ResRec: A Multi-criteria Tool for Resource Recommendation

Michael Arias, Eric Rojas, Jonathan Lee, Jorge Munoz-Gama, and Marcos Sepúlveda

Computer Science Department, School of Engineering
Pontificia Universidad Católica de Chile, Santiago, Chile
{m.arias,eric.rojas,wllee}@uc.cl, {jmun,marcos}@ing.puc.cl

Abstract. Dynamic resource allocation is considered a key aspect within business process management. Selecting the most suitable resources is a challenge for those in charge of making the allocation, because the efficiency with which this task is executed, can contribute to the quality of the results, and improve the process performance. Different mechanisms have been proposed to improve resource allocation. However, there is a need for more flexible allocation methods that integrate a set of conditions and requirements defined at run-time, and also, allow the combination of different criteria to evaluate resources. In this paper, we present ResRec, a novel Multi-factor Criteria tool that can be used to recommend and allocate resources dynamically. The tool provides the feature of solving individual requests (On-demand), or requests made in blocks (Batch) through a recommender system developed in ProM.

Keywords: resource allocation, process mining, business processes, recommendation systems, organizational perspective

1 Significance of Resource Recommendation for BPM

The increasing interest in the study of resources and their relationship within the organizational mining perspective [1] has helped to generate insight on how to improve the process efficiency. One of the main categories within this perspective is the resource allocation. The existing allocation approaches aim to provide mechanisms to coordinate in a better way the resources involved in their business processes. Allocate resources dynamically is a relevant challenge [7]. It implies the selection of the most appropriate resource to execute a request created at run-time, considering a set of specific process characteristics that influence the decision of what resource allocate to execute the corresponding activities. Few methods pay attention to the information contained in event logs and organizational models to allocate resources dynamically. For instance, Reinforcement learning [5] has been used to adjust and optimize the resource allocation based on learning appropriate policies incorporating a feedback analysis at run-time; whereas, a Naïve Bayes Model approach has been proposed to select the best performer to execute an activity based on the total completion time [6]. There

is a need to optimize the task of allocating resources through methods that: i) support resource allocation at run-time; ii) use different criteria to evaluate resources, combining process information with other additional information (e.g., resource characteristics); iii) execute the allocation considering different abstraction levels (e.g., activity, sub-process, or process level); and iv) support different decision strategies faced by the person in charge of making the allocation in their daily processes. In this paper, we present ResRec, a novel multi-criteria tool to allocate/recommend resources dynamically. The tool combines different criteria to assess resources, allows the definition at run-time of individual requests (On-demand) or requests made in blocks (Batch), and provides distinct decision alternatives –a decision maker-oriented approach–. The remainder of the article is structured as follows. Section 2 introduces concepts related with the ResRec framework for resource allocation/recommendation. In Section 3 we present the details about the ResRec implementation. Section 4 presents a demonstration of the tool and its functionality. Finally, Section 5 presents the work conclusions.

2 The Resource Recommendation Framework

The proposed framework for allocating/recommending the most suitable resources for executing activities in a business process is shown in Figure 1. For allocation we mean selecting the most appropriate resource. For recommendation we mean retrieving a ranking of the most suitable resources that could be considered by the person in charge of the allocation. The user responsible for undertaking the allocation specifies the characterization of a resource allocation/recommendation. This characterization is integrated by different factors that represent the desired request properties, e.g., what part of the process is the resource request for or what specific characteristic the process instance has. For example, in a Help-Desk process, $Contact\ level\ 1 - English$ are two factors that can characterize a request. Resources are evaluated considering several criteria: frequency, performance, quality, cost, expertise and workload. Furthermore, specific weights are defined according to the importance level that the responsible wants to give to each criterion.

