Batch Processing: Definition and Event Log Identification

Niels Martin, Marijke Swennen, Benoît Depaire, Mieke Jans,
An Caris, Koen Vanhoof

Hasselt University, Agoralaan Building D, 3590 Diepenbeek, Belgium
{niels.martin, marijke.swennen, benoit.depