Identification and analysis of handovers in organisations using process model repositories

Article *in* Business Process Management Journal · February 2020

DOI: 10.1108/BPMJ-01-2019-0041

3 authors:

Michael Leyer
Philipps University of Marburg
127 PUBLICATIONS **974** CITATIONS

SEE PROFILE

Deniz İren
Center for Actionable Research of the Open University
48 PUBLICATIONS **223** CITATIONS

SEE PROFILE

Banu Aysolmaz
Eindhoven University of Technology
53 PUBLICATIONS **303** CITATIONS

SEE PROFILE

Identification and analysis of handovers in organisations using process model repositories

Michael Leyer

Institute for Business Administration, Universitat Rostock, Rostock, Germany and School of Management, Queensland University of Technology, Brisbane, Australia

Deniz Iren

Center for Actionable Research, Open Universiteit, Heerlen, The Netherlands

Banu Aysolmaz

Maastricht University, Maastricht, The Netherlands and

Department of Industrial Engineering and Innovation Sciences, TU Eindhoven, Eindhoven, The Netherlands

1 Introduction

Handovers are important when it comes to designing efficient production processes, especially service delivery processes. On the one hand, handovers are necessary to benefit from expert knowledge or specific functions of machines and systems via labour division; on the other, each handover has the disadvantage that it requires an investment of time into communication, potential misunderstandings or idle times when the receiver is busy with other work (Womack and Jones, 2003; Daft *et al.*, 2007). In order to balance these advantages and disadvantages to achieve optimal process efficiency, handovers have to be identified and reduced throughout the whole organisation (Pentland *et al.*, 2017), as employees, machines, systems and processes are connected in multiple ways (Leyer and Moormann, 2012).

Information on handovers related to processes can be retrieved from various sources, including questioning employees (Leyer *et al.*, 2017), and more objective data sources related to business processes, such as process models (Aysolmaz and Reijers, 2017) and workflow management systems (Russell *et al.*, 2005). Event logs from workflow management systems bear the advantage that they reflect daily work practices more accurately than process models do. However, not all processes and activities are implemented in a workflow management system, even when the organisation uses such a system to perform its processes (Dumas *et al.*, 2005).

Process models are good candidates by which to objectively identify relations between roles and units, as such models are a structured source of activities and the organisational constituents that perform them (Hong *et al.*, 2012). Organisations typically store their process-related information in process model repositories, which may contain hundreds of process models (Rosemann, 2006). Furthermore, the process model hierarchy – i.e. the relations between process models in a repository – is a potent source of knowledge for hidden relations among roles and units (Turetken *et al.*, 2016). However, since process models are primarily designed based on the flow between activities, additional analysis is required in order to reveal these relations. Moreover, due to the high number of process models, it is challenging to perform such analysis in a manual way. We therefore pose the following research question: How can the handover density be identified using process model repositories?

This paper addresses shortcomings of prior research by proposing a novel method by which to objectively and practically identify the handover density in an organisation. We apply our method to the process model repository of a major telecommunications company with a set of 1,012 process models to show the method's feasibility. Our results contribute to research on information systems by showing how handovers in organisations can be identified in order to inform relevant management actions in organisations.

2 Theoretical background

2.1 The relevance of handovers in process execution

Organisations are characterised by a number of employees working together in the same legal entity (Jones, 2010). Employees are connected in terms of processes, as well as hierarchies (Segatto *et al.*, 2013). The processes describe how employees work together to generate products and services (Bitici *et al.*, 2011), while hierarchies assign employees to organisational units and define supervisory roles (Jones, 2010). As there is typically a division of labour in executing processes, handovers occur between employees working together in processes. While positive effects arise from labour division, such as specialisation and learning effects, handovers entail negative effects, such as the potential for miscommunication and misunderstanding, and additional time spent by the receiving employee (Bronzo *et al.*, 2013; Škrinjar and Trkman, 2013). The time spent by the employee incorporates getting familiar with the case or product that has been handed over, or lead to quality issues with the final products and services. Thus, balancing both negative and positive effects is important to achieve process efficiency, but also flexibility to allow for necessary changes (Pentland *et al.*, 2017). Companies that want to balance handovers have to consider the grouping of employees according to functions and processes, and the roles (of employees with certain task responsibilities) assigned to processes and activities (Leyer *et al.*, 2017).

2.2 Process models including the organisational perspective

Process models capture information on process activities, the flow relation between those activities, organisational resources that perform the activities, and information artefacts that are used and produced by the activities (Curtis *et al.*, 1992). The information provided by process models on organisational resources – the organisational perspective – is used in a variety of ways in organisations, such as in learning responsibilities (Davies *et al.*, 2006), managing resources (Browning, 2010) and allocating resources to tasks (Schefer-Wenzl and Strembeck, 2014).

In process models, the organisational perspective includes roles, organisational units (hereinafter referred to as units) and systems involved to perform the activities (Davis and Brabander, 2007). Despite differences in the representation styles of modelling languages, they support process model users in conveying this organisational knowledge in a similar way (List and Korherr, 2006). An example process model with activities and organisational resources is depicted in Figure 1 using e-EPC notation (Davis and Brabander, 2007). As shown in Figure 1, an activity may be performed by one role (Activity 1), in a fully automated way by a system (Activity 2), by a role using a system (Activity 3), by two or more roles in a collaborative way (Activity 4), or by a unit (Activity 5). The "+" sign on Activity 5 indicates that it is a high-level activity, and its details are modelled as a sub-process. The flow between activities is identified by gateways. For example, Activities 1 and 2 both have to be completed before Activity 3 can be performed, as indicated by two AND (∧) gateways – the first of which is a split and the second a join gateway. Activities 4 and 5 are exclusive since they are enclosed by split and join XOR (X) gateways.

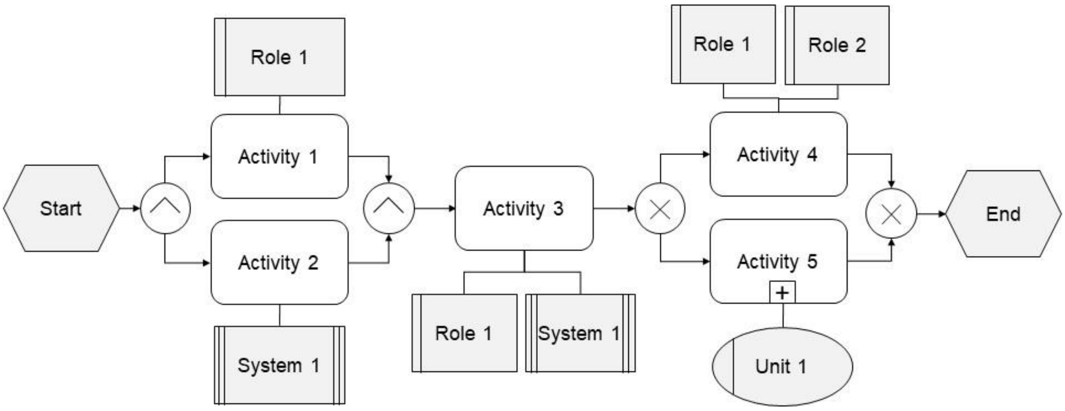


Figure 1: Exemplary process model with activities and organisational resources

Focusing on the activities, the flow between the activities and the organisational resources in a process, we define a process model as follows:

