ICIC Express Letters
Part B: Applications
Volume **10**, Number **7**, July **2019**

ICIC International ©2019 ISSN 2185-2766
pp. **643–650**

VISUALIZING HIERARCHICAL PROPERTIES OF BUSINESS PROCESS ENTITIES

Hyun Ahn[1], Kyoungsook Kim[2], Tai-Woo Chang[3]
and Kwanghoon Pio Kim[1,*]

[1]Division of Computer Science and Engineering
[3]Department of Industrial and Management Engineering
Kyonggi University
154-42, Gwanggyosan-ro, Suwon-si, Gyeonggi 16227, Korea
{ hahn; keenbee }@kgu.ac.kr; *Corresponding author: kwang@kgu.ac.kr

[2]Department of Computer Engineering
Kyung Hee University
1732, Deogyeoung-daero, Yongin-si, Gyeonggi 17104, Korea
khmjmc@khu.ac.kr

Received December 2018; accepted March 2019

Abstract. *Business Process Intelligence (BPI) is an active research field for discovering valuable knowledge about business processes and leveraging it to improve such processes or perform relevant decision-making tasks. Many aspects and properties of business processes have been addressed for the BPI. In this paper, we mainly focus on the hierarchical aspect of business processes. A hierarchical structure formed among business process entities has various analytical properties; therefore, it ought to be analyzed and visualized properly. In particular, we propose a treemap visualization method to gain insights that how business process entities have mutual relationships among them in a hierarchical manner. To this end, we describe an overall procedure including phases from the creation of business process entity hierarchy to the visualization phase.*
Keywords: Business process intelligence, Hierarchical property, Process entity hierarchy, Treemap method

1. **Introduction.** As many organizations today consider their business processes as core assets affecting the enterprise value, Business Process Management (BPM) methodology and Process-Aware Information Systems (PAIS) have been applied in various industrial fields. Accordingly, those organizations have begun to pay attention to the development of intelligent analysis methods for optimizing their processes using vast amounts of process execution data.

Business Process Intelligence (BPI [1]) aims to discover valuable knowledge from process-related data (e.g., process models and event logs) and utilize such knowledge in the decision-making task. [2] defines that BPI serves as the interface between BPM and data science, and thus it is closely related to the several fields for analyzing business processes, such as process mining and Business Activity Monitoring (BAM). Broadly, the essential capabilities required for performing BPI activities fall into three major capabilities: the data acquisition, analysis, and visualization capabilities. In this paper, we mainly focus on the visualization capability of discovered knowledge of business processes from the hierarchical aspect.

A hierarchical structure can be formed among business process entities; we call this structure a process entity hierarchy. This hierarchy has various analytical properties and therefore should be properly analyzed and visualized. In this regard, we propose

DOI: 10.24507/icicelb.10.07.643

a treemap visualization method to gain insights that how business process entities have mutual relationships among them in a hierarchical manner. That is, our treemap method visualizes a process entity hierarchy and the analyzed result. To this end, we exploit a meta-model [3] that includes hierarchical relations between process entity types. Based on this meta-model, we are able to generate a process entity hierarchy that we want to visualize. Finally, as a validation step, we demonstrate a running example of the generation and visualization of process entity hierarchy by using an implemented prototype system.

2. **The Hierarchical Relations among Business Process Entities.** In many information systems, it is common to exploit a hierarchical data model that consists of different levels of data (e.g., a hierarchical data model for the authority management). In general, the purpose of the analysis of such hierarchies is to understand hierarchical characteristics and relationships between data elements and to evaluate each element by its proportion. Likewise, hierarchical structures and analytical properties exist in the environment that operates business processes. For example, a business process model, which is operated by a PAIS, is hierarchically structured in terms of control-flow, such as process packages, process models, and individual activities.

