\mathcal{N}, \mathcal{E}, size, pie, weight)$ is a resource swap weighted digraph, with resource allocation view v and graph perspective g over event logs L and $\overline{L}$ where:

- $\mathcal{N}$ is the set of nodes, such that $\mathcal{N}$=
  - $\{\mathcal{X}_v^r | r \in \mathcal{R}\}$, if g = 'res',
  - $\{\mathcal{X}_v^y | y \in \mathcal{R} \cup \mathcal{C}\}$, if g = 'case', and
  - $\{\mathcal{X}_v^y | y \in \mathcal{R} \cup \mathcal{T}\}$, if g = 'task',
- $\mathcal{E}$ is the set of directed edges representing resource swaps, such that $\mathcal{E}$=
  - $\{(res(a), res'(a)) \in \mathcal{R} \times \mathcal{R} | a \in \mathcal{A}^{\Delta}\}$, if g='res',
  - $\{(res'(a), case(a)) \in \mathcal{R} \times \mathcal{C} | a \in \mathcal{A}^{\Delta}\} \cup \{(case(a), res(a)) \in \mathcal{C} \times \mathcal{R} | a \in \mathcal{A}^{\Delta}\}$, if g = 'case', and
  - $\{(res'(a), task(a)) \in \mathcal{R} \times | a \in \mathcal{A}^{\Delta}\} \cup \{(task(a), res(a)) \in \mathcal{T} \times \mathcal{R} | a \in \mathcal{A}^{\Delta}\}$, if g = 'task',
- for all $n \in \mathcal{N}$, $size \in \mathcal{N} \to \mathbb{N}$ is a function assigning sizes to nodes, $size(n) = |\mathcal{X}_v^n|$,
- for all $r \in \mathcal{R}$, $pie^r \in \to [0,1]$ is a function specifying the proportion of activities that were executed by resource rdependent on resource allocation view v, such that for all $t \in \mathcal{T}$, $pie^r(t)$=
  - $\frac{|\{a \in \mathcal{X}_v^r \mid task(a) = t\}|}{|\mathcal{X}_v^r|}$, *if* $\mathcal{X}_v^r \neq \emptyset$,
  - 0, otherwise.
- for all $(x, y) \in \mathcal{E}$, $weight \in \mathcal{E} \to \mathbb{N}$ is a function assigning weights to edges, such that $weight((x, y))$=
  - $|\mathcal{A}^x \cap \overline{\mathcal{A}}^y|$, if $(x, y) \in \mathcal{R} \times \mathcal{R}$,
  - $|(\overline{\mathcal{A}}^x \setminus \mathcal{A}^x) \cap \mathcal{A}^y|$, if $(x, y) \in (\mathcal{R} \times \mathcal{C}) \cup (\mathcal{R} \times \mathcal{T})$, and
  - $|(\mathcal{A}^y \setminus \overline{\mathcal{A}}^y) \cap \mathcal{A}^x|$, if $(x, y) \in (\mathcal{C} \times \mathcal{R}) \cup (\mathcal{T} \times \mathcal{R})$.

Fig. 2 in Section 3.2.1 illustrates a resource-swap weighted digraph from the resource perspective, with a resource allocation view v = 'new' (the resource nodes depicting the resource allocation in the perturbed event log). It can be observed that, although a high number of activities are reallocated from R2 to R1 (thicker edge pointing from R2 to R1), R2 is still executing more activities compared to R1 (larger node size for R2).

3.2.2. *Differences in resource utilisations*

The reallocation of activities from one resource to another will affect the rate of resource utilisation. The resource utilisation and the differences in resource utilisation are defined below.

Definition 8 **(*Resource Utilisation*)**. Let $L = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \text{case}, \text{task}, res, art, st, et)$ be an event log, and $WH\colon \mathcal{R} \to \mathcal{P}(Intv)$ be a set of valid working hours for each resource. For $r \in \mathcal{R}$, $util_L(r)$=

$$\frac{\sum_{a \in \mathcal{A}^r} dur_L^{WH}(a)}{\sum_{p \in WH(r)} dur(\min_{c \in C} art(c), \max_{c \in C} cet_L(c), stw(p), etw(p))}$$

is the utilisation of resource r over the duration of L, calculated by dividing the total duration of all activities executed by a resource, by the total working hours of that particular resource.

A divergent bar chart is chosen to highlight the differences between two event logs, illustrating either positive or negative changes to characteristics of the event logs' elements. Fig. 5 conceptualises a resource utilisation shift divergent bar chart. Each row represents a resource, where a bar illustrates the differences in utilisation for that resource. An increase in resource utilisation is illustrated with a green horizontal bar extending towards the right. Likewise, a decrease in resource utilisation is illustrated with a red horizontal bar extending towards the left. The length of each bar is commensurate with the magnitude of the corresponding change in utilisation. Potential observations from this visualisation type include:

- the change in resource utilisation; and
- whether the change exhibits itself predominantly for a particular resource or role.

Opposite directions, in this case, the increase or decrease in resource utilisations are represented by green and red colours. This is supported by the length of the bars, which indicates the magnitude of

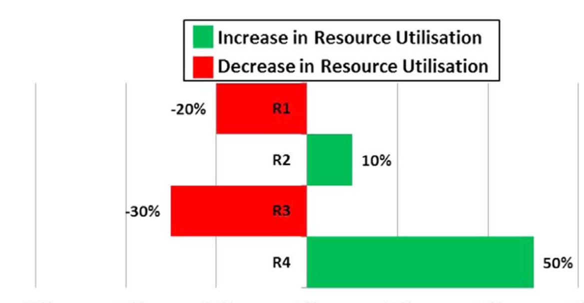


Fig. 5. The differences in resource utilisation between two event logs.

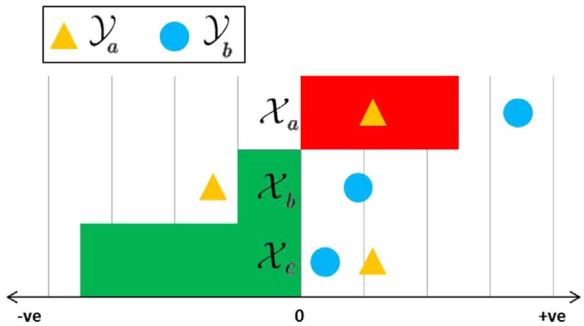


Fig. 6. An abstract picture of the divergent bar chart. The bars represent the base set $\mathcal{X}$, mapped based on the values in Val_{bar}. The plots represent the base sets $\mathcal{X} \times \mathcal{Y}$, mapped based on the values in Val_{plot}.

the change in utilisations. A divergent bar chart is defined and mapped onto an abstract picture (Fig. 6) below.

Definition 9 (***Divergent Bar Chart***). A divergent bar chart $(\mathcal{X}, \mathcal{Y}, Val_{bar}, Val_{plot})$ has base sets $\mathcal{X}$ and $\mathcal{Y}$ with bar-value function $Val_{bar} \in \mathcal{X} \to \mathbb{R}$ and partial plot-value function $Val_{plot} \in \mathcal{X} \times \mathcal{Y} \rightarrowtail \mathbb{R}$.

A resource utilisation shift divergent bar chart is defined as follows.

Definition 10 (***Resource Utilisation Shift Divergent Bar Chart***). Let $L = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, and $\overline{L} = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L. $RDC_{L,\overline{L}} = (\mathcal{R}, \emptyset, util_{\overline{L}} - util_L, \emptyset)$ is a divergent bar chart, referred to as a resource utilisation shift divergent bar chart, that maps the differences in resource utilisations between L and $\overline{L}$.

The difference in resource utilisations is always between the range of -1 to 1, and it can be translated to a percentage value.

3.2.3. Comparison of resource busy and idle times

The changes in a resource's busy and idle times might also be one of the many factors that influence the performance of a business process. Fig. 7 illustrates a visualisation concept of a resource twin-row chart. Each twin-row represents a resource, and each bar represents an activity that was executed by the resource. The length of each bar represents the duration of the corresponding resource performing the activity involved. The stability of an activity (whether it is a stable activity or a swapped activity) is expressed using different colours. Corresponding activities can be easily traced with the highlighting of matching activities (when an activity is being hovered over). Observations that can be drawn from this visualisation type include:

- the busy and idle times of resources;
- the activities that a particular resource executed;
- the activities' execution durations; and
- correlations between activities and resources.

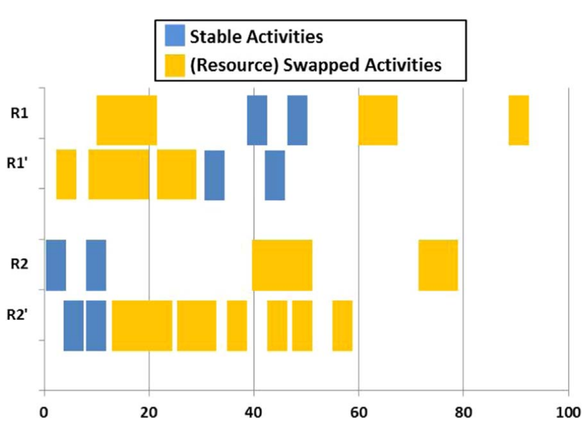


Fig. 7. Comparison of the busy and idle times of resources between two event logs.

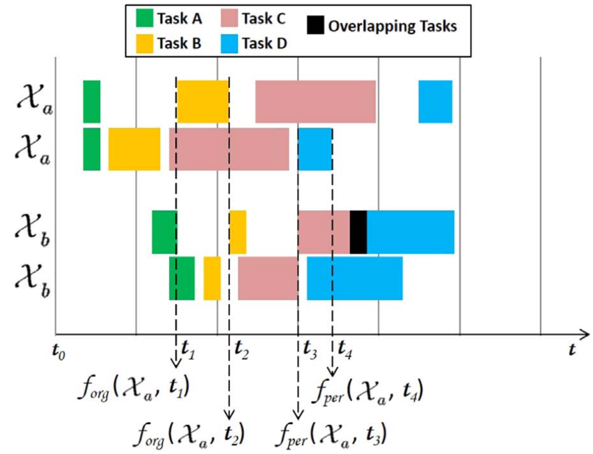


Fig. 8. An abstract picture of the twin-row chart.