Definition 1: A tuple $P = (A, G, E, F, OE, p, q, t)$ is a *process model*, where:

- A is a finite non-empty set of activities;
- G is a finite set of gateways;
- $E = ES \cup EE \cup EN$ is a finite non-empty set of events composed of the set of start events ES, end events EE and intermediate events EN;
- $\mathrm{A} \cap G = \emptyset$, $A \cap E = \emptyset$, $G \cap E = \emptyset$ and $N = A \cup G \cup E$ is a finite set of nodes;
- $F \subseteq N \times N$ is a set of sequence flows. Each sequence flow $f = (n_x, n_y) \in F$ represents a directed edge between two nodes n_x and n_y that are connected to each other;

- $OE = R \cup OU$ is a finite non-empty set of organisational resources composed of the set of roles R and units OU;
- p: $R \rightarrow a \in A$ is a mapping function that associates a set of roles R with an activity a;
- q: $OU \rightarrow a \in A$ is a mapping function that associates a set of units OU with an activity a;
- t: $G \rightarrow \{split, join\}$ is a mapping function that associates each gateway with a type.

Organisations create process model repositories to store their process information. These repositories contain a number of process models that are hierarchically related to each other. We define such a process model collection as follows:

Definition 2: A tuple $PMR = (PM, A_{pmr}, R_{pmr}, OU_{pmr}, r, sp)$ is a *process model repository*, where:

- PM is a non-empty finite set of process models with elements $P_i = (A_i, G_i, E_i, F_i, OE_i, p_i, q_i, r_i, t_i)$, where $i = 1, 2, \ldots, |PM|$.
- $A_{pmr} = \cup_{i=1, 2, \ldots, |PM|} A_{\mathrm{i}}$ is the set of all activities in the process model repository.
- $R_{pmr} = \cup_{i=1, 2, \ldots, |PM|} R_{\mathrm{i}}$ is the set of all roles in the process model repository.
- $OU_{pmr} = \cup_{i=1, 2, \ldots, |PM|} OU_{\mathrm{i}}$ is the set of all units in the process model repository.
- r: $r \in R_{pmr} \rightarrow ou \in OU_{pmr}$ is a mapping function that associates a role r with an organizational unit ou;
- sp: $a_i \in A_i \rightarrow P_j \in PM$ is a sub-process mapping from an activity a_i that is an element of P_i to another process P_{j}.

Handovers in process models – i.e. cases in which two subsequent activities are performed by employees with different roles – are major causes of communication problems and errors in processes (Weske, 2012). Two activities that follow, or may follow, each other entail a potential handover, which can then indicate an actual handover; these relations are defined below:

Definition 3: $PH \subseteq A_{pmr} \times A_{pmr}$ is the potential handover relation. Each $ph = (a_x, a_y)$ represents a directed edge between two activities a_x and a_y that *may* follow each other.

Definition 4: $H \subseteq PH$ is the handover relation. Each $h = (a_x, a_y)$ represents a directed edge between two activities a_x and a_y that are performed by a different set of roles/units identified with p, q and r mappings.

2.3 Identifying handovers

Using process models is another relevant type of approach to identifying handovers; however prior work (discussed in the following) has not focused on this. In this regard, we carefully and systematically analysed prior literature in the scientific databases. Koschmider *et al.* (2009) generated social networks from process models. The nodes represent the people and the business units, while the edges represent relationships such as transfer of work, subcontracting and cooperation. (Koschmider *et al.*, 2009) used the social network with the purpose of making recommendations for process modelling.

Hong *et al.* (2012) used process models to construct social networks. Using social network analysis, they recommended organisation structure changes, tackling problems such as revealing the verticality of workflows (hierarchy of relationships) and how well the organisation structure fits to foster collaboration among the workers and business units.

Next to using process models, employees can also be asked about handovers in their processes by means of surveys (Leyer *et al.*, 2017). Therein, employees have to answer questions regarding the number of interfaces or employees involved in a process in general, rather than naming specific handovers.

Summing up, extant approaches have not considered process models and determined handovers within the whole organisation. Thus, in the following section, we describe a methodology to analyse different cases in process models so as to determine handovers in an objective way.

3 Method

Our methodology has four main steps (Figure 2). First, using the process model repository as input we perform pre-processing to extract the set of process models, their elements and mappings. Next, we identify the potential handovers *PH* within each process model and between the process models based on the hierarchical structure of the repository. Then, by processing this information, we extract actual handovers *H* observed in the processes. Lastly, all information extracted is used to calculate the metrics to identify the handover density for the organisation. The details of each step are explained in the following sections.

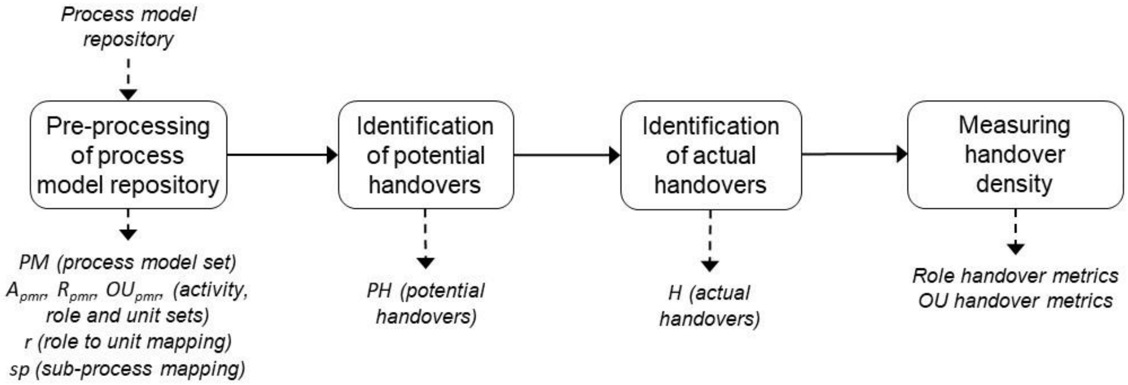

Process model repository
Pre-processing of process model repository
Identification of potential handovers
Identification of actual handovers
Measuring handover density
PM (process model set)
A_{pmr}, R_{pmr}, OU_{pmr} (activity, role and unit sets)
r (role to unit mapping)
sp (sub-process mapping)
PH (potential handovers)
H (actual handovers)
Role handover metrics
OU handover metrics

Figure 2: Overview of the method

The process model repository structure is critical for our method in terms of identifying handovers between processes. Process models in a repository are bound together with hierarchical relations. The hierarchical structure is achieved through the use of sub-processes, where an activity at the high-level process refers to a sub-process (as in Activity 5 in Figure 1) and the sub-process is depicted in another process model (Dijkman *et al.*, 2012). An example process model repository structure is shown in Figure 3, where process hierarchy is established by decomposing processes of the organisation and using sub-processes (Davies *et al.*, 2006; Aysolmaz and Demirörs, 2015). Our method identifies handovers in process models, which can then be added up within the hierarchy of the repository structure.

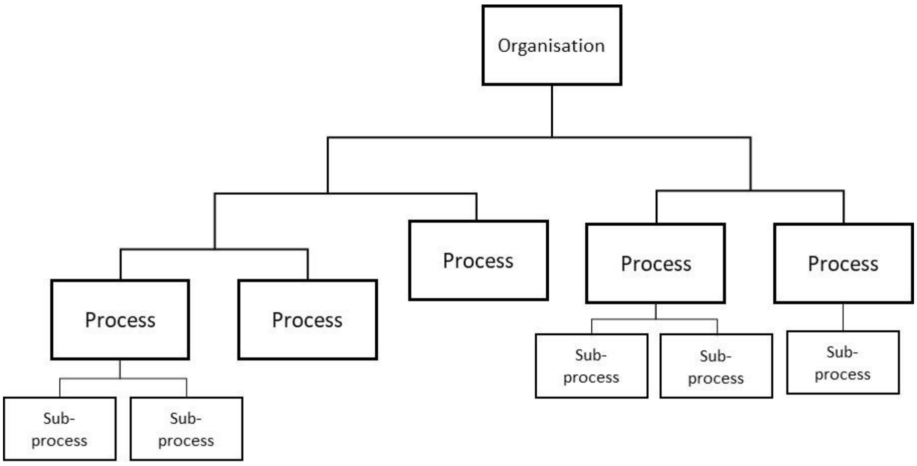


Figure 3: An exemplary process model repository structure

3.1 Step 1: Pre-processing

In this phase, we examine the process model repository *PMR* to extract the information identified in Definitions 1, 2 and 3. For each process model in the repository P_i the set of process model activities A_i, gateways G_i, events E_i and organisational elements OE_i (roles, units and systems) are identified. Based on the connections between A_i, G_i and E_i, the flow relation F_i is constructed. For the sake of simplicity, we construct F_i by removing each intermediate event en_i and placing a direct connection between the element preceding and following it, since those events do not affect how handovers are identified. Using the connections from OE_i to A_i, the mappings *p* and *q* are identified. Handovers between people and units are the most important handover types (Škrinjar and Trkman, 2013). We disconnect the completely automated activities from the flow relation, as we do for intermediate events, since they do not cause handovers. The sub-process mapping information *sp* is assigned when an activity is encountered in the process model that refers to another process P_{j} as its sub-process. Lastly, based on the organisational structure information that is frequently stored together with a process model repository (Iren and Reijers, 2017), the mapping *r* is assigned between the roles and their related unit.

3.2 Step 2: Identification of potential handovers

In identifying the handover density, it is important that the method recognises potential handovers both within and between process models. The subsections below present the approach for within a process, and then between processes.

3.2.1 Step 2.1: Identification of potential handovers within a process

In a process model, the way in which activities and gateways are connected to each other identifies the process flow, and thus the activities that may follow each other. Six different patterns of these connections can be identified, each of which require the flow to be handled differently. We present these patterns, or different process structures that may result in the existence of a handover *PH* between two activities, a_x and a_y, in Figure 4. We analyse these patterns to derive the set of potential handovers within a process, $PHW_i \subseteq PH$.

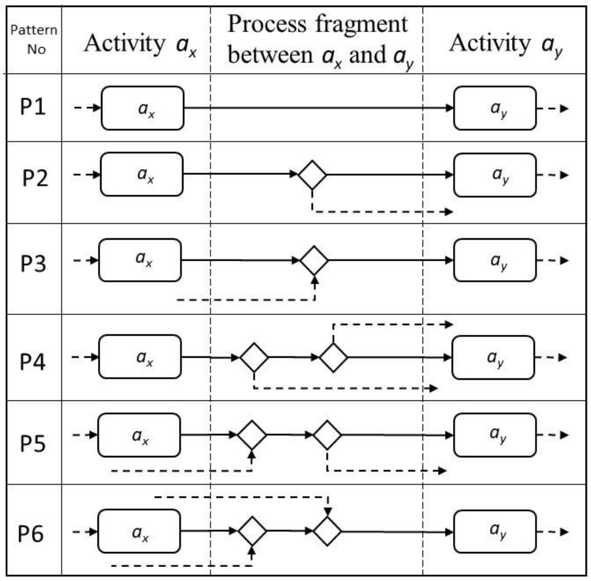


Figure 4: Patterns of process fragments between two activities that may follow each other

Pattern 1 represents the most straightforward case, in which an activity is directly followed by another activity. We identify the set of potential handovers conforming to *Pattern 1* within a process model P_i, namely $PHWP1_i$, based on the following set of operations:

$$\forall f \in F_i, f = (n_x, n_y)$$

$$PHWP1_i = \{f \colon f \in F_i \mid n_x \in A_i \;\&\; n_y \in A_i\} \qquad (1)$$

Pattern 2 is observed when an activity, a_x, is followed by a split gateway g_x. When there is either a parallel or an exclusive split gateway, for all the paths of this gateway there is a potential handover from a_x to the set of activities A_{iy} following this g_x. More specifically, for a parallel gateway such a potential handover exists in all instances of this process, whereas for an exclusive gateway a handover for each splitting path of this gateway is observed in some instances. Since all types of gateway create a potential handover, we treat them in the same way. Thus, we identify the set of potential handovers conforming to *Pattern 2* for a process model P_i, namely $PHWP2_i$, based on the following set of operations:

$$\forall\ f_n, F_m \in F_i,\ f_n = (a_x, g_x),\ F_m = (g_x, A_{iy})$$

$$t(g_x) = split,\ |A_{iy}| > 1,$$

$$PHWP2_i = \{ph\text{: } ph = (a_x, a_{iy}) \mid \forall a_{iy} \in A_{iy}\} \qquad (2)$$

Pattern 3 is the case in which a set of activities A_{ix} is followed by a join gateway g_x connected to one activity a_y. In this case, we identify a potential handover between each activity a_{ix} in the set A_{ix}, and the activity following the join gateway, a_y. As in the second pattern, we treat the parallel and exclusive gateways in the same way. Then, the set of potential handovers conforming to *Pattern 3* for a process model P_i, namely $PHWP3_i$, is identified with the following set of operations:

$$\forall\ F_n, f_m \in F_i,\ F_n = (A_{ix}, g_x),\ f_m = (g_x, a_y)$$

$$t(g_x) = join,\ |A_{ix}| > 1,$$

$$PHWP3_i = \{ph\text{: } ph = (a_{ix}, a_y) \mid \forall a_{ix} \in A_{ix}\} \qquad (3)$$

Patterns 4, *5* and *6* depict the more complex cases of an activity followed by a gateway, wherein that gateway is also followed by other gateways of the same or different split/join types. In a process model, there may be an indefinite number of gateways until another activity is reached. To extract the potential handovers for process model fragments following these patterns, we apply similar set of operations for *Patterns 2 and 3*, considering transitional connections from one activity to the next and only changing our treatment to consider multiple gateways. As a result, we obtain the values $PHWP4_i$, $PHWP5_i$ and $PHWP6_i$ for a process model P_i. To identify the set of potential handovers within a process model emerging through all patterns, we get the union of the individual sets with the following set of operations:

$$PHW_i = PHWP1_i \cup PHWP2_i \cup PHWP3_i \cup PHWP4_i \cup PHWP5_i \cup PHWP6_i \quad (4)$$

Lastly, we obtain the set of potential handovers within processes in the whole repository with the following set operations:

$$\mathrm{PHW} = \cup_{\mathrm{i}=1,\,2,\,\ldots,\,|\mathrm{PM}|}\, \mathrm{PHW}_\mathrm{i} \qquad (5)$$

3.2.2 Step 2.2: Identification of potential handovers between processes

The next step in detecting potential handovers between processes is on the level of multiple processes that are connected to each other with the process hierarchy. The aim is to identify handovers by detecting the last and first activity that are connected with the activity referring to a sub-process in the main process, and the respective sub-process identified by the mapping *sp*. This operation is called flattening, which enables us to investigate the flow of activities for a process and its sub-process as if they are on the same hierarchical level (Turetken *et al.*, 2016). On the left side of Figure 5 we examine the possible fragments, which include the activity a_{iy} mapping to another process P_j in the main process P_i and the nodes preceding that; the right side shows the start fragment of the sub-process P_j until an activity begins. Similarly, Figure 6 depicts the process fragment patterns of the end of a sub-process P_j and the process P_i, including its nodes a_{iy} mapping to the sub-process P_j and the following nodes.

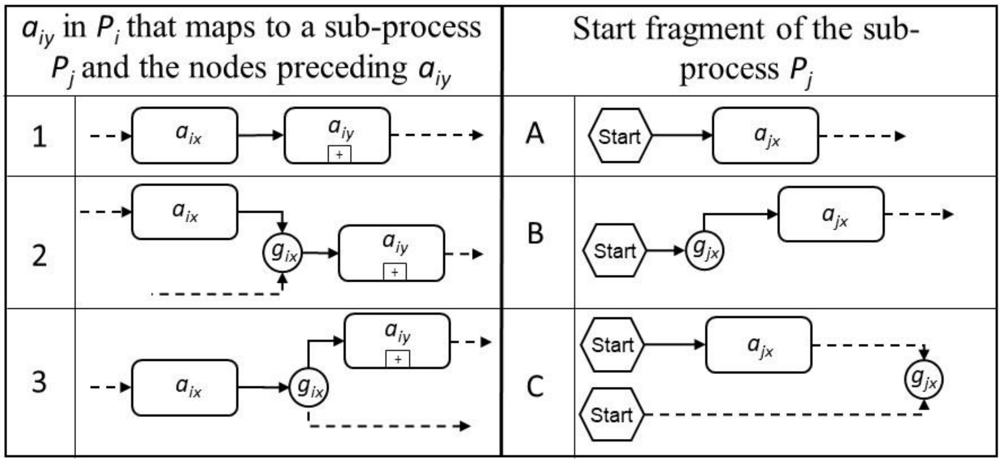


Figure 5: Process fragment patterns before a high-level activity and the start of the sub-process

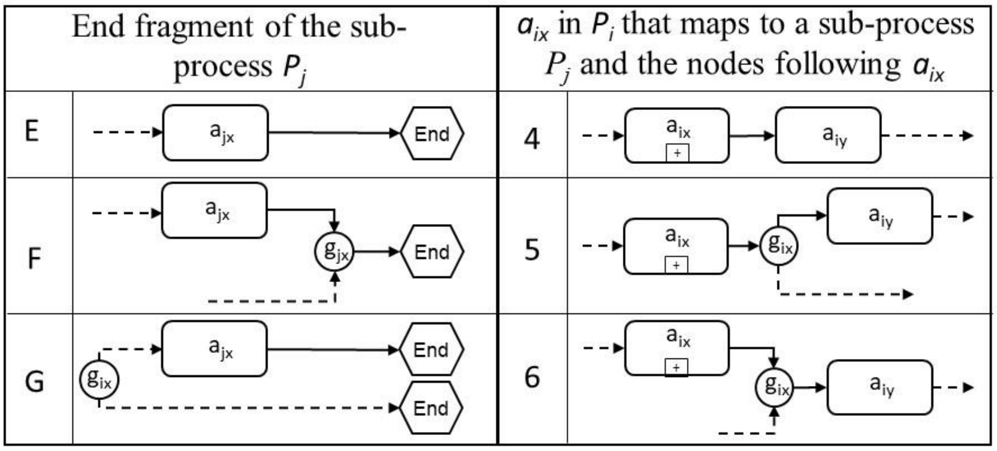


Figure 6: Process fragment patterns at the end of the sub-process and after a high-level activity

The combinations of these process fragment patterns correspond to one of the patterns in Figure 4 when the process and its sub-process are flattened. For each possible combination of these fragments, the corresponding flattened pattern defined in Figure 4 is as follows:

- 1-A; E-4 : P1
- 1-B; 1-C; 3-A; E-5 : P2
- 2-A; E-6; F-4; G-4 : P3
- 3-B; 3-C : P4
- 2-B; 2-C; F-5; G-5 : P5
- F-6; G-6 : P6

For all sub-processes in a process model, upon identification of the corresponding patterns, set of operations 1 to 3 outlined in Section 3.2.1 for identification of potential handovers are used in the same manner. Thus, we identify the set of potential handovers between a flattened version of a process P_i and the start of the sub-process P_j (Figure 5), $PHB1_i \subseteq PH$, and the end of the sub-process $PHB2_i \subseteq PH$ (Figure 6), by using the following set operations:

$$PHB_i = PHB1P1_i \cup PHB1P2_i \cup PHB1P3_i \cup PHB1P4_i \cup PHB1P5_i \cup PHB1P6_i \cup PHB2P1_i \cup PHB2P2_i \cup PHB2P3_i \cup PHB2P4_i \cup PHB2P5_i \cup PHB2P6_i \qquad (6)$$

The set of all potential handovers between process models is obtained by adding up individual sets obtained for each process model, as in the following set operations:

$$PHB = \cup_{I = 1, 2, \ldots, |PM|} PHB_i \qquad (7)$$

Lastly, we gather the complete set of potential handovers by combining the sets for within- and between-process models, as in the following set operations:

$$PH = PHW \cup PHB \qquad (8)$$

3.3 Step 3: Identification of actual handovers

In this step, based on the set of potential handovers *PH* the method identifies the actual handovers. We can distinguish between the handovers taking place among different roles and among roles that belong to different units. Based on Definition 4 of a handover (a_x, a_y), we identify the set of handovers according to the following three set operations:

The set of role handovers (*Hr*) in a process model repository *PMR*:

$$p(a_x) \neq \varnothing, p(a_y) \neq \varnothing, Hr = \{ph: ph = (a_x, a_y) \mid p(a_x) \neq p(a_y)\} \qquad (9)$$

The set of OU handovers (*Hou*) in a process model repository *PMR*:

$$p(a_x) \neq \varnothing, p(a_y) \neq \varnothing, Hou = \{ph: ph = (a_x, a_y) \mid (\mathrm{r}(p(a_x)) \cup q(a_x)) \neq (\mathrm{r}(p(a_y)) \cup q(a_y))\} \qquad (10)$$

The set of role handovers within the same OU (*Hrou*) in a process model repository *PMR*:

$$p(a_x) \neq \emptyset, p(a_y) \neq \emptyset, Hrou = \{ph: ph = (a_x, a_y) \mid p(a_x) \neq p(a_y) \;\&\; \mathrm{r}(p(a_x) = \mathrm{r}(p(a_y)\} \quad (11)$$

3.4 Measuring the handover density

According to the dimensions identified in the theoretical background section, handover density has to be analysed from the perspective of design of processes, as well as that of organisational units. Hence, from a process perspective it should be measured by the number of handovers between the roles involved (Leyer and Wollersheim, 2013). Consequently, the following types of metrics can be calculated using the data extracted via the set operations described: (1) handovers between roles in processes (role handover metrics), and (2) handovers considering organisational units in processes (OU handover metrics). Values are then calculated for each process and unit, and can be added up within the hierarchy of a process repository (Figure 7). Hence, there may be one value for the whole organisation, or sub-values grouping several processes or units together on a higher level. The number of levels depends on the hierarchical structure of processes and units.

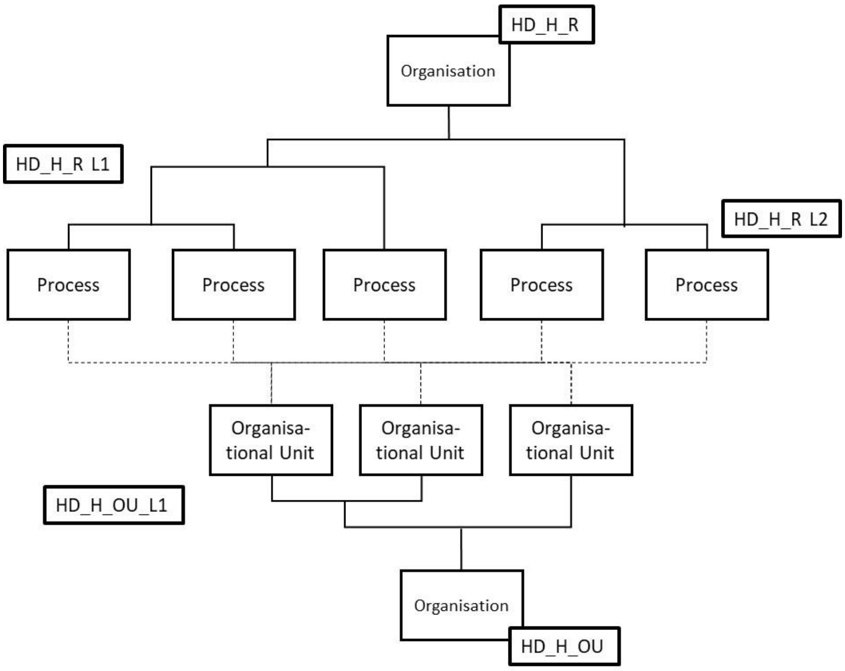


Figure 7: Measurement levels for handover density from different perspectives

3.4.1 Role handover metrics

Handovers occur in a process between the roles involved – i.e. every time an employee with a certain role hands over to another role. The handover density for a process is shown by the ratio of role handovers to the size of the process model. With this metric, an organisation can obtain a comparative view of its processes based on role handovers. The comparison can also be made among process areas. We use the number of activities as a common metric for measuring process size (Reijers and Mendling, 2011). Handover density (HD) and related metrics for role handovers are as follows:

- #HrP$_i$: Number of role handovers taking place in one process model P$_i$ (handovers in the set Hr with a_x & $a_y \in A_i$)
- HD_HrP$_i$: Role handover density for a process P$_i$, which is the ratio of the number of role handovers #HrP$_i$ to the number of activities in the process. A score of 1 indicates that one handover exists per each activity in the process.
- HD_H_R: Role handover density score for the organisation or a process set, which is the average value of HD_HrP$_i$ for all processes in the examined process set, which can be a process area or the whole repository.

3.4.2 Organisational unit handover metrics

If organisations align their organisational units with processes, then handovers in a process occur between employees that are in the same organisational unit. If the focus is more on functions, then the number of handovers between employees in different organisational units is higher. This is reflected by the metric HD_HrOU$_i$, and the related metrics as follows:

- #HrouOU$_i$: For a specific unit OU$_{i,}$, number of role handovers occurring between roles of that unit – handovers within OU$_{i,}$ (handovers in the set H$_{rou}$ with r(p(ax) = r(p(ay) = OU$_i$)
- #HouOU$_i$: Number of OU handovers that involve a specific unit OU$_i$ – handovers between OU$_{i,}$ and other units (handovers in the set H$_{ou}$ with (r($p(a_x)) \cup q(a_x))$ = OU$_i$ || (r($p(a_y)) \cup q(a_y))$ = OU$_i$)
- HD_HrOU$_i$: Handover density for a unit OU$_i$, which is the ratio of (#HrouOU$_i$ + 1) to (#HouOU$_i$ + #HrouOU$_i$ + 1). This indicates the ratio of handovers within OU$_i$ to the sum of handovers within OU$_i$ and between OU$_i$ and other units, where +1 is added to both numerator and denominator to compensate for 0 values in the numerator. HD_HrOU$_i$ score that is close to 1.0 indicates that the unit works heavily through handovers within the unit.
- HD_H_OU: OU handover density score for the organisation, which is the average of every HD_HrOU$_i$ for all OU$_i$.