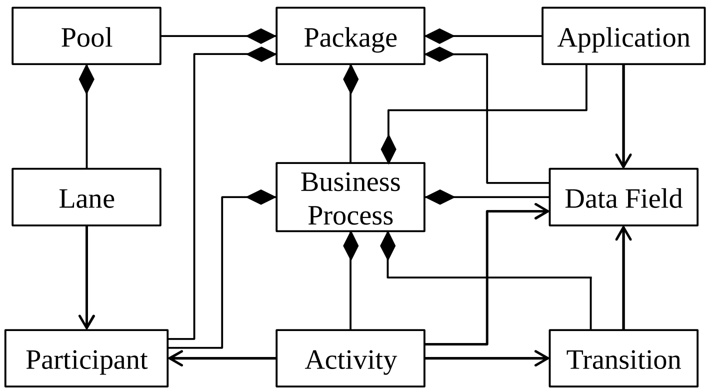


FIGURE 1. Meta-model representing hierarchical relations (compositions and directed associations) between process entity types [3]

To facilitate the analytics of hierarchical structure, we utilize a meta-model that includes essential process entity types and their hierarchical relations between them (shown in Figure 1). This meta-model is based on the XPDL (XML Process Definition Language [4]) and BPMN (Business Process Model and Notation [5]) standard specifications. Even though these standards are the outstanding standards for business process definition, there is still a need to extend the meta-model to be compatible with other related standards in the future. Accordingly, process entity types included in the meta-model are defined as follows.

- A Package refers to a grouping of business processes in a specific business domain. In general, a package is mapped to the root element in many XML-formatted process models (e.g., the XPDL standard).
- A Business Process is the central entity type in a business process model. Therefore, it contains subordinate entity types of main aspects, including entities of Activity, Participant, and Data Field.
- A typical Activity represents a logical unit of work in a business process. Also, both the routing activities (e.g., OR split/join, AND split/join) and events are other types of activity, but since these types do not imply a hierarchical property or related semantics we only take account of typical activities in this paper.

- A Transition connects two ordered activities. Therefore, these entities are core elements of the control-flow aspect of a business process. In cases of transitions connected from a disjunctive routing activity, a transition condition is attached to a corresponding Transition entity. Accordingly, the transition will be activated only when the transition condition holds true during the execution of the business process model.
- A Pool is a top-level container for resources allocated in a business process. In business process modelling systems, it is used to graphically augment the view of the resources to a business process diagram. For the BPMN model, a Pool entity includes a set of Lane entities as its child nodes.
- A Lane is used to describe a responsibility or accountability of a certain set of activities in a business process. Therefore, a Lane is often relevant to a business role (e.g., administrator and contract manager), a system (e.g., a legacy system), and an organizational unit (e.g., department and project team).
- A Participant is a resource that participates in specific business processes and actually performs activities within the business processes. It can be not only a human resource but also a machine resource.
- A Data Field is created and consumed within each business process execution. It is used as variables to exchange intermediate results between activities or evaluate transition conditions.
- An Application refers to a software program or a service invoked from a process engine during business process executions. This entity provides functionalities required to carry out automatic tasks involved in a business process.

Regarding the hierarchical relations between entity types, there are two types of relations: composition and directed association. A composition (depicted as −⧫) indicates an explicit hierarchical relationship between a superior entity type and subordinate entity type. In XPDL models, these relationships are explicitly represented by a hierarchical structure of the corresponding XML elements (e.g., <Package>−⧫ <Workflow Process>−⧫ <Activity>). Likewise, for BPMN models, there are many types of process entity hierarchies explicitly represented. (e.g., <Diagram>−⧫ <Process>−⧫ <Pool>−⧫ <Lane>).

On the other hand, a directed association (depicted as →) indicates an association relationship between a referring entity type (source) and referenced entity type (sink) that is navigable in only one direction and is interpreted as an implicit hierarchical relationship. In general, these relationships are not represented as an explicit hierarchy in a process model, but it implies the hierarchical semantics between process entities. For the case of an XPDL model, an Activity element contains reference information for a number of Participant elements which is responsible in performing that activity (<Activity>→<Participant>).

3. **Process Entity Hierarchy and Its Analytics.** Based on the meta-model, we can generate and analyze process entity hierarchies. Figure 2 shows the overall procedure of the process entity hierarchy analysis. This section describes details of the operations of the overall procedure, especially focusing on the implemented prototype system.

3.1. **Parsing process models.** Firstly, the prototype system takes process models (e.g., XPDL files) as an input set, and then performs the parsing task to transform information of XML elements corresponding process entities into a group of process entity objects that are manipulable by the system. Figure 3 shows the group of process entity objects extracted from the parsing task. These objects are categorized into three object classes (control-flow, data and resource) and serve as sources for the process entity hierarchy

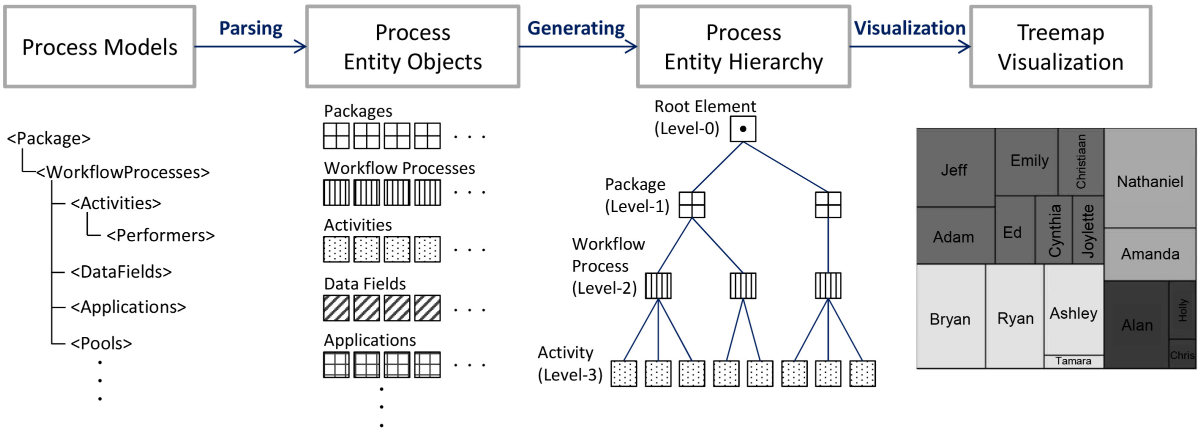


FIGURE 2. The overall procedure for the analytics of process entity hierarchy

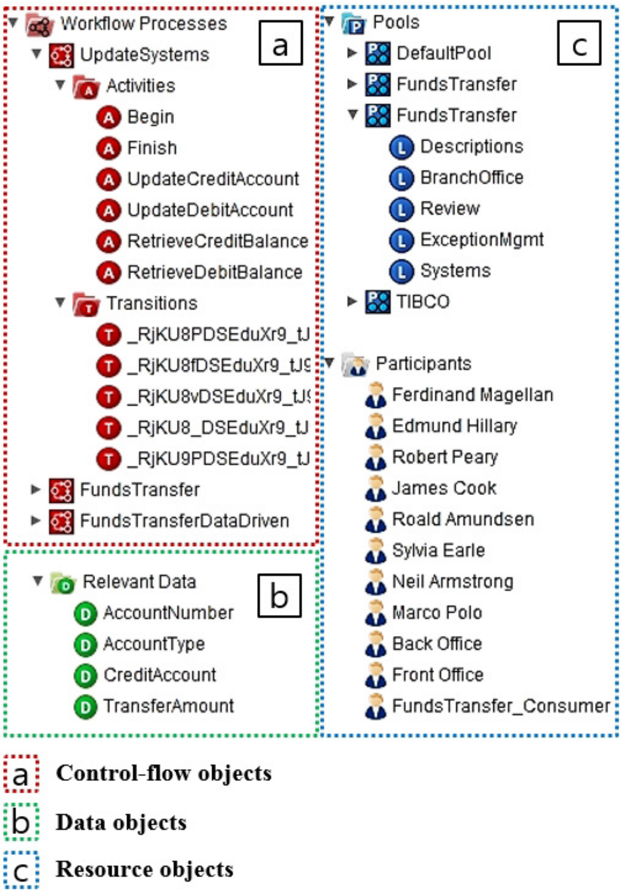


FIGURE 3. A screen capture of the tree of the extracted process entity objects

analysis. Further, the analysis capability can be enriched if we add event logs or external data sources (e.g., organizational models, measured values) in addition to these objects.

3.2. **Generating a process entity hierarchy.** Next task is the generation of process entity hierarchy. Figure 4 represents an entity tree representation of the hierarchical relations of process entity types defined in the meta-model. A graphical user interface provided by the system guides users to easily generate a process entity hierarchy based on the entity tree.

More specifically, the hierarchy generation is implemented by traversing the entity tree for choosing the entity types to be included in the hierarchy from the first entity type to

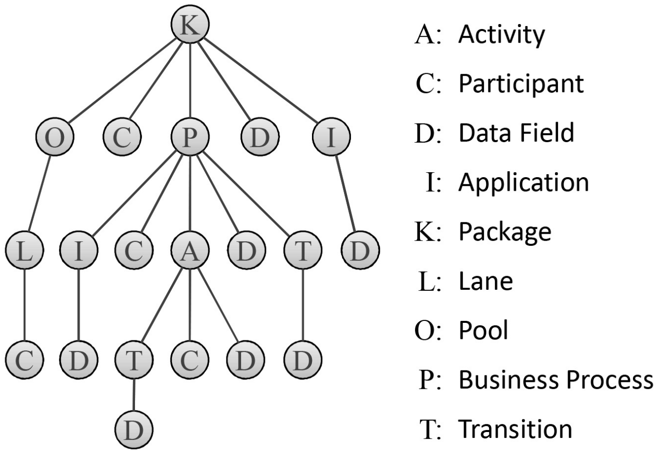


FIGURE 4. Tree representation of the hierarchical relations of the meta-model

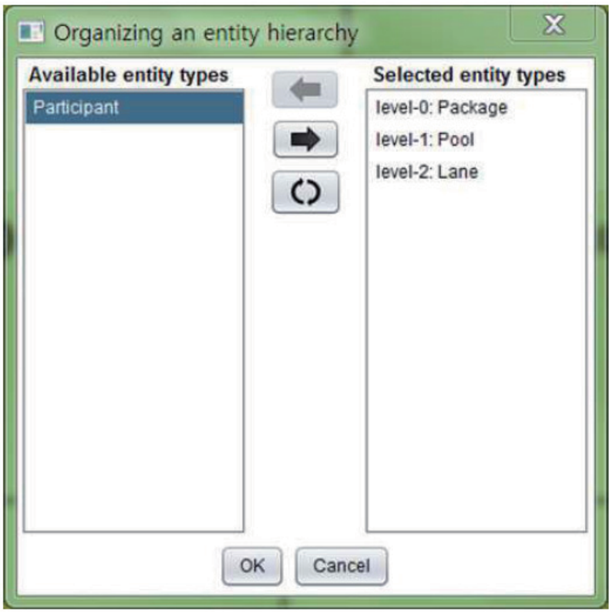


FIGURE 5. Generating an entity hierarchy using the graphical user interface

the last entity type in order as represented in Figure 5. During the hierarchy generation, available entity types will be varied according to the last selected entity type. Additionally, all the entity types, excluding the leaf entity types on the tree, can be the root entity type since the minimum required level for hierarchy is two.

3.3. **Analyzing the process entity hierarchy.** After the generation of the process entity hierarchy, the system allows users to perform the analytics task on the generated hierarchy by means of analysis techniques that fit the hierarchical data.

4. Treemap Visualization Method.

4.1. **The configuration of the treemap visualization.** After the generation of the process entity hierarchy, the system allows users to perform the analytics task on the generated hierarchy by means of analysis techniques that fit the hierarchical data. Treemap [6] is the most common visualization method for hierarchical data. It assigns a size to a node according to its value. Also, it displays all nodes of the hierarchy including, leaf nodes, branch nodes, and root node as squares and eventually puts all the squares into one large square. Therefore, it provides convenience to intuitively understand the hierarchical structure of the whole data and, portions of each node at the same time.