To map out the periods where the elements are busy or idle, and compare the similarities and differences between two event logs, a twin-row chart is defined and mapped onto an abstract picture (Fig. 8) as follows.

Definition 11 (***Twin-Row Chart***). A twin-row chart $(\mathcal{X}, f_{org}, f_{per}, Val)$ has a base set $\mathcal{X}$ with two functions $f_{org}, f_{per} \in \mathcal{X} \times \mathbb{T} \to Val$. f_{org} maps $\mathcal{X}$ to a set of values Val that can be derived from the original event log at time points in $\mathbb{T}$, whereas f_{per} maps $\mathcal{X}$ to a set of values Val that can be derived from the perturbed event log at time points in $\mathbb{T}$.

A resource twin-row chart compares the resource busy and idle times between the two event logs, as well as whether the activities were reallocated from one resource to another. A resource twin-row chart is defined as follows.

Definition 12 (***Resource Twin-Row Chart***). Let $L = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, and $\overline{L} = (\mathcal{C}, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L. $RTC_{L,\overline{L}} = (\mathcal{R}, f_L^{rsc}, f_{\overline{L}}^{rsc}, \{\text{'stable'}, \text{'swapped'}, \text{'idle'}\})$ is a twin-row chart, referred to as a resource twin-row chart, such that for all $r \in \mathcal{R}$ and $z \in \mathbb{T}$,

- $f_L^{rsc}(r, z)$=
  - 'stable', if $\exists\, a \in \mathcal{A}^r\ [res'(a) = r \wedge st(a) \leq z < et(a)]$,
  - 'swapped', if $\exists\, a \in \mathcal{A}^r\ [res'(a) \neq r \wedge st(a) \leq z < et(a)]$, and
  - 'idle', otherwise, and
- $f_{\overline{L}}^{rsc}(r, z)$=
  - 'stable', if $\exists\, a \in \overline{\mathcal{A}}^r\ [res(a) = r \wedge st'(a) \leq z < et'(a)]$,
  - 'swapped', if $\exists\, a \in \overline{\mathcal{A}}^r\ [res(a) \neq r \wedge st'(a) \leq z < et'(a)]$, and
  - 'idle', otherwise.

The 'stable' and 'swapped' values in Val can then be mapped to different colours on a chart, whereas the 'idle' value in Val is transparent on a chart (Fig. 7).

The next section discusses the concept of time shift, where the start times of corresponding activities from two event logs are compared for shifts in time.

3.3. Changes from the time perspective

A *time shift* is defined as the shift in activity start times, where the affected activity could either start earlier (*forward time shift*) or later (*backward time shift*). Rescheduling of activities might result in a number of outcomes such as a change in case throughput time, a reduction of SLA (service level agreement) violations, and a change in resource utilisation. Thorough analysis of the changes that resulted in an improved business process, as well as suitable visualisations are essential in order to provide further insights. Again, inspired by the research studies in the area of BPI and reinforced by our research team's experience in the area of BPM, it is interesting to consider and answer resource-related questions such as:

- How many time shifts occurred within a particular case?
- Which cases are more prone to time shifts?
- How much has the throughput time reduced for a particular case?
- How much has the average waiting time reduced for a particular case?
- Which tasks are more prone to time shifts?
- Are activities that start within a certain time period more prone to time shifts?
- How much has an activity's start time shifted?
- When is the idle time within a case?

Providing answers for the questions above allow analysts to identify patterns (such as changes in activity start times and differences in idle times) that contributed to a reduction in execution cost. The insights gathered will enable stakeholders to take appropriate actions to improve business processes.

The time shifts will result in a number of changes, which include 1) differences in case throughput times; 2) differences in average activity start times; and 3) changes in case busy and idle times. These time-related differences are formalised, and a number of visualisations are conceptualised to portray these changes.

3.3.1. Case time shift (rescheduling of activities – case perspective)

The time shifts can manifest at both case and task levels. Within a case, the rescheduling of start times for individual activities will result in a change in case throughput times. In Fig. 9 a case time shift divergent bar chart is visualised. Each row represents a case, where the bar length represents the difference in case throughput time. A reduction in case throughput time is expressed by a green horizontal bar extending to the left. Likewise, an increase in case throughput time is expressed by a red horizontal bar extending to the right. The length of each bar is commensurate with the magnitude of the corresponding time shift. Each activity is represented by a point (with a certain shape and a certain colour) along the row of its corresponding case. The position of a point reflects the shift in an activity's start time. The further left a point is, the earlier the corresponding activity has started in the perturbed event log compared to the original event log. Similarly, the further right a point is, the later the corresponding activity has started in the perturbed event log compared to the original event log. Potential observations from this visualisation type include:

- the throughput time differences for a case;
- the time shift of an activity within a particular case; and
- the activities that tend to be rescheduled earlier or later.

To better illustrate the shifts in activity start times and case throughput times between two event logs, an activity time shift is defined below. This is followed by the definition of a case time shift divergent bar chart.

Definition 13 ***(Activity Time Shift)***. Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, and $\overline{L} = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L. $tshift^{act}_{L,\overline{L}} \in \mathcal{A} \to \mathbb{R}$ is a function specifying the start time difference for an activity over event logs L and $\overline{L}$, where for all $a \in \mathcal{A}$, $tshift^{act}_{L,\overline{L}}(a) = st'(a) - st(a)$. Furthermore, $tshift^{act.case}_{L,\overline{L}} \in C \times \mathcal{A} \to \mathbb{R}$ is a function specifying the start time difference for an activity within a case, and $tshift^{act.task}_{L,\overline{L}} \in \mathcal{T} \times \mathcal{A} \to \mathbb{R}$ is a function specifying the start time difference for an activity of a task, where for all $a \in \mathcal{A}$, $c \in C$, and $t \in \mathcal{T}$,

- $tshift^{act.case}_{L,\overline{L}}(case(a), a) = tshift^{act.task}_{L,\overline{L}}(task(a), a) = tshift^{act}_{L,\overline{L}}(a)$,
- $tshift^{act.case}_{L,\overline{L}}(c, a)$ is undefined for $c \neq case(a)$, and
- $tshift^{act.task}_{L,\overline{L}}(t, a)$ is undefined for $t \neq task(a)$.

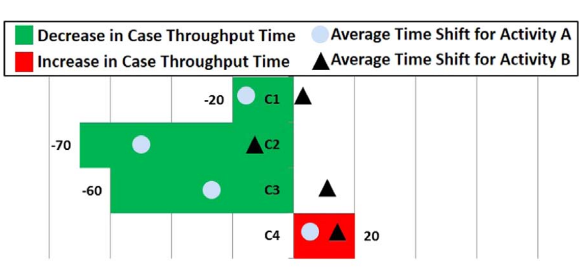


Fig. 9. The differences in case throughput times between two event logs.

Definition 14 ***(Case Time Shift Divergent Bar Chart)***. Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, and $\overline{L} = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L. $CDC_{L,\overline{L}} = (C, \mathcal{A}, tpt_{\overline{L}} - tpt_L, tshift^{act.case}_{L,\overline{L}})$ is a divergent bar chart, referred to as a case time shift divergent bar chart, that maps the difference in throughput time of cases, as well as the time shift of activities within cases in L and $\overline{L}$.

3.3.2. Task time shift (rescheduling of activities - task perspective)

Time shifts can be aggregated and analysed from a task perspective. Fig. 10 conceptualises the task time shift divergent bar chart. Each bar on a row represents a task and its average time shift across all cases that contain that particular task instance (activity). A green horizontal bar that extends towards the left shows that on average, the task instances tend to start earlier (forward time shift). Likewise, a red horizontal bar that extends towards the right shows that on average the task instances tend to start later (backward time shift). The length of each bar is commensurate with the magnitude of the corresponding time shift. If a task instance is present in a case, the case is represented by a point (coloured shape) along the row of that task. The further left a point is, the earlier an instance of that task in that case has started in the perturbed event log compared to the original event log (forward time shift). Similarly, the further right a point is, the later an instance of that task in that case has started in the perturbed event log compared to the original event log (backward time shift). Potential observations from this visualisation type include:

- the average time shift of a task;
- whether activities in a certain case tend to reschedule earlier or later; and
- possible task patterns and trends.

A task time shift divergent bar chart is defined below.

Definition 15 ***(Task Time Shift Divergent Bar Chart)***. Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, and $\overline{L} = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L.

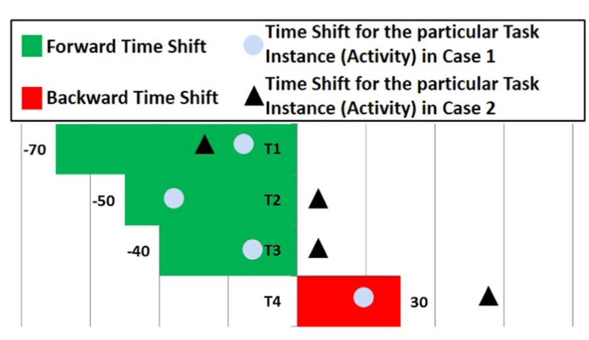


Fig. 10. Aggregation and visualisation of the average time shift for tasks and task instances within cases (each case is represented by a combination of colour and shape).

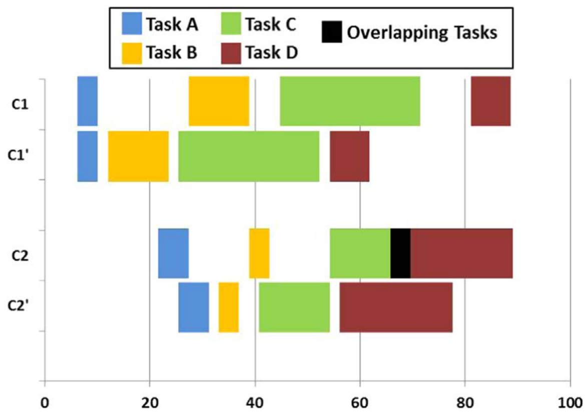


Fig. 11. A bird's-eye-view of the cases, and the activities that are executed within them.