3.5 Application summary

Organisations that want to apply the method need to have a repository of process models, as well as a chart of their organisational hierarchy. These can be processed using the steps of the method described above to calculate the necessary metrics. It is important to note that for some metrics regarding roles and handovers, thresholds have to be defined by organisations in order to determine a normative best possible value by which to make a comparison. While measures can be added up to the organisational level for each dimension, the dimensions can also be merged with regard to the scores HD_H_r and HD_OU. An average of these three values can be calculated to determine an overall measurement (HD) of the organisation’s handover density.

4 Evaluation

4.1 Test-data collection

To evaluate the applicability of our method, we used an existing process model repository, and organisational hierarchy data that originate from a major telecommunications company. The repository is divided into 11 main process areas (Level 1) in its value chain. Processes are modelled for all the process areas in a hierarchical way, yielding at most six sub-process levels. We obtained the test data as a set of xml files and developed a Java application that parses these files to obtain the elements in definitions 1–4, and implement the set operations 1–11. The characteristics of the test data for the overall process model repository and each process area are depicted in Table 1.

	Repository	PA1	PA2	PA3	PA4	PA5	PA6	PA7	PA8	PA9	PA10	PA11
#Processes	1010	108	17	30	202	33	70	19	293	150	26	62
Deepest hierarchy	7	6	6	4	6	5	7	3	6	6	5	6
#Activities	9080	777	112	254	1994	400	727	82	2740	1380	236	378
#Gateways	3118	308	44	89	676	121	236	21	907	535	63	118
#Potential handovers	11197	711	155	274	2724	482	850	73	3486	1818	211	413
#Roles	442	102	23	38	105	53	96	24	143	62	20	45
#OUs	157	59	12	28	57	32	54	18	66	33	14	24

Table 1: Overview of the test-data collection

The main application scenario for the presented method to measure handover density is to provide an organisation with measurement results based on a process model repository as input. Thus, application of the method is not necessarily time critical. However, if the method is applied under changing conditions – for instance, while experts analyse changes in the handovers with respect to alternative process model designs – the computation time must be reasonable. We tested the application of our method on the evaluation process model repository by using a MacBook Pro with a 2.60 GHz Intel Core i5 processor and 8 GB RAM on a Java implementation. Execution times were as follows: Step 1: 125 sec., step 2: 242 sec., step 3: 1 sec., step 4: 1 sec. (total: 369 sec.). Most of the execution time was allocated to Step 2, in which potential handovers were identified according to set operations 1 to 8. Even for a repository with 1,012 process models, it took about *six minutes* to execute the complete method. Given that it is possible to partially execute the method when only some parts of the repository are updated, we consider this a reasonable performance.

4.2 Measurement results

We calculated the two types of metrics on the handover density, as presented below.

4.2.1 Measurements for role handover metrics

Figure 8 shows the number of processes with certain #HrPi values and the number of processes with certain HD_HrP$_i$ values. Many processes in the repository are observed to have no handovers. However, a high number of processes contain more than 10 handovers, although the average value is 4.7 for the repository. There is a high variation among role handover density values of processes, as seen on the right. The organisation may specifically aim to lower the handovers for those processes having density value close to and above 1, which signifies that usually a handover takes place for each different activity in the process.

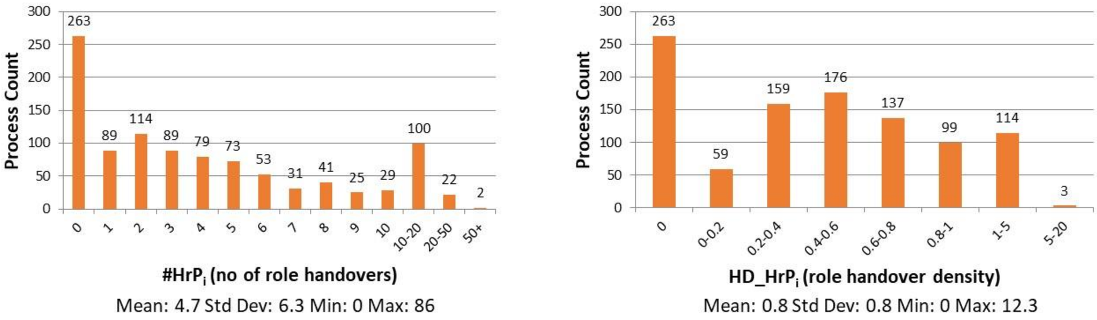


Figure 8: Count of processes with certain #HrP$_i$ values (left), and HD_HrP$_i$ values (right)

Figure 9 depicts the HD_H_R (role handover density) scores for processes in specific process areas and the overall repository. Based on these scores, the organisation may prioritize their efforts to lower handovers for specific process areas.

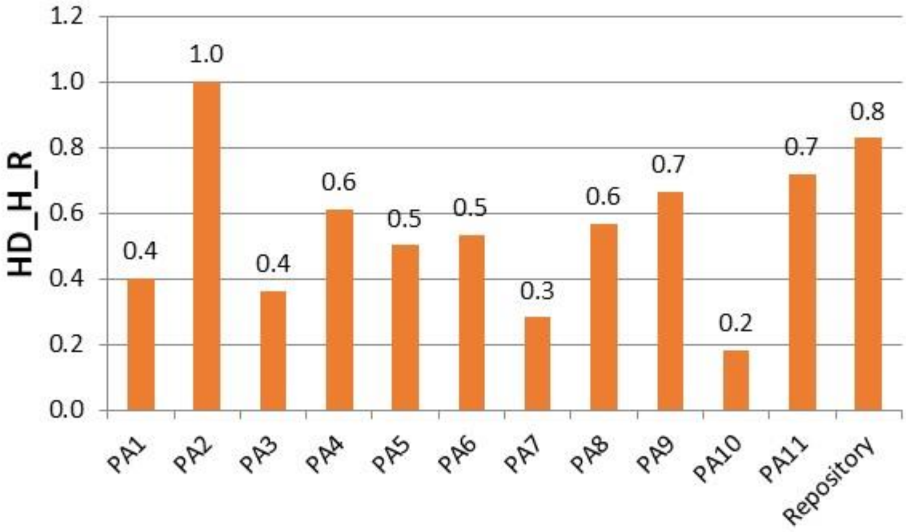
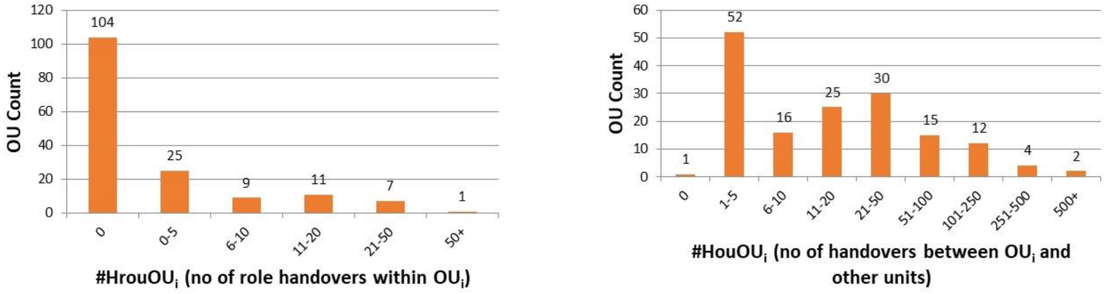