To visualize the analyzed entity hierarchy, the configuration step is needed for deciding which hierarchy will be visualized and which metric will be applied to the treemap visualization. Figure 6 shows two configuration steps for the different entity hierarchy visualizations. In case of the left side, the selected entity hierarchy is Hierarchy 1 and the selected metric is *workrate* for the Participant entities corresponding to the type of leaf node in Hierarchy 1. On the other hand, the right side in Figure 6 shows that another hierarchy (Hierarchy 2) and metric (*resolution time*) are selected for the visualization.

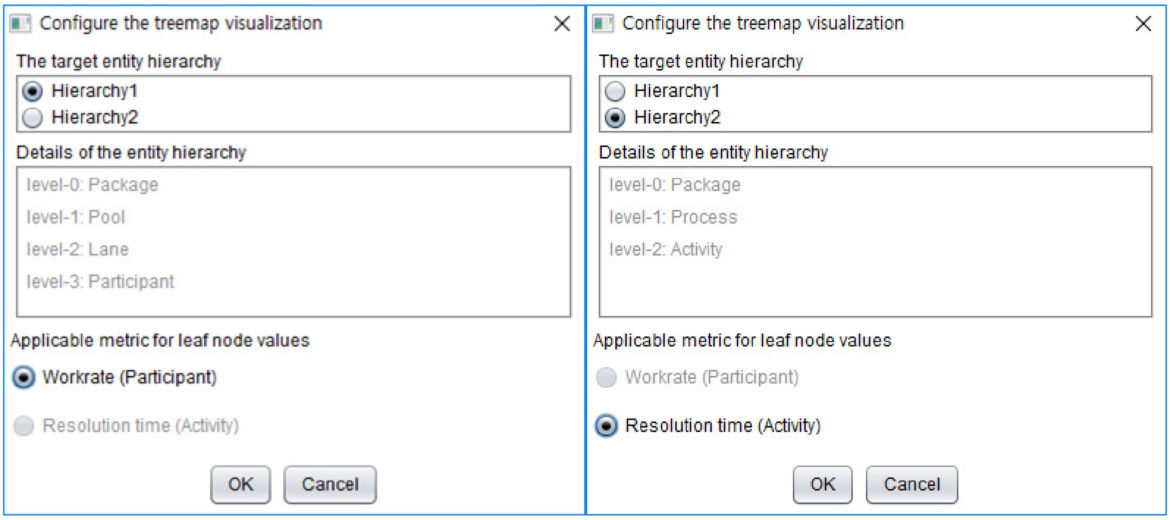


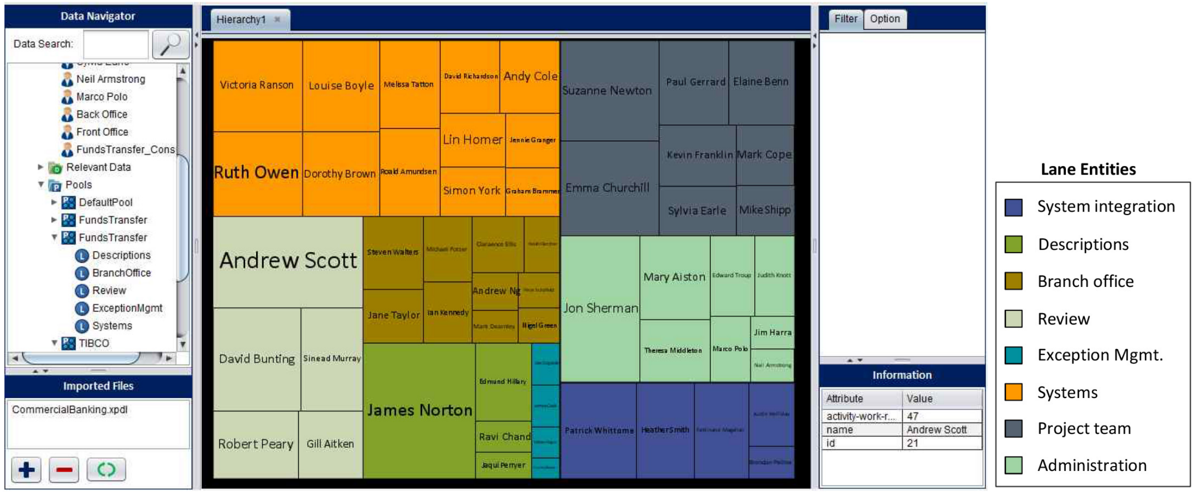
FIGURE 6. Two examples of the configuration step for the treemap visualization

4.2. **Running example.** After the generation of the process entity hierarchy, the system allows users to perform the analytics task on the generated hierarchy by means of analysis techniques that fit the hierarchical data.