$TDC_{L,\bar{L}} = (\mathcal{T}, \mathcal{A}, tshift^{task}_{L,\bar{L}}, tshift^{act,task}_{L,\bar{L}})$ is a divergent bar chart, referred to as a task time shift divergent bar chart, that maps the difference of the average time shift for all activities that are instances of a task, as well as the time shift of activities of different tasks in L and $\bar{L}$, such that for all $t \in \mathcal{T}$,

$$tshift^{task}_{L,\bar{L}}(t) = \frac{\sum_{a \in \mathcal{A}^t} tshift^{act}_{L,\bar{L}}(a)}{|\mathcal{A}^t|}.$$

3.3.3. Comparison of case busy and idle times

In Fig. 11 a case twin-row chart is visualised. Every twin-row compares the execution difference of a particular case between the two event logs. The activities that are executed in the case are represented by the bars, distinguished by their colours. The length of each bar is commensurate with the execution duration of the corresponding activity. The colour of a bar represents the task of its corresponding activity. If two or more activities are concurrently being executed within the same case, the colour black will represent the period where the activities overlap in time. The total length of a row represents the throughput time of its corresponding case. Empty gaps between bars illustrate idle time, where no activity is being executed. Corresponding activities can be easily traced with the highlighting of matching activities (when an activity is being hovered over). Below are some possible observations that can be made from this visualisation type:

- the throughput time and idle time differences for a particular case between two event log variants; and
- possible patterns, e.g., a certain activity is usually associated with an increase in idle time.

A twin-row chart is adapted to compare the case busy and idle times between two event logs. A case twin-row chart is defined as follows.

Definition 16 ***(Case Twin-Row Chart)***. Let $L = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res, art, st, et)$ be an event log, $\bar{L} = (C, \mathcal{A}, \mathcal{T}, \mathcal{R}, case, task, res', art, st', et')$ be a perturbation of L, and $tv \in \rightarrow Val$ is an injective function mapping tasks to values. $CTC_{L,\bar{L}} = (C, f^{case}_{L}, f^{case}_{\bar{L}}, \cup \{\perp, \top\})$ is a twin-row chart, referred to as a case twin-row chart, such that for all $c \in C$ and $z \in \mathbb{T}$,

- $f^{case}_{L}(c, z)$=
  - v, if $\exists\, t \in \mathcal{T}, \exists\, !a \in \mathcal{A}^c\ [tv(t) = v \wedge st(a) \leq z < et(a)]$,
  - $\perp$, if $\neg\, \exists\, a \in \mathcal{A}^c\ [st(a) \leq z < et(a)]$, and
  - $\top$, otherwise, and
- $f^{case}_{\bar{L}}(c, z)$=
  - v, if $\exists\, t \in \mathcal{T}, \exists\, !a \in \mathcal{A}^c\ [tv(t) = v \wedge st'(a) \leq z < et'(a)]$,
  - $\perp$, if $\neg\, \exists\, a \in \mathcal{A}^c\ [st'(a) \leq z < et'(a)]$, and
  - $\top$, otherwise.

Values in Val, (in this case, the set of tasks, $\perp$, and $\top$) can then be mapped to different colours on a chart (Fig. 11).

4. Implementation

The visualisation concepts are realised via plug-ins within the ProM framework [5]. Three ProM plug-ins have been implemented to illustrate the visualisation concepts presented in this paper[2]. Two event logs are required as input. The first ProM plug-in checks whether the second event log is a perturbation of the first event log. An event log is deemed as a perturbation of the original event log only if its resource allocations and activity start and end times have been modified, while everything else remains unchanged. The two event logs will form a pair of logs, used as a basis to analyse resource swaps and time shifts. The analysis results are illustrated using two visualisation types. The second ProM plug-in is responsible for the graph visualisations (as defined in Section 3.2.1), whereas the third ProM plug-in presents the chart visualisations (as defined in Sections 3.2.2, 3.2.3, 3.3.1, 3.3.2, and 3.3.3).

4.1. Visualisation design principles

This work proposes a number of visualisations, each of them providing a distinctive, analysis-worthy perspective for process analysts. The three attributes below can be used to describe the characteristics of these visualisations:

- **Process Aspects**. Two process aspects are investigated for differences - the resource perspective (resource swaps) and the time perspective (time shifts). These aspects are portrayed using different visualisations.
- **Visualisation Types**. Two visualisation types are utilised in this study. As graphs are commonly used to represent relationships among elements [30,31], we used social network graphs to illustrate the reallocation of resources. A timeline visualisation approach is useful for depicting time-related information and identifying temporal patterns [32,33]. As charts are easy to learn and are a standard approach to visualise temporal information, (divergent and twin-row) bar charts are used to compare timing information from different process-related elements for this study.
- **Comparison Types**. The comparative analysis can be achieved using two methods. A side-by-side comparison provides analysts with a clear view of two event logs, where not only the differences, but also the similarities can be identified clearly. This type of comparison manifests itself in the twin-row chart. Secondly, a δ-view illustrates and highlights only the differences between two event logs. The resource swap weighted digraph and divergent bar chart exhibit this type of comparison.

The proposed visualisations were guided by Shneiderman's visual information seeking mantra – "overview first, zoom and filter, then details-on-demand" [26]. The visualisations are based on the following design rationale:

- The initial views of the visualisations are generated and presented at an appropriate **abstraction level** to increase their comprehensibility.
- Analysts can further **sort, filter, and drill down** into the visualisation elements according to their needs for a more targeted analysis. In addition, analysts can refer to the statistics tables and panels to acquire further information as necessary.
- Specialised design principles and interactivity (e.g., the **pie-in-node** concept and **highlighting functionality**) are also applied to certain visualisations to reduce complexity, and at the same time, increase the amount of information that can be portrayed in the visualisations. These design principles for each visualisation are

[2] The developed ProM plug-ins can be found online at http://www.yawlfoundation.org/cost/logbasedcostanalysisandimprovement.html.

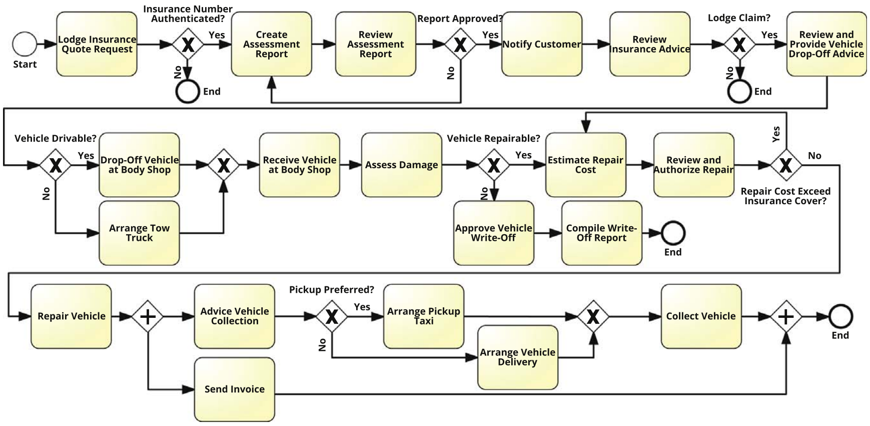


Fig. 12. A BPMN model illustrating the car insurance claim process.

elaborated in their corresponding sections.

- The visualisations' **interfaces** are standardised. The configuration panel is always on the left, the statistics panel (for the chart visualisations) is on the right, and the legend bar is positioned on top. (Refer to Figs. 13 and 15.)
- Colour is considered to be the most important visualisation factor, followed by size and shape [51]. In this work, **colours** are utilised in a number of ways, depending on the visualisation's purpose. To represent flows in opposite directions, complementary colours (e.g., green and red) are used [52]. To represent multiple items (nominal data), colours with easily distinguishable colour schemes (quantitative/segmented colour scales) are chosen [52]. In addition, users can change the colours according to their preferences.
- **Shapes** are used to complement the colours for visualisations that contain many elements. Only shapes such as circles, squares, triangles, diamonds, and stars are used, as they are visually distinctive for process analysts to differentiate amongst them.
- **Node size**, **edge weight** and **bar length** are used to represent magnitude. For charts, the X-axis represents either the magnitude or the duration. The magnitude is determined by the positioning of a graphical representation, whereas the duration of an item is represented by its length. To represent the magnitude of change in a graph, the size (of nodes), and the weight (of edges) are used.

4.2. Illustrative example

A simulated car insurance claim process (illustrated in Fig. 12) is used as a running example in this paper. The process starts off within the Insurance Company when a customer lodges an insurance quote request for his damaged car. If the customer's insurance number is positively authenticated, an Insurance Adjuster will create an assessment report based on the quote description. The Insurance Supervisor will then review the assessment report. If the report is approved, the Insurance Adjuster will notify the customer. If the report is rejected, the Insurance Adjuster will need to recreate the report. The customer will then decide whether to lodge a claim by reviewing the insurance advice. Should the customer lodge a claim, a Service Coordinator will review and provide advice on dropping-off the vehicle at the body shop. If the vehicle is still drivable, the customer will drop-off the vehicle at the body shop. If not, the Service Coordinator will arrange a tow truck to collect the vehicle on behalf of the customer.

Within the Body Shop, a Foreman receives the damaged vehicle and assesses the damage. If the damage is not repairable, a Body Shop Supervisor will approve the vehicle for write-off, and a Write-Off Specialist will compile a write-off report afterwards. If the damage is repairable, the Foreman will estimate the repair cost and provide feedback to the Insurance Supervisor. If the estimation exceeds the insurance cover, the estimation will be rejected, and the Foreman will need to readjust the estimated repair cost. Otherwise, the Foreman will repair the vehicle accordingly. After the repair is completed, the Body Shop Supervisor will send an invoice to the Insurance Company, while the Service Coordinator advises the customer for vehicle collection. The vehicle can either be delivered to the customer, or a pick-up taxi be arranged to transport the customer to the Body Shop for vehicle pick-up. The process is complete when the customer collects the vehicle.

The process has been translated into a Petri net [53] and is simulated using CPN Tools[3] in order to obtain an event log. We created an event log with 500 cases in this way. The process consists of 16 tasks and spans across two organisational groups. 29 resources are involved, that are categorised into 6 roles. Each task can only be executed by a certain role or combination of roles. Each resource group (role) is responsible for a fixed set of tasks. Realistic working hours are also introduced in the process simulation, where all resources work between 9am and 5pm.