Figure 9: HD_H_R scores for process areas and the repository

4.2.2 Measurements for OU handover metrics

Figure 10 displays the results of the metrics related to the alignment of organisational units. On the left, it can be observed that many units do not have handovers between their own roles. This may point out to problems in an organisation such as an extreme function-oriented structure with units having only one role or generalisation of roles with only one role name in the unit. In our repository, we observed that 72 of 104 units with no handovers within the unit have indeed only one role defined. On the right, we observe varied values of handovers between units, signifying the existence of units working heavily with other units.



Mean: 3.5 Std Dev: 8.8 Min: 0 Max: 76

Mean: 52 Std Dev: 138.4 Min: 0 Max: 1330

Figure 10: Count of units with certain #HrouOU$_i$ values (left) and #HouOU$_i$ values (right)

Figure 11 displays the handover density scores for units, HD_HrOU$_i$, and the overall OU handover density score for the repository, HD_H_OU (0.17). We expect that more process-oriented units would have more handover within the unit with respect to handovers from that unit to other units, thus having a higher density score. The figure indicates a high number of units for the organisation having low density scores close to 0.

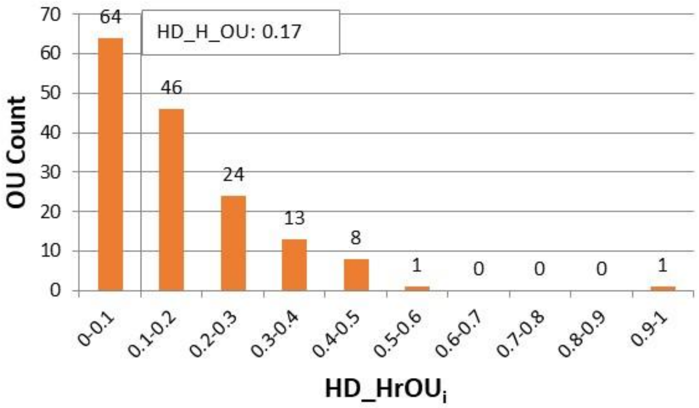


Figure 11: Count of units with certain HD_HrOU$_i$ scores and the overall HD_H_OU score for the repository

5 Discussion and conclusion

The method allows us to determine handovers using process models and organisational charts within an organisation. Such information is typically available within medium-sized to large organisations. While it offers the advantage of being objective compared to gathering subjective data from employees, the accuracy of the measurement is dependent on the quality of the process models. If there is missing information, inaccurate models or incorrect connections, the method might provide inaccurate information regarding handover density as well. This can also be the case when the models do not reflect the reality of process execution, but rather represent paper documentation that is not used in practice.

The evaluation results show that application of the method is not only possible with the described data, but also necessary, as there are major differences between the processes, process areas and organisational units. Hence, the analysis can serve as a starting point to concentrate on process areas with low values first, and to identify whether the low scores are due to the roles, handovers or organisational units. This is also acknowledged by the company providing the process model repository. The results indicate a high number of handovers (e.g., more than 10, as in Figure 8) and high role handover density (e.g., close to and higher than 1.0) for some processes. Differences among process areas in terms of handovers are also highlighted (as seen in Figures 9). Moreover, differences among how organisational units work has been revealed (as seen in Figure 10 and 11). Issues with the definition of some units, such as having only one role within the unit, are revealed. Such insights enable the company to take actionable decisions on the reorganization of processes and organisational units.

5.1 Theoretical implications

First, evaluating handovers in organisations has been highlighted as important for process efficiency and innovation by Leyer *et al.* (2017), but the authors rely on subjective questioning of employees. Our method contributes to the literature by providing an objective and automated way to measure and analyse handovers based on process model repositories.

Second, our results contribute to information systems theory by showing how handovers in organisations can be identified. Such identification of handovers (also termed handoffs) is an important aspect for determining efficiency and innovativeness of production processes (Pentland *et al.*, 2017), but this conceptualisation and measurement has not been emphasized sufficiently in prior literature. Our approach enables identification of handovers, and thus determination of changes in patterns for working together from functions to processes; i.e. different connections between employees regarding work and information, and how information flows change across production settings.

Third, our results contribute to the alignment of organisational designs with the increasing implementation of the Internet of Things (IoT) in organisations. While IoT promotes permanent connection between objects and employees over the internet (Xia *et al.*, 2012), it also emphasises the connection of functions (Caputo *et al.*, 2016). Solutions in this domain, however, have not considered value chains and their integration into the hierarchical structure of organisations (Ferretti and Schiavone, 2016). As such, our method contributes to a better understanding of how IoT solutions can be successfully embedded in organisational structures. The method enables the identification of handovers and can help in considering these with an IoT introduction, as well as determining changes in handovers.

5.2 Practical implications

The method enables companies to determine their handover densities throughout the whole organisation. In case they want to reduce handovers, an initial measurement can be conducted using data that is typically available in an organisation. The parts of the organisation in which the density of handovers is lower or higher can then be determined, thus allowing an organisation to decide how to prioritise resource investment towards process changes. It also enables the determination of which types of actions are necessary – i.e., whether emphasis should be on the dimension of processes or organisational units – and trial of design changes to immediately see their effect.

In addition, organisations can use the method to evaluate the introduction of new technologies related to IoT, as outlined in Section 5.1. They can determine how permanently connected objects can change in order to reduce handovers between employees, as well as between employees and systems/machines. This can foster successful introduction and usage of the new possibilities in this regard.

The extent of the practical implications of our work is dependent on several organisational and process factors. The method can deal with various degrees of complexity within the chosen business processes, but greater effects of its usage can be expected when there are more handovers between functions. Furthermore, the developed method provides results from an analytic perspective, while the practical usage and implementations of recommendations are dependent on the process-oriented culture and managerial maturity in an organisation. Hence, practical implications – i.e., the impact of the results on the practice of processes and organisational structure – have to be viewed against this managerial background. The higher the process-oriented culture and managerial maturity, the more the results will be considered to change the organisation and, thus, have a higher practical impact.