We use an information repository to store the resource contextual data (e.g., expertise), and the historical information about past process execution (e.g., frequency, performance, quality, and cost). We created a knowledge base, where we use a Resource Process Cube to abstract at a conceptual level (not at an implementation level) the historical information, and expertise matrices to represent the expertise of a resource and the desired level of expertise required to execute a given characterization. Each available resource to execute a given characterization is evaluated considering different criteria, which are computed across several metrics defined in [4]. The definition of the Resource Process Cube is closer to the well-known OLAP cubes, providing slice and dice operations for the analysis of each specific characterization and resource. We propose four use cases: 1) On-demand Resource Allocation: a resource is allocated to a single request; 2) On-demand Resource Recommendation: a ranked list of resources is recommended for a single request; 3) Batch Resource Allocation: a batch of requests is evaluated, and a resource is allocated to each of them; and

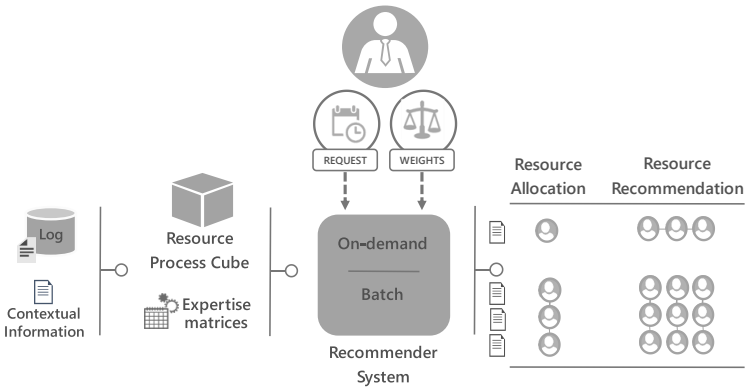

Fig. 1. General overview of the framework to allocate/recommend resources

4) Batch Resource Recommendation: a batch of requests is considered, and a ranked list of resource tuples is recommended to perform all requests. We use a recommender system that uses two methods to allocate/recommend the most suitable resources: a method based on Integer Linear Programming (to support On-demand or Batch Resource Allocation), and a method based on the Best Position Algoritm (BPA2) [2] (to support On-demand or Batch Resource Recommendation). The details about the proposed methods and the implementation will be published shortly [3].

3 ResRec: The Tool

To perform a dynamic resource allocation/recommendation, the ResRec tool provides two plugins implemented in ProM: 1) *GenerateResourceKnowledge*, and 2) *Recommend Resources*. Both plugins are available in the *Resource Recommendation* package in the open-source framework ProM (available in ProM nightly-builds [1]).

3.1 Generate Resource Knowledge

To use the ResRec tool, we have first to import the information (contextual and historical) needed for the decision making within the framework (Figure 2a). For this purpose, we created a standardized format to store the information that is considered. ResRec includes an extension to the Java OpenXES library [2] that allows to standardize and manipulate the information that is used to generate the required knowledge. Using the *GenerateResourceKnowledge* plugin (Figure 2b), the different criteria are evaluated through different metrics, generating the resource knowledge.

[1] www.promtools.org/prom6/nightly

[2] https://svn.win.tue.nl/trac/prom/browser/Packages/ResourceRecommendation/Trunk/src/org/processmining/resourcerecommendation/utils/resrecxes

3.2 Recommend Resources

Based on the resource knowledge, we apply the *RecommendResources* plugin (Figure 2c). This plugin allows the configuration of the required parameters to obtain the final allocation/recommendation. Figure 2d shows the needed configuration. First, the person in charge of making the allocation must decide which method wants to perform: resource allocation (single resource), or resource recommendation (a ranking of resources). Then, the person in charge specifies the amount of requests for each type of available characterizations (On-demand for 1 request, Batch for 2 or more requests). After that, weights are defined to describe the importance of each criterion considered in the framework. Finally, the recommender system computes the allocation/recommendation (Figure 2e) according with the chosen configuration, showing as an output the resource(s) allocated/recommended for each request.
Note that the ResRec tool is designed to be generic and extensible, being able to incorporate new criteria to evaluate the resources, so as to make the resource allocation a more effective task. Moreover, the framework can be adapted to be used in any domain or scenario.