The generated event log was optimised using a cost-informed process improvement approach [7], resulting in a perturbed event log. The perturbed event log demonstrated a cost reduction of 6.36%, as well as an 85% reduction in average waiting time within a case. Using the original event log and perturbed event log as inputs, the differences between the two event logs are analysed and visualised. The visualisations to illustrate the differences from a resource perspective are demonstrated first, followed by the time perspective.

4.3. Resource perspective implementation

The implementation of resource swap weighted digraph, resource utilisation shift divergent bar chart, and resource twin-row chart, are illustrated as follows.

4.3.1. Resource swap weighted digraph. Three resource swap weighted digraphs are realised to depict resource swaps from resource, case, and task perspectives. Fig. 13 illustrates a resource swap weighted digraph, which visualises the reallocation of activities from one resource to another. Fig. 14a illustrates a resource swap weighted digraph from a case perspective, whereas Fig. 14b illustrates a resource swap weighted digraph from a task perspective. The resource nodes in these three visualisations represent the number of activities that did not reallocate (stable activities). The Spring layout algorithm of the JUNG Framework[4] was chosen as its simple force-directed layout automatically disperses the nodes for easy comprehension by users. Note that the edges represent all tasks (that are not filtered out). If the user would like to examine specific task(s) in detail, filtering of tasks can be performed using the settings on the left panel. The configuration panel on the left allows process analysts to configure node and label settings, filtering options, and resource allocation view options

[3] http://cpntools.org/

[4] http://jung.sourceforge.net/

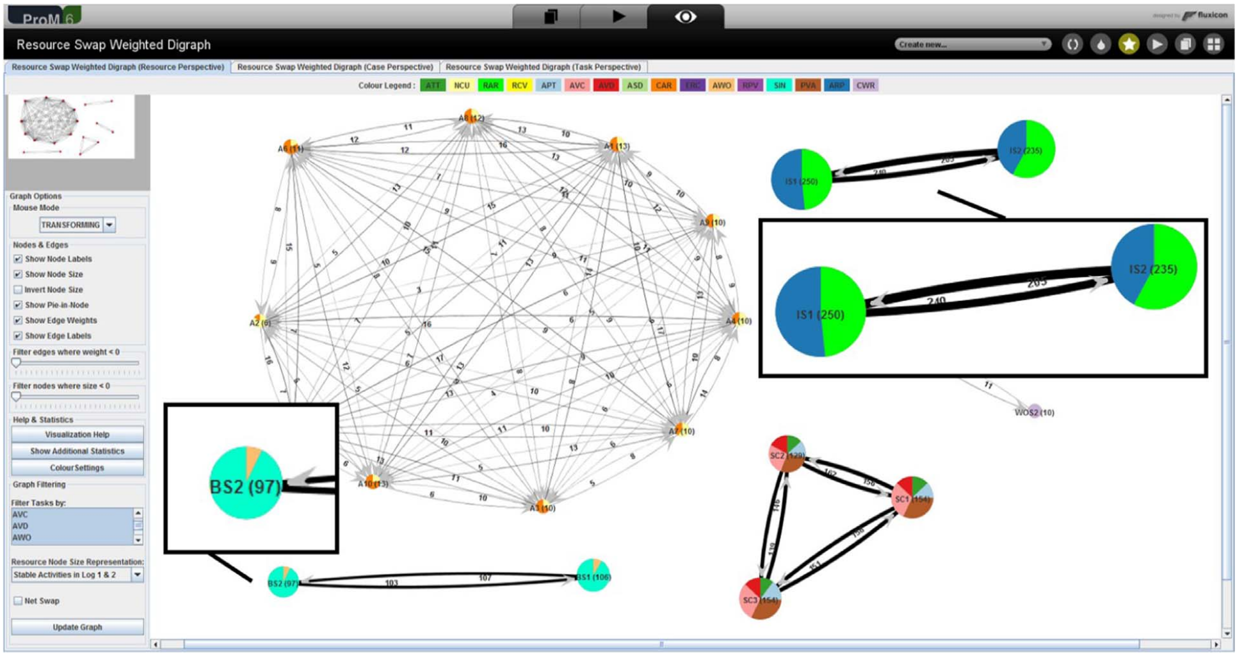
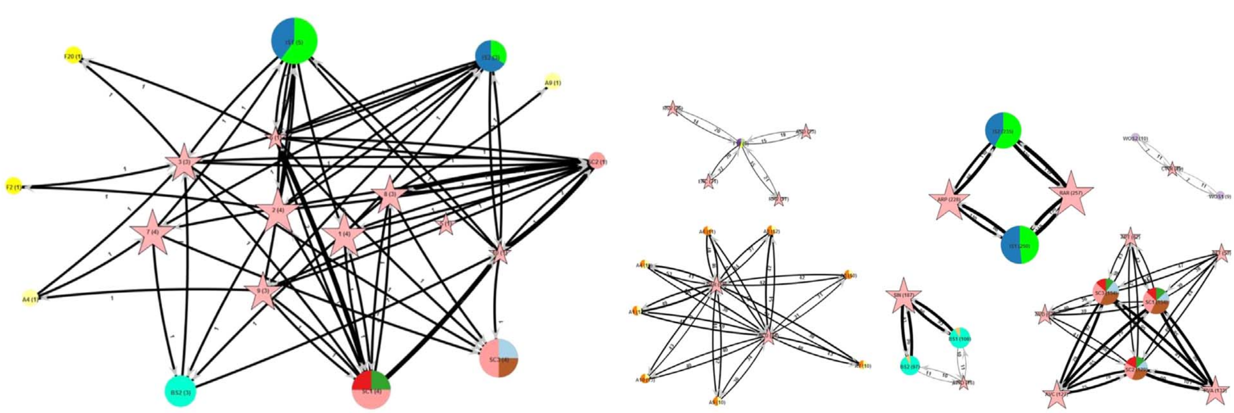


Fig. 13. An example visualisation of the resource swap weighted digraph (resource perspective), which corresponds to the conceptualisation in Fig. 2.



(a) (b)

Fig. 14. Fig. 14a is an example visualisation of a resource swap weighted digraph (case perspective); Fig. 14b illustrates a resource swap weighted digraph from a task perspective, which corresponds to the conceptualisations in Figs. 3 and 4.

according to their need. Focusing on Fig. 13, a number of observations can be made:

- Resource swaps only occur within the same role, as there are no edges between different roles. (The graphs presented above have had their layout adjusted to emphasise this.)
- Among Body Shop Supervisors (BS1 and BS2, lower left-corner), most of the SIN activities did not reallocate, as depicted by the larger proportion of the cyan-coloured pie slice.
- Activities executed by the Adjusters (A1 – A10, upper left) tend to be reallocated (high number of edges), possibly due to sub-optimal resource allocation.
- Both Insurance Supervisors (IS1 and IS2, upper right-corner) retain a high number of activities (large node sizes). A reason for this may be that they were already highly utilised, minimising the chances of resource reallocation occurring.

4.3.2. Resource utilisation shift divergent bar chart. A resource utilisation shift divergent bar chart (Fig. 15) visualises the change in resource utilisations between two event logs. A green bar that extends to the right indicates an increase in resource utilisation, whereas a red bar that extends to the left indicates a decrease in resource utilisation. Process analysts are able to sort and filter the resources using the configuration panel on the left. A number of observations can be made from Fig. 15:

- F6 has the highest increase in its utilisation (26%), followed by F11 (12%).
- The majority of the resources only have a slight change in their utilisation (± 5%), likely limited by their eligibility and working hours.
- Most of the resources that had their utilisations increased are Foremen (F1–F20), again, possibly due to their suboptimal resource allocation.

4.3.3. Resource twin-row chart. The resource twin-row chart (Fig. 16) enables users to view the resources busy and idle times, and compares

Fig. 15. An example visualisation of the resource utilisation shift divergent bar chart, which corresponds to the conceptualisation in Fig. 5.

the differences between two event logs. An orange-coloured bar suggests that the activity is reallocated from one resource to another, whereas a blue-coloured bar highlights the unchanged parts. The configuration panel enables the sorting and filtering of resources, as well as the option to change the colour representations. A number of observations can be made from Fig. 16:

- Most of the activities were reallocated (indicated by the orange-coloured bars), indicating that the scheduling and execution of activities could be improved.
- Most of the activities executed by Foremen (F1, F2, etc., from the third twin-row onwards) have a longer duration compared to activities executed by Body Shop Supervisors (BS1 and BS2, in the first and second twin-rows).
- Resources are idle more than half of the time, indicating a low resource utilisation.

Process analysts can locate the corresponding activity within the other event log using the **highlighted activity pair** functionality. By pointing to a certain activity, both the activity of interest and its corresponding activity will be highlighted. In addition, by double clicking any activity within the resource bar of interest, all activities within the resource bar, along with all corresponding activities, will be highlighted (**highlighted resource**). These actions help analysts to quickly determine the resource allocation (and timing) differences of

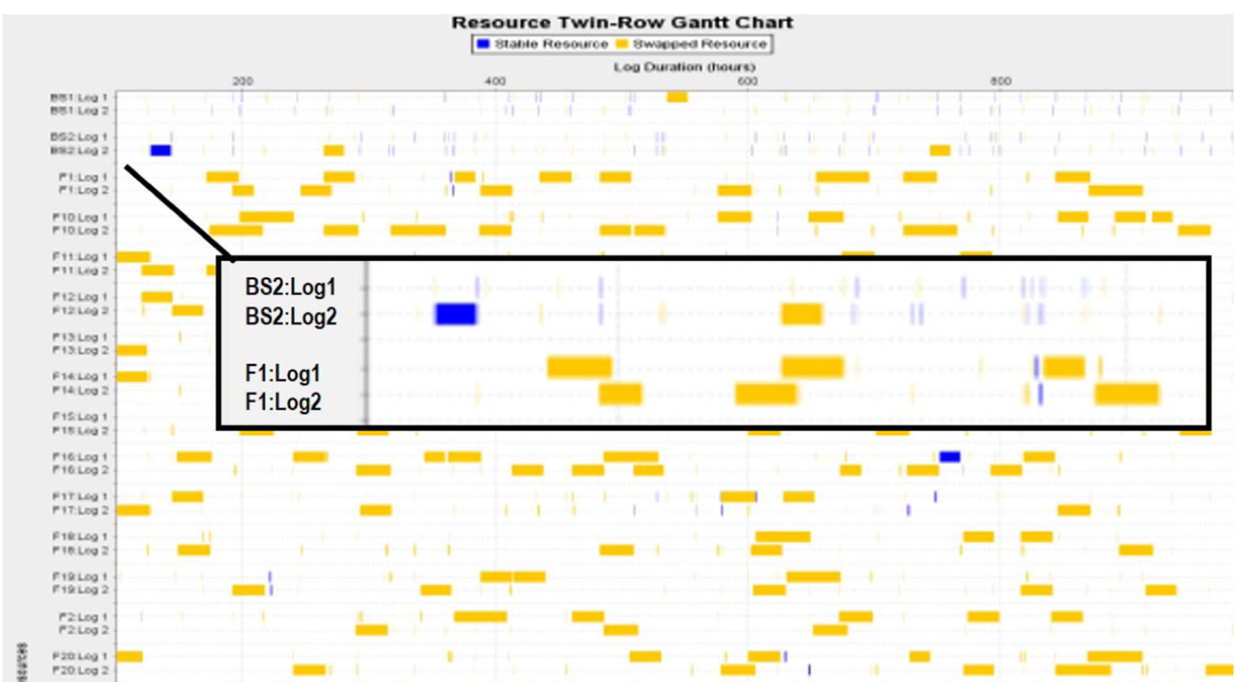


Fig. 16. An example visualisation of the resource twin-row chart, which corresponds to the conceptualisation in Fig. 7.