5.3 Limitations and outlook

Our method comes with some limitations that should be addressed in future work. First, researchers should focus on integrating the use of event logs into the method. This can provide an opportunity to deal with the risk of outdatedness of process models while keeping their benefits for complete and accurate handover identification. Second, we do not weight the occurrence of handovers in terms of frequency. Event logs could also be used to determine the frequency of handovers or roles that work together to weigh connections identified in the current method. Third, we do not consider further aspects, such as the geographical closeness of roles or the ability of employees to collaborate, with which the data could be enriched in future research.

References

Aysolmaz, B. and O. Demirörs (2015), "Unified process modeling with UPROM tool", in P.E. Nurcan S. (Ed.) *Information Systems Engineering in Complex Environments. CAiSE Forum 2014*, Springer, Cham, pp. 250-266.

Aysolmaz, B. and H.A. Reijers (2017), *Use Cases for Understanding Business Process Models*, in Proceedings of the Advanced Information Systems Engineering: 29th International Conference, Nr., Cham, p. 428-442.

Bitici, U.S., F. Ackermann, A. Ates, J. Davies, P. Garengo, S. Gibb, J. MacBryde, D. Mackay, C. Maguire, R. van der Meer, F. Shafti, M. Bourne and S.U. Firat (2011), "Managerial processes. Business process that sustain performance", *International Journal of Operations and Production Management*, Vol. 31 No. 8, pp. 851-887.

Bronzo, M., P.T.V. de Resende, M.P.V. de Oliveira, K.P. McCormack, P.R. de Sousa and R.L. Ferreira (2013), "Improving performance aligning business analytics with process orientation", *International Journal of Information Management*, Vol. 33 No. 2, pp. 300-307.

Browning, T.R. (2010), "On the alignment of the purposes and views of process models in project management", *Journal of Operations Management*, Vol. 28 No. 4, pp. 316-332.

Caputo, A., G. Marzi and M.M. Pellegrinie (2016), "The internet of things in manufacturing innovation processes. Development and application of a conceptual framework", *Business Process Management Journal*, Vol. 22 No. 2, pp. 383-402.

Curtis, B., M.I. Kellner and J. Over (1992), "Process Modeling", *Communications of the ACM*, Vol. 35 No. 9, pp. 75-90.

Daft, R.L., J. Murphy and H. Willmott (2007), *Organization theory design*, South-Western Cengage Learning, Andover.

Davies, I., P. Green, M. Rosemann, M. Indulska and S. Gallo (2006), "How do practitioners use conceptual modeling in practice?", *Data & Knowledge Engineering*, Vol. 58 No. 3, pp. 358-380.

Davis, R. and E. Brabander (2007), *ARIS design platform. Getting started with BPM*, Springer, London.

Dijkman, R.M., M. La Rosa and H.A. Reijers (2012), "Editorial: Managing large collections of business process models. Current techniques and challenges", *Computers in Industry*, Vol. 63 No. 2, pp. 91-97.

Dumas, M., W.M.P. van der Aalst and A.H. ter Hofstede (2005), *Process-Aware Information Systems. Bridging People and Software Through Process Technology* Wiley, Hoboken, NJ.

Ferretti, M. and F. Schiavone (2016), "Internet of things and business processes redesign in seaports. The case of Hamburg", *Business Process Management Journal*, Vol. 22 No. 2, pp. 271-284.

Hong, S., Y. Lee, J. Kim and I. Choi (2012), "A methodology for redesigning an organizational structure based on business process models using SNA techniques", *International Journal of Innovative Computing, Information and Control*, Vol. 8 No. 7, pp. 5411-5424.

Iren, D. and H.A. Reijers (2017), *Leveraging business process improvement with natural language processing and organizational semantic knowledge*, in Proceedings of the International Conference on Software and System Process (ICSSP) Nr., Paris, p. in press.

Jones, G.R. (2010), *Organizational theory, design, and change*, Pearson Education, Upper Saddle River, NJ.

Koschmider, A., M. Song and H.A. Reijers (2009), "Social software for business process modeling", in *BPM 2008 Workshops*, Springer, Berlin Heidelberg, pp. 666-677.

Leyer, M. and J. Moormann (2012), "A method for matching customer integration with operational control of service processes", *Management Research Review*, Vol. 35 No. 11, pp. 1046-1069.

Leyer, M., J. Stumpf-Wollersheim and F. Pisani (2017), "The influence of process-oriented organizational design on operational performance and innovation", *International Journal of Production Research*, Vol. 55 No. 18, pp. 5259-5270.

Leyer, M. and J. Wollersheim (2013), "How to learn process-oriented thinking. An experimental investigation of the effectiveness of different learning modes", *Schmalenbachs Business Review*, Vol. 65 No. 4, pp. 454-473.

List, B. and B. Korherr (2006), *An evaluation of conceptual business process modelling languages*, in SAC ’06 Proceedings of the 2006 ACM symposium on applied computing Nr., Dijon, p. 1532-1539.

Pentland, B.T., J. Recker and G. Wyner (2017), "Rediscovering handoffs", *Academy of Management Discoveries*, Vol. 3 No. 3, pp. 284-301.

Reijers, H.A. and J. Mendling (2011), "A Study Into the Factors That Influence the Understandability of Business Process Models", *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 41 No. 3, pp. 449-462.

Rosemann, M. (2006), "Potential pitfalls of process modeling. Part A", *Business Process Management Journal*, Vol. 12 No. 2, pp. 249-254.

Russell, N., W.M.P. van der Aalst, A.H.M. ter Hofstede and D. Edmond (2005), "Workflow resource patterns: Identification, representation and tool support", in *Advanced Information Systems Engineering: 17th International Conference, CAiSE 2005*, Springer, Berlin Heidelberg, pp. 216-232.

Schefer-Wenzl, S. and M. Strembeck (2014), "Modeling support for role-based delegation in process-aware information systems", *Business & Information Systems Engineering*, Vol. 6 No. 4, pp. 215-237.

Segatto, M., S.I.D. de Pádua and D.P. Martinelli (2013), "Business process management. A systemic approach?", *Business Process Management Journal*, Vol. 19 No. 4, pp. 698-714.

Škrinjar, R. and P. Trkman (2013), "Increasing process orientation with business process management. Critical practices", *International Journal of Information Management*, Vol. 33 No. 1, pp. 48-60.

Turetken, O., T. Rompen, I. Vanderfeesten, A. Dikici and J. van Moll (2016), "The effect of modularity representation and presentation medium on the understandability of business process models in BPMN", in *Business Process Management: 14th International Conference, BPM 2016*, Springer, Cham, pp. 289-307.

Weske, M. (2012), *Business Process Management. Concepts, Languages, Architecture*, Springer, Heidelberg.

Womack, J.P. and D.T. Jones (2003), *Lean Thinking. Banish waste and create wealth in your corporation*, Free Press, New York.

Xia, F., L.T. Yang, L. Wang and A. Vinel (2012), "Internet of things", *International Journal of Communication Systems*, Vol. 25 No. 1, pp. 1101-1102.