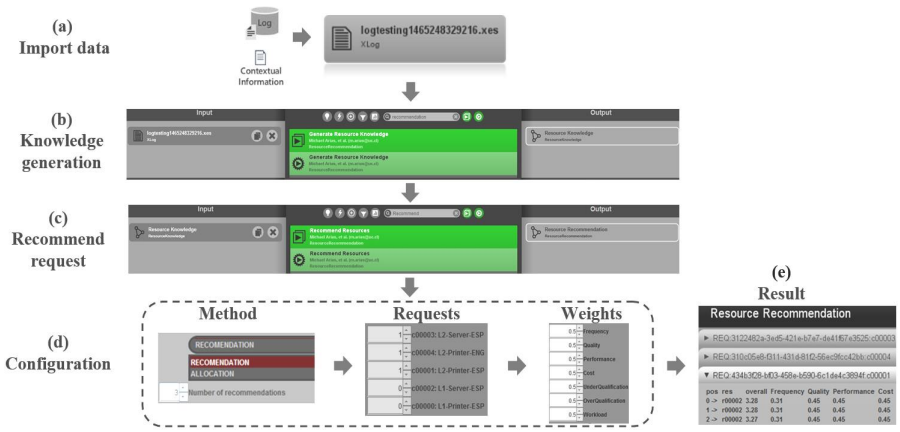

Fig. 2. General walkthrough of the ResRec tool in ProM

4 Demostration

A screencast that demonstrates the usage of the ResRec tool is available on the web (http://is.ing.uc.cl/dcc/index.php/resrec/). The screencast shows the steps followed to generate the resource allocation/recommendation considering the use cases introduced in Section 2. We used a Help-Desk process as a running example. First, we add the contextual and the historical information required to evaluate the resources. Then, we create the knowledge base needed to perform the resource recommendation. After that, we configure the corresponding parameters in order to generate a Batch Resource Recommendation, defining a batch of requests and the weights describing the importance of each criterion. We show the obtained

results by applying the method based on BPA2, performing a Batch Resource Recommendation. Additional to the Batch Resource Recommendation, the other three use cases outlined are also presented. Furthermore, we used the ResRec tool to recommend resources in a real life scenario. The case study included the resource allocation/recommendation for a consulting firm specializing in the sale of business software solutions in Costa Rica, based on its help-desk process.

5 Conclusions

We presented ResRec, a multi-criteria tool that consider historical and contextual information to allocate/recommend the most suitable resources to execute either a single request or a batch of requests, defined at run-time. To obtain the final recommendation, we propose a recommender system that use two methods: 1) Resource allocation based on Integer Linear Programming, and 2) Resource recommendation based on the Best Position Algorithm (BPA2). This tool also has been tested through a real-life help desk scenario of a software consulting company that handles a management solution developed for the automotive industry named DMS-One SAP system. We considered a event log with 1.778 cases registered between August and November 2015. As a future work, we may consider integrating a greedy-based approach as an alternative method to produce a Batch Resource Recommendation with several requests that need to be resolved simultaneously. We also plan to evaluate the incorporation of new criteria to enrich the knowledge base available to determine the most suitable resources. Finally, we aim to apply the framework to allocate/recommend resources with more case studies.

Acknowledgments. This project was partially supported by the Ph.D. Scholarship Program of CONICYT Chile (CONICYT-Doctorado Nacional 2014 - 63140181), Universidad de Costa Rica Professor Fellowships, and by Fondecyt (Chile) Project No.1150365.

References

1. van der Aalst, W.M.P.: Process Mining - Data Science in Action, Second Edition. Springer (2016)
2. Akbarinia, R., Pacitti, E., Valduriez, P.: Best position algorithms for efficient top-k query processing. Inf. Syst. 36(6), 973–989 (2011)
3. Arias, M., Munoz-Gama, J., Sepúlveda, M., Carmona, J., Miranda, J.C.: On-demand and batch resource allocation/recommendation based on multi-factor criteria. (under revision) (2016)
4. Arias, M., Rojas, E., Munoz-Gama, J., Sepúlveda, M.: A framework for recommending resource allocation based on process mining. In: BPM Workshops (2015)
5. Huang, Z., van der Aalst, W.M.P., Lu, X., Duan, H.: Reinforcement learning based resource allocation in business process management. DKE 70(1), 127–145 (2011)
6. Wibisono, A., Nisafani, A.S., Bae, H., Park, Y.: On-the-fly performance-aware human resource allocation in the business process management systems environment using naïve bayes. In: AP-BPM 2015. pp. 70–80 (2015)
7. Zhao, W., Zhao, X.: Process mining from the organizational perspective. In: Foundations of Intelligent Systems, pp. 701–708 (2014)