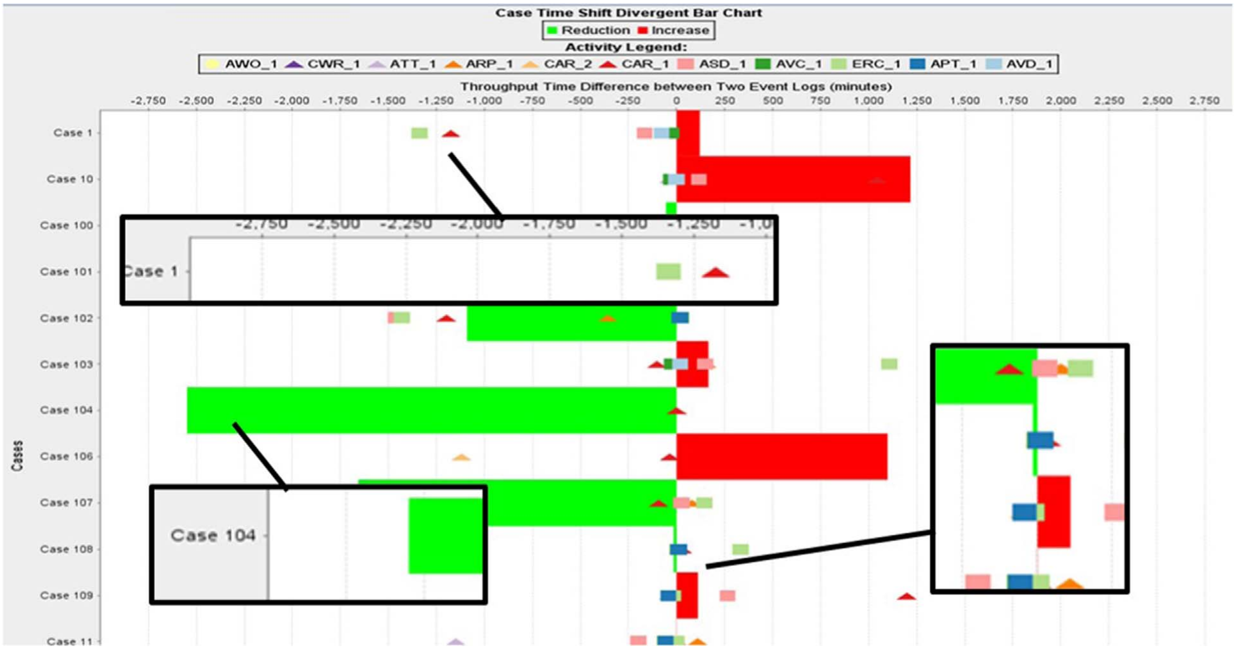


Fig. 17. An example visualisation of the case time shift divergent bar chart, which corresponds to the conceptualisation in Fig. 9.

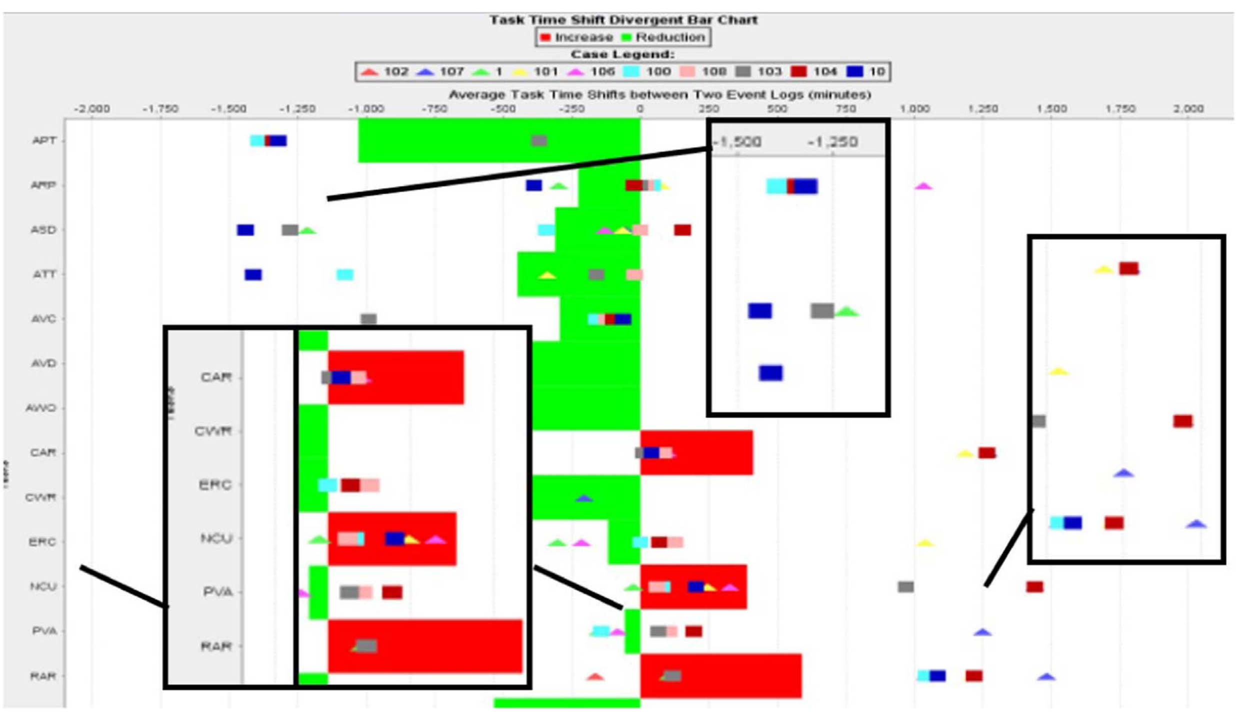


Fig. 18. An example visualisation of the task time shift divergent bar chart, which corresponds to the conceptualisation in Fig. 10.

activity between the two event logs.

By utilising these visualisations, business process redesign best practices (notably, best practices from a resource perspective, e.g., flexible resource assignment, specialist-generalist, and extra resources) proposed by Mansar and Reijers [1] can be supported.

4.4. Time perspective implementation

The implementation of case time shift divergent bar chart, task time shift divergent bar chart, and case twin-row chart, are illustrated as follows.

4.4.1. Case time shift divergent bar chart. Fig. 17 illustrates the case time shift divergent bar chart, which visualises the difference in case throughput times and activity start times between two event logs. A reduction in case throughput time in the perturbed event log compared to the original event log is expressed by a green-coloured bar extending to the left. An increase in case throughput time in the perturbed event log compared to the original event log is expressed by a red-coloured bar extending to the right. The configuration panel on the left allows process analysts to sort the cases, as well as to filter the cases and tasks. A number of observations can be made from Fig. 17:

- Case 104 had its throughput time reduced by roughly 2500 min, signifying that the case has been idle most of its throughput time in the original execution.
- Case 10 had its throughput time increased by roughly 1200 min.

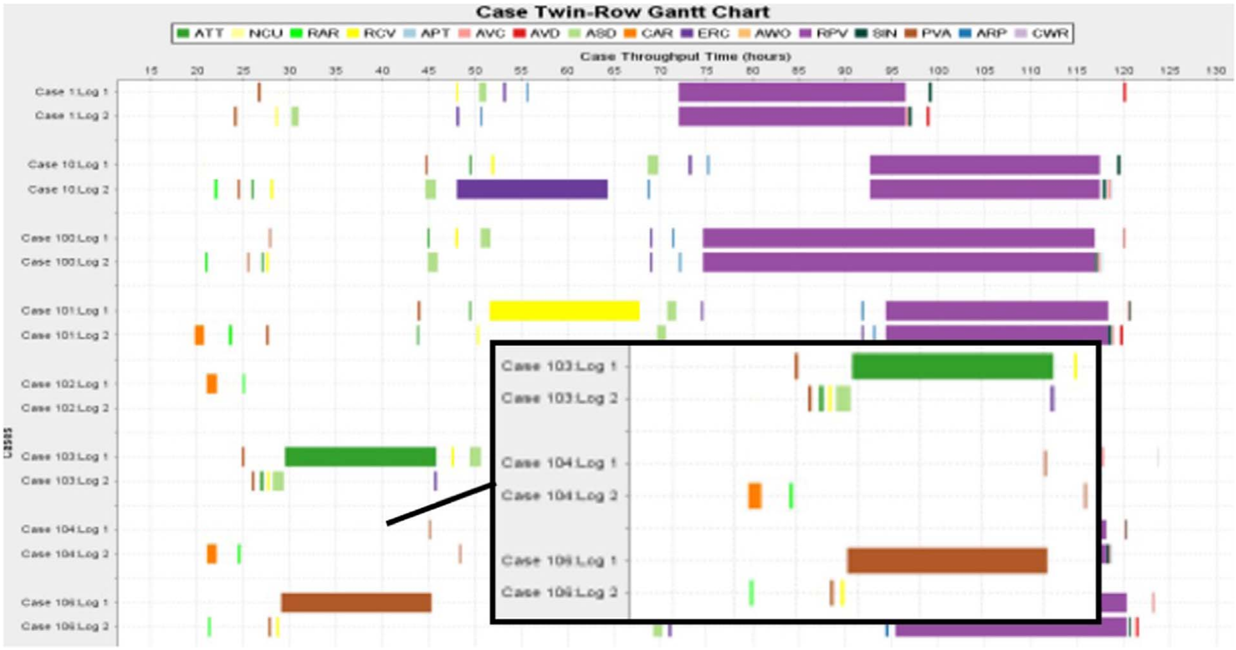


Fig. 19. An example visualisation of the case twin-row chart, which corresponds to the conceptualisation in Fig. 11.