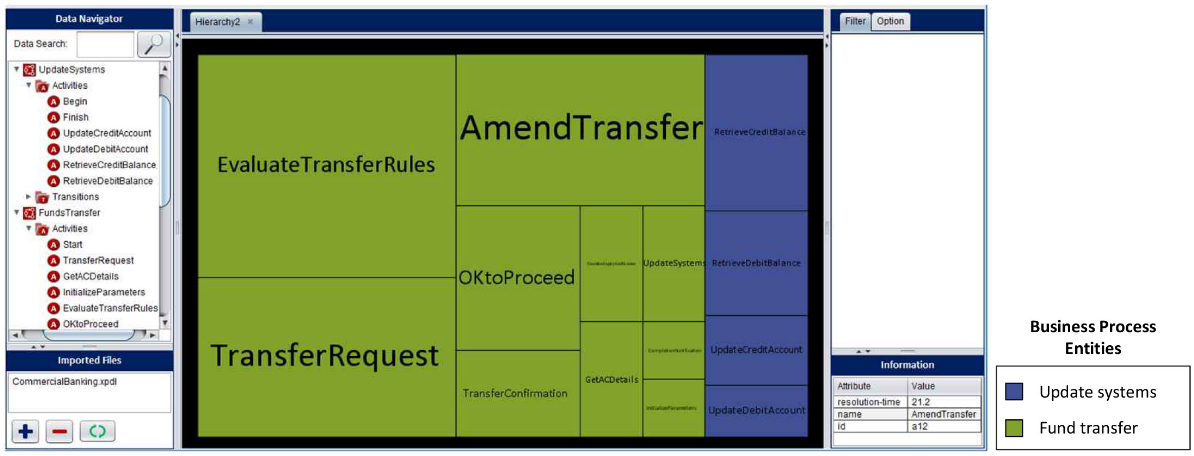
For the validation of the proposed method, we demonstrate visualization results for an example using sample XPDL process models. Figure 7 shows two different treemap visualizations. In case of the above side, the treemap contains 3 Pool, 8 Lane, and 58 Participant entities. The color assigned to each node indicates the Lane entity which the corresponding Participant entity is involved in and the *workrate* of each Participant decides the size of the corresponding node. On the other hand, the treemap visualization on the bottom side contains 1 Package, 3 Business Process, and 14 Activity entities. Each node is coloured according to the Business Process entity which the corresponding Activity entity is included in and is scaled by their *resolution time* measurements.

In the first result (shown in Figure 7(a)), the participant, James Norton occupies the largest space among the entire participants in the treemap visualization that means this participant has completed the most work items compared to the others. In the Lane entity level, the result shows that the lane, Systems contributed to the completion of the most work items than other Lane entities. For the second result (Figure 7(b)), the executions of the Fund transfer process have taken the most time than the Update systems process. In particular, the activity, EvaluateTransferRules shows the highest measured value of *resolution time*.

5. **Related Work.** In this paper, we concentrate on the visualization of hierarchical properties between business process entities. In this regard, this section briefly introduces the previous works divided into the business process visualization and visual analytics of business processes.



(a)



(b)

FIGURE 7. (color online) Treemap visualizations of two different entity hierarchies

In [7], the authors presented a graph layout algorithm to effectively visualize a business process model which consists of the large set of task nodes and their connections of high-density. Similarly, to interactively adjust the visualization, Reichert et al. [8] proposed a method to reflect user-defined parameters to the business process visualization. These prior works mainly focused on the effective solution to enhance perception in understanding large-scale business process visualizations.