- In Case 1, activities ERC (light green-coloured square) and CAR (red-coloured triangle) were shifted to start roughly 1250 min earlier.
- ERC instances tend to shift a lot (indicating a big room for improvement), whereas AVD (light blue-coloured squares) and APT (dark blue-coloured squares) instances did not deviate a lot from their original start times (indicating minimal room for improvement).

4.4.2. Task time shift divergent bar chart. The task time shift divergent bar chart (Fig. 18) aggregates and visualises the time-related differences between two event logs from the task perspective. A green bar that extends to the left indicates a "forward time shift", where instances of its corresponding task tend to start earlier. Likewise, a red bar that extends to the right indicates a "backward time shift", where instances of its corresponding task tend to start later. The sorting and filtering of cases and tasks can be achieved using the configuration panel on the left. A number of observations can be made from Fig. 18:

- The majority of the task instances started earlier, resulting in shorter case throughput times.
- Tasks CAR, NCV, and RAR tend to start later, possibly delayed to allow more critical activities to execute first.
- The majority of activities within Case 10 (blue squares) started earlier.
- A number of activities within Case 104 (dark red squares) were delayed.

4.4.3. Case twin-row chart. The case twin-row chart (Fig. 19) provides an overall comparative view of cases and their activities between two event logs. A unique colour for each bar represents the task of the activity. If two or more activities are running concurrently within the same case, the colour black will represent the period where the activities overlap in time. Process analysts can sort and filter the cases and tasks using the configuration panel on the left. A number of observations can be made from Fig. 19:

- The process improvement approach resulted in increased case utilisation and reduced throughput time for most cases.
- However, most cases have various periods where no work is done on the case.
- With working hours of resources taken into account, activities that ran overnight were discouraged, as longer execution duration often translates to higher costs.
- RPV instances (long, purple-coloured bars) have a lengthy execution time, and often executed throughput the night. This resulted in the inability to reschedule such activities.

To increase the visualisation's usability, process analysts can locate the corresponding activity within the other event log using the **highlighted activity pair** functionality. By pointing to a certain activity, both the activity of interest and its corresponding activity will be highlighted. This action can enable analysts to quickly determine the timing differences for that particular activity between the two logs.

Visualisations from a time perspective can provide insights into best practices such as integration between customers or suppliers, operational behaviours, and resourcing requirements [1]. These insights will lead to evidence-based business process improvement actions.

5. User evaluation

A user evaluation was conducted via a survey to find out 1) how users from varying demographics perceive visualisations in general; 2) whether the developed visualisations are doing what they are intended to do; and 3) the perceived usefulness and the user acceptance of the developed visualisations [54]. Pilot visualisations were first implemented and evaluated via this survey. The feedback gathered from the survey was already taken into account (refer to the final paragraph of Section 5.1) to improve the visualisations presented in Section 4. A copy of the survey is available online[5].

The survey is targeted towards participants from around the world who have prior BPM knowledge, e.g., BPM practitioners, BPM aca-

[5] http://www.yawlfoundation.org/cost/logbasedcostanalysisandimprovement.html

demics, and BPM (masters) students. The survey was conducted online, and was distributed via a number of channels, including:

- process mining-based social media (LinkedIn and Yammer);
- IEEE Task Force on Process Mining mailing list;
- ProM developers community mailing list; and
- BPM masters student mailing lists within some of the authors' universities[6].

The participants were expected to answer a total of 38 survey questions within an estimated duration of 20 min. The survey first provided the background and motivation for this research. Then, the participants were required to answer four questions regarding their demographics. After that, six survey sections were presented in order, each of them containing one visualisation. In each section, some introductory notes for a particular type of visualisation were given. The participants were then asked to answer an average of six questions by observing the given visualisations, to evaluate the perceived intuitiveness and usefulness of the particular visualisation type, and to provide any additional comments. Example questions include "Which resource retained the most number of activities in Log 2 compared to Log 1?" (multiple choice), "Which case has the biggest forward time shift (started earlier) for task ASD?" (multiple choice), and "It is easy and intuitive to find the answers to the questions above using this type of visualisation." (Likert scale). The perceived intuitiveness and usefulness were measured using the Likert scale (Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree). Finally, participants were asked to evaluate the usefulness of the visualisations as a whole and provide any final remarks.

5.1. Evaluation results

Out of the 42 people that started the survey, a total of 28 people completed it, which translates to a completion ratio of 40%. Responses from two participants were eliminated, as one selected the option "I don't know" for every question, whereas another displayed a pattern of selecting the first option for all questions, leading us to believe that these two participants did not complete the survey in good faith. This resulted in 26 participants (15 BPM academics, 4 BPM students, 3 BPM practitioners, and 4 others), with varying levels of experience (7 have more than 5 years, 10 have between 1 to 5 years, 6 have less than 1 year, and 3 have no experience).

All participants agree or strongly agree that they are aware of the concept of BPM. When the survey questioned their preferred information conveying methods, 84.6% (46.2% strongly agree and 38.5% agree) prefer visualisations over textual representations. A majority of the participants make use of visualisations during their work (84.6%), and more than half of the participants make use of visualisations for business process analysis or business process improvement purposes (65.4%).

A total of six visualisations were evaluated, where participants were required to answer a set of questions for each visualisation. The six visualisations that were chosen for evaluation are the *case twin-row chart*, *resource twin-row chart*, *resource swap weighted digraph (resource perspective)*, *resource utilisation shift divergent bar chart*, *case time shift divergent bar chart*, and *task time shift divergent bar chart*. Fig. 20 illustrates the accuracy of the answers for all visualisations submitted by the participants. On average, participants got 85% of the answers correct across all six visualisations. In particular, students and practitioners have a slightly higher number of correct answers, with 89% and 88% of the questions answered correctly, as opposed to 83% for academics and 86% for other participants. Notably, the questions for the resource utilisation shift divergent bar chart were answered most accurately, with only one participant getting one answer wrong. The accuracy of the answers for the resource twin-row chart scored lowest. As much as 30.8% of the participants either got both answers wrong, or selected the option "I don't know".

Figs. 21 and 22 illustrate the perceived intuitiveness and perceived usefulness of the developed visualisations, respectively. On average, 62.6% perceived the visualisations to be intuitive and useful. Notably, the visualisations resonated positively with the practitioners, with an average of 84.3% perceiving the visualisations to be intuitive and useful. The resource utilisation shift divergent bar chart and case twin-row chart fared better, with an average of 88.5% and 76.9% of the participants, respectively, perceiving the visualisations as intuitive and useful. The other visualisations performed moderately on the intuitiveness and perceived usefulness scale, with two of the participants strongly disagreeing that the visualisation is intuitive and useful. Overall, no participants disagree or strongly disagree that the tool is useful.

The feedback given by the survey participants was consolidated and analysed. Overall, the visualisations' scalability, the use of colours, and the perceived ease-of-use of the visualisations are the top concerns. Participants raised the issue of visualisation scalability, reinforcing their arguments with the large amount of real world data that is to be visualised. The use and interpretation of colours was questioned as well, on the grounds that colour-blindness, overloading of activities, and smaller activity bars, might result in colours and their corresponding tasks not being interpreted properly. A number of comments were directed towards the perceived ease-of-use of the visualisation, which was not measured in this evaluation. These comments suggest the use of labels, hover-on tooltips, and highlighting of matching activities will be helpful. While not mentioned in the survey, these functionalities were already implemented before the survey was conducted (refer to Section 4).

We now investigate and discuss the comments provided for individual visualisations:

- **Case twin-row chart.** It is assumed that the end of the last activity bar denotes the end time of the particular case. It appears this is not clear, because two participants suggested the case end times be marked or visualised as well.
- **Resource twin-row chart.** The reason for the low answer accuracy and the low intuitiveness and usability ratings for this particular visualisation can be explained as follows. The unconventional concept of *resource swap* proved to be difficult for participants to interpret. One participant even questioned the importance of this perspective, with the comment "*I am not convinced that it is important to see which particular resource has changed task with which particular resource*". Another participant also suggested the importance of viewing the type of tasks that were executed by the resources, instead of only viewing the resource stability.
- **Resource swap weighted digraph.** Again, the concept of *resource swap* was difficult for participants to grasp. However, the use of a graph proved to be beneficial, as a number of participants mentioned that the visualisation was intuitive and the questions easy to answer without any semantic comprehension.
- **Resource utilisation shift divergent bar chart.** Most comments pointed out the use of colours in the visualisation, where a decrease in resource utilisation is illustrated using a green bar, and an increase illustrated by a red bar. We acknowledge that a reduction in resource utilisation might not be desired for some organisations.
- **Case time shift divergent bar chart.** Participants deemed this visualisation to be potentially useful, with comments such as "*It's still a bit difficult but I think useful, because it shows correlations between two aspects of the process that are hard to get together*".
- **Task time shift divergent bar chart.** One participant provided interesting feedback, stating "*seeing going from right to left as a*

[6] Queensland University of Technology (QUT), Brisbane, Australia, and Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium.

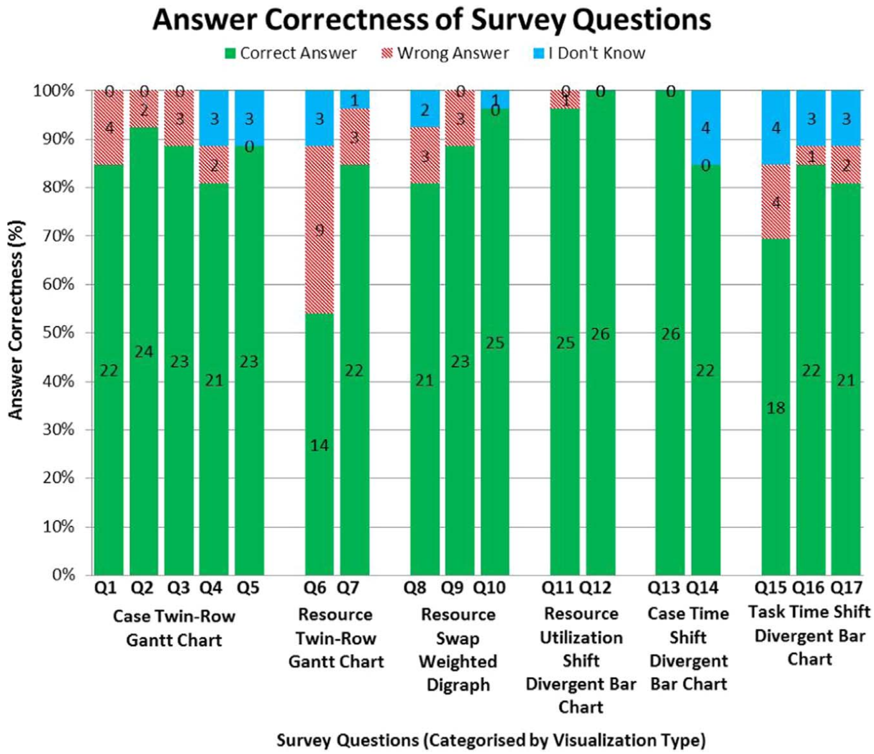


Fig. 20. Breakdown of the answers' accuracy for questions from all six visualisations.

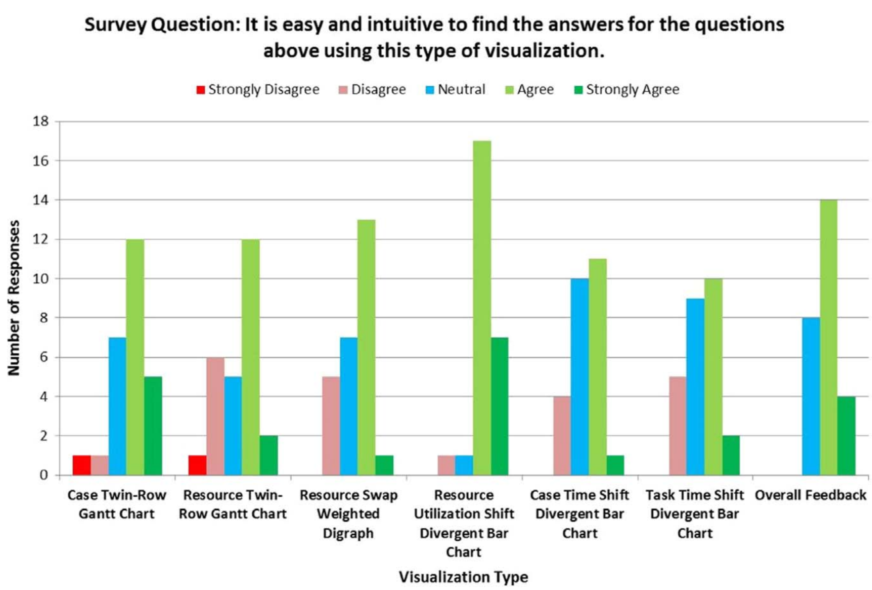


Fig. 21. How participants perceive intuitiveness, for the six individual visualisations and overall feedback.

forward time shift is confusing, when you are used to read from left to right". This made us think that intuitiveness is perceived differently depending on the cultural background.

Finally, analysis of the overall feedback indicates that colour scheme usage and scalability remain the main concerns for the visualisations. Although there are improvements to be made, the feedback indicates that the visualisation is interesting, useful, and has the potential to promote business process improvement activities.

Based on the feedback, a number of improvements were made to the visualisations:

- **Resource utilisation shift divergent bar chart** – the colours used to represent the change in resource utilisation were switched around, using the colour green to represent an increase in resource utilisation, and red to represent a decrease in resource utilisation.
- **Resource twin-row chart** – in the visualisation, the option to change the colour representation of the activity bars is enabled, allowing users to view the resource stability of activities, as well as of the tasks of the activities.
- **All visualisations** – labelling changes were made to enhance comprehension of visualisation elements (e.g., edge labels were repositioned and emphasised for easier viewing).

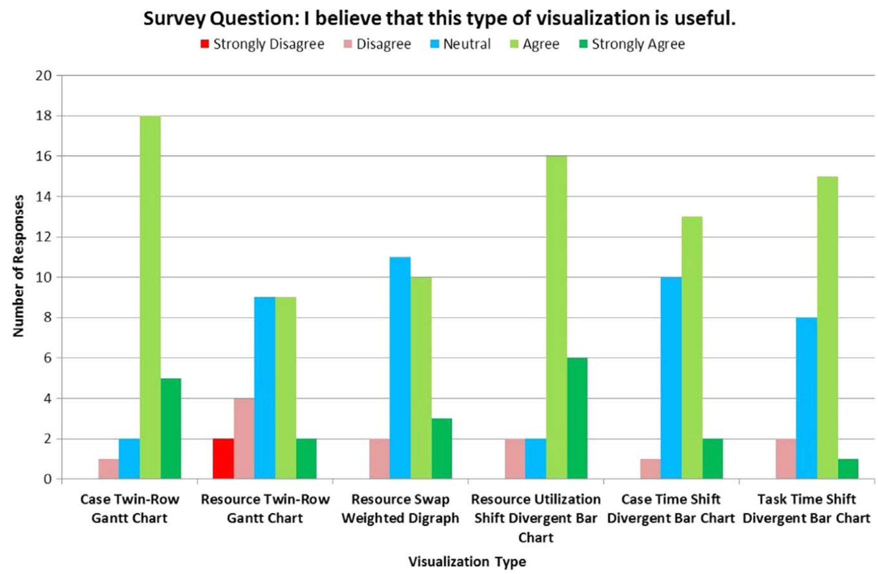


Fig. 22. How participants perceive usefulness, for the six individual visualisations.

5.2. Evaluation and future improvements

For most questions, participants managed to get the correct answer by studying the visualisations. The fact that over 85% of the questions are answered correctly reflected that the visualisations did what they are designed to do. It was also observed that participants tend to prefer simple visualisations that are easy to understand, hence the favourable results for the resource utilisation shift divergent bar chart.

We also point out potential biases and limitations with respect to the presented user evaluations. First, despite the low number of participants, all of them have prior experience in the business process management area, hence they are the kind of users for whom the proposed visualisations are intended. In addition, the level of participant involvement strengthens the evaluation result, as over 50% of the participants provided additional comments and improvement suggestions for the open-ended questions. Second, although the survey was distributed widely (BPM practitioners, academics, and students), many of our respondents identified themselves as having an academic background. This may have introduced some participant-selection bias, potentially resulting in the views of practitioners being under represented. However, as many BPM academics are themselves involved in conducting process improvement case studies and presenting their findings to stakeholders, we believe their views could be used as proxies for the lack of practitioners' input. Third, the survey did not showcase the interactive nature of the prototype tool. For instance, actions such as hovering over or double clicking certain graphical icons, changing colours, filtering, and other parameter settings have not been evaluated. Actions such as ensuring the participants' anonymity, as well as providing introductory statements for the visualisations, were taken to reduce or mitigate leniency biases and item complexity and/or ambiguity biases respectively. In addition, other response bias that may always be present in such evaluation method also applies [55], as participants are naturally subjected to different cognitive and socio-historical background.

Further improvements that could be made to the visualisations based on the feedback received were noted. A more detailed, lay person's explanation is desirable, as some participants did not understand certain concepts. The usage of certain colour schemes, and the scalability of the visualisations need to be investigated further. Additional functions (e.g., tooltip, double clicking actions, etc.) should be implemented as well to further promote the usability and comprehension of the visualisations.

In summary, it is believed that the evaluation clearly assessed the validity of the developed visualisations, the purpose of which is to gain insights into the resource and timing differences between two event logs in order to further promote BPI activities.

6. Conclusion

This study developed visualisation techniques to facilitate targeted analysis of resource reallocation and activity rescheduling between two event logs. Two executions of the same process are compared using a bottom up approach to develop insights into evidence-based improvements. Appropriate and specialised visualisation techniques were utilised to portray the differences effectively, where analysts can efficiently analyse.

Further improvements to the proposed visualisation approach can be investigated. The scalability of the visualisation approach can be enhanced by pre-processing event logs. A more in-depth investigation into the usage of colours could be carried out, with the aim of tailoring visualisations to cater for colour-blindness, cultural background, and many more variables. Continual improvements in scalability, usability, and comprehensibility of the visualisation will always be sought to provide stakeholders with additional, useful insights.

With existing techniques, information on the efficiency of the business process is explored by looking at the cost and some quality measures (such as average case duration and resource utilisation). It is now possible to go one level down to identify the changes that contribute to a cheaper and more efficient business process. This work is a first of its kind, where knowledge from visual analytics was used to identify and visualise the differences between two event logs. Commercial stakeholders are also able to further investigate and initiate BPI activities based on insights gathered from the visualisations. For future work, an in-depth investigation of changes that contribute towards a number of different business goals could be performed. Stakeholders can then receive recommendations on business process improvement best practices and opportunities, which include but are not limited to redesign of organisational structure and improvement of business process operations (e.g., those proposed by Mansar and Reijers [1]).

References

[1] S. Mansar, H. Reijers, Best practices in business process redesign: use and impact, Bus Process Manag. J. 13 (2) (2007) 193–213.
[2] R. Neuman, R. Cavanagh, The Six Sigma way: how GE, Motorola, and Other Top Companies are Honing Their Performance, McGraw Hill Professional, 2000.
[3] J. Womack, D. Jones, Lean Thinking: banish Waste and Create Wealth in Your

Corporation, Simon and Schuster, 2010.
[4] I. Masaaki, Kaizen: The Key to Japan's Competitive Success, New York, McGraw-Hill.
[5] W. van der Aalst, H. Reijers, A. Weijters, B. van Dongen, A. de Medeiros, M. Song, H. Verbeek, Business process mining: an industrial application, Inf. Syst. 32 (5) (2007) 713–732 [URL ⟨http://140.118.1.157/cht/Research%20Pro/Process%20Mining/Business%20process%20mining_An%20industrial%20application.pdf⟩].
[6] IEEE Task Force on Process Mining, Process Mining Case Studies, ⟨http://www.win.tue.nl/ieeetfpm/doku.php?Id=shared:process_mining_case_studies⟩, 2015.
[7] W. Low, J.D. Weerdt, M. Wynn, A. ter Hofstede, W. van der Aalst, S. vanden Broucke, Perturbing event logs to identify cost reduction opportunities: a genetic algorithm-based approach, in: Evolutionary Computation (CEC), 2014 IEEE Congress on, IEEE, 2014, pp. 2428–2435.
[8] D. Keim, G. Andrienko, J. Fekete, C. Görg, J. Kohlhammer, G. Melançon, Information visualization: human-centered issues and perspectives, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, Ch. Visual Analytics: Definition, Process, and Challenges, pp. 154–175. http://dx.doi.org/10.1007/978-3-540-70956-5_7.
[9] M. Netjes, Process improvement: the creation and evaluation of process alternatives (Ph.D. thesis), Eindhoven University of Technology, 2010.
[10] H. Reijers, S. Mansar, Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics, Omega 33 (4) (2005) 283–306.
[11] M. Jansen-Vullers, M. Netjes, H. Reijers, M. Stegeman, A redesign framework for call centers, in: Business Process Management, 2006, pp. 306–321.
[12] M. zur Muehlen, D. Ho, Service process innovation: a case study of BPMN in practice, in: Proceedings of the 41st Hawaii International Conference on Systems Science (HICSS-41 2008), Proceedings, 7-10 January 2008, Waikoloa, Big Island, HI, USA, IEEE, 2008, pp. 372–372.
[13] M. Jansen-Vullers, P. Kleingeld, M. Loosschilder, M. Netjes, H. Reijers, Trade-offs in the performance of workflows-quantifying the impact of best practices, Bus Process Manag. Work 4928 (2008) 108–119.
[14] W. van der Aalst, Process-aware information systems: lessons to be learned from process mining, in: K. Jensen, W. van der Aalst (Eds.), Transactions on Petri Nets and Other Models of Concurrency II, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 1–26. http://dx.doi.org/10.1007/978-3-642-00899-3_1.
[15] IEEE Task Force on Process Mining, Process mining manifesto, in: F. Daniel, S. Dustdar, K. Barkaoui (Eds.), BPM 2011 Workshops, Vol. 99 of Lecture Notes in Business Information Processing, Springer-Verlag, Berlin, 2011, pp. 169–194.
[16] W. van der Aalst, M. de Leoni, A. ter Hofstede, Process mining and visual analytics: breathing life into business process models, BPM Center Report BPM-11-15, BPMcenter.org, vol. 17, 2011, pp. 699–730.
[17] H. Verbeek, J. Buijs, B. van Dongen, W. van der Aalst, ProM 6: The process mining toolkit, Proceedings of BPM Demonstration Track, vol. 615, 2010, pp. 34–39.
[18] H. Schonenberg, B. Weber, B. van Dongen, W. van der Aalst, Supporting flexible processes through recommendations based on history, in: Business Process Management, Springer, 2008, pp. 51–66.
[19] W. van der Aalst, K. van Hee, Workflow management: models methods, and systems, The MIT Press, 2004.
[20] S. Chaudhuri, U. Dayal, An overview of data warehousing and OLAP technology, ACM Sigmod record, vol. 26 (1), 1997, pp. 65–74.
[21] W. van der Aalst, Process cubes: slicing, dicing, rolling up and drilling down event data for process mining, in: Asia Pacific Business Process Management, Springer, 2013, pp. 1–22.
[22] W. van der Aalst, A. de Medeiros, A. Weijters, Process equivalence: comparing two process models based on observed behavior, in: Business Process Management, Springer, 2006, pp. 129–144.
[23] J. Buijs, M.L. Rosa, H. Reijers, B. van Dongen, W. van der Aalst, Improving business process models using observed behavior, in: Data-Driven Process Discovery and Analysis, Springer, 2013, pp. 44–59.
[24] A. Rozinat, W. van der Aalst, Conformance testing: measuring the fit and appropriateness of event logs and process models, in: Business Process Management Workshops, Springer, 2006, pp. 163–176.
[25] W. van der Aalst, W. Low, M. Wynn, A. ter Hofstede, Change your history: learning from event logs to improve processes, in: G. Fortino, W. Shen, J. Barthes, J. Lou, W. Li, S. Ochoa, M. Abel, A. Guerrieri, M. Ramos (Eds.), Proceedings of the 19th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD 2015), IEEE Computer Society Press, 2015, pp. 7–12.
[26] B. Shneiderman, The Eyes have it: a task by data type taxonomy for information visualizations, in: Proceedings of the IEEE Symposium on Visual Languages, 1996, pp. 336–343. http://dx.doi.org/10.1109/VL.1996.545307.
[27] I. Vessey, Cognitive fit: a theory-based analysis of the graphs versus tables literature, Decis. Sci. 22 (2) (1991) 219–240.
[28] I. Vessey, D. Galletta, Cognitive fit: an empirical study of information acquisition, Inf. Syst. Res. 2 (1) (1991) 63–84.
[29] N. Andrienko, G. Andrienko, Exploratory analysis of spatial and temporal data, Springer, 2006.
[30] I. Herman, G. Melançon, M. Marshall, Graph visualization and navigation in information visualization: a survey, IEEE Trans. Vis. Comput. Graph. 6 (1) (2000) 24–43.
[31] W. van der Aalst, H. Reijers, M. Song, Discovering social networks from event logs, Comput. Support. Coop. Work (CSCW) 14 (6) (2005) 549–593.
[32] W. Aigner, S. Miksch, H. Schumann, C. Tominski, Visualization of time-oriented data, Springer Science & Business Media, 2011.
[33] G. Chin, M. Singhal, G. Nakamura, V. Gurumoorthi, N. Freeman-Cadoret, Visual analysis of dynamic data streams, Inf. Vis. 8 (3) (2009) 212–229.
[34] M. de Leoni, F. Maggi, W. van der Aalst, An alignment-based framework to check the conformance of declarative process models and to preprocess event-log data, Inform. Syst.
[35] M. de Leoni, S. Suriadi, A. ter Hofstede, W. van der Aalst, Turning event logs into process movies: animating what has really happened, Softw. Syst. Model. (2014) 1–26.
[36] H. Reijers, M. Song, B. Jeong, Analysis of a collaborative workflow process with distributed actors, Inf. Syst. Front. 11 (3) (2009) 307–322.
[37] W. van der Aalst, M. Schonenberg, M. Song, Time prediction based on process mining, Inf. Syst. 36 (2) (2011) 450–475.
[38] J. Scott, Social network analysis: developments, advances, and prospects, Soc. Netw. Anal. Min. 1 (1) (2011) 21–26.
[39] A. Koschmider, M. Song, H. Reijers, Social software for business process modeling, J. Inf. Technol. 25 (3) (2010) 308–322.
[40] M. Song, W. van der Aalst, Towards comprehensive support for organizational mining, Decis. Support Syst. 46 (1) (2008) 300–317.
[41] J. Claes, I. Vanderfeesten, J. Pinggera, H. Reijers, B. Weber, G. Poels, A visual analysis of the process of process modeling, Inf. Syst. e-Bus Manag. (2013) 1–44.
[42] K. Wongsuphasawat, J.G. Gómez, C. Plaisant, T. Wang, M. Taieb-Maimon, B. Shneiderman, LifeFlow: visualizing an overview of event sequences, in: Proceedings of the SIGCHI conference on human factors in computing systems, ACM, 2011, pp. 1747–1756.
[43] K. Wongsuphasawat, D. Gotz, Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization, IEEE Trans. Vis. Comput. Graph. 18 (12) (2012) 2659–2668.
[44] A. Perer, F.Wang, Frequence: Interactive mining and visualization of temporal frequent event sequences, in: Proceedings of the 19th international conference on Intelligent User Interfaces, ACM, 2014, pp. 153–162.
[45] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, A.Wilson, MatrixWave: visual comparison of event sequence data, in: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, ACM, 2015, pp. 259–268.
[46] S. Kriglstein, S. Rinderle-Ma, Change visualization in business processes-requirements analysis, in: International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications, vol. 02, 2012, pp. 1–10.
[47] S. Kriglstein, G. Wallner, S. Rinderle-Ma, A visualization approach for difference analysis of process models and instance traffic, in: Business Process Management, Springer, 2013, pp. 219–226.
[48] J. Buijs, H. Reijers, Comparing business process variants using models and event logs, in: Enterprise, Business-Process and Information Systems Modeling, Springer, 2014, pp. 154–168.
[49] R. Mans, M. Schonenberg, M. Song, W. van der Aalst, P. Bakker, Application of process mining in healthcare–a case study in a dutch hospital, in: Biomedical Engineering Systems and Technologies, Springer, 2008, pp. 425–438.
[50] R. Mans, W. van der Aalst, R. Vanwersch, A. Moleman, Process mining in healthcare: data challenges when answering frequently posed questions, in: Process Support and Knowledge Representation in Health Care, Springer, 2013, pp. 140–153.
[51] R. Christ, Review and analysis of color coding research for visual displays, Hum. Factors: J. Hum. Factors Ergon. Soc. 17 (6) (1975) 542–570.
[52] S. Silva, B. Santos, J. Madeira, Using color in visualization: a survey, Comput. Graph. 35 (2) (2011) 320–333.
[53] T. Murata, Petri nets: properties, analysis and applications, Proc. IEEE 77 (4) (1989) 541–580.
[54] F. Davis, Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS Q. (1989) 319–340.
[55] P. Podsakoff, S. MacKenzie, J. Lee, N. Podsakoff, Common method biases in behavioral research: a critical review of the literature and recommended remedies, J. Appl. Psychol. 88 (5) (2003) 879.