Beyond the business process visualization, there are few studies to support activities for the visual analytics of business processes [9-12]. Visual analytics aims for not just visualizing, but discovering valuable knowledge by exploring the visualized results. For example, the visualization technique for replaying event logs [9] to strengthen the capability of users to analyze the dynamic behaviors of business process executions. In addition, other studies, such as the multi-perspective visual comparison method [10], visual monitoring for the business process compliance [11], and the process-oriented social network visualization technique [12], have contributed to the visual analytics of business processes.

While those prior works mentioned above, the hierarchical property of business process has not been significantly considered in the literature. Therefore, we believe that our contribution is distinguished from other works in that it raises the feasibility of the analysis of process entity hierarchies and presents a configurable treemap visualization method for it.

6. **Conclusion.** A hierarchical structure among business process entities has analytical properties and it ought to be analyzed and visualized by proper techniques. In this paper, we propose a treemap method for visualizing the process entity hierarchy and analyzed results. The contributions of the paper are summarized below.

- The overall procedure for the analytics of process entity hierarchy is proposed.
- Developing the visualization method which enables us to interactively configure options to set the process entity hierarchy we want to visualize and select the metric.

In conclusion, the proposed treemap method can be configured to reflect the generated hierarchy and the selected metrics, so that it is expected that knowledge related to process entity hierarchies can be effectively conveyed to stakeholders.

As future work, we plan to revise the meta-model to be compatible with other standards (e.g., WS-BPEL [13], XES [14]). As another issue, we are interested in the development of new visualization method for the process entity hierarchy. The basic treemap method, which we implemented in the system, works well with the process entity hierarchies generated simple process models. However, for the hierarchy containing large-scale nodes, our method is not suitable to visualize in that case. Therefore, we are going to study a new visualization method which is more effective in visualizing process entity hierarchies and those analyzed results in the near future.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (Grant No. 2017R1A2B2010697).

REFERENCES

[1] D. Grigori et al., Business process intelligence, *Computers in Industry*, vol.53, no.3, pp.321-343, 2004.

[2] W. M. P. van der Aalst, J. L. Zhao and H. J. Wang, Editorial: Business process intelligence: Connecting data and processes, *ACM Transactions on Management Information Systems*, vol.5, no.4, article no.18e, 2015.

[3] H. Ahn, K. Kim and K. P. Kim, Generating process entity hierarchies from XPDL process models, *Proc. of the 20th International Conference on Enterprise Information Systems*, vol.2, pp.690-695, 2018.

[4] WfMC, *Process Definition Interface: XML Process Definition Language Version 2.2*, WFMC-TC-1025, 2012.

[5] OMG, *Business Process Model and Notation Version 2.0*, formal/11-01-03, 2011.

[6] M. Bruls, K. Huizing and J. J. van Wijk, Squarified treemaps, *Data Visualization 2000*, pp.33-42, 2000.

[7] B. Albrecht et al., An automatic layout algorithm for BPEL processes, *Proc. of the 5th International Symposium on Software Visualization*, pp.173-182, 2010.

[8] M. Reichert, R. Bobrik and T. Bauer, Enabling personalized visualization of large business process through parameterizable views, *Proc. of the 27th ACM Symposium on Applied Computing*, pp.1653-1660, 2012.

[9] W. M. P. van der Aalst, M. de Leoni and A. H. M. ter Hofstede, Process mining and visual analytics: Breathing life into business process models, *BPM Center Report*, BPM-11-16, BPMcenter.org, 2011.

[10] A. Pini, R. Brown and M. T. Wynn, Process visualization techniques for multi-perspective process comparisons, *Proc. of the Asia-Pacific Conference on Business Process Management*, pp.183-197, 2015.

[11] D. Knuplesch, M. Reichert and A. Kumar, A framework for visually monitoring business process compliance, *Information Systems*, vol.64, pp.381-409, 2017.

[12] M. Park, H. Ahn and K. P. Kim, Workflow-supported social networks: Discovery, analyses, and system, *Journal of Network and Computer Applications*, vol.75, pp.355-373, 2016.

[13] OASIS, *Web Services Business Process Execution Language Version 2.0*, wsbpel-v2.0-OS, 2007.

[14] IEEE Computational Intelligence Society, *Draft Standard for XES: Extensible Event Stream: For Achieving Interoperability in Event Logs and Event Streams*, P1849/D03, 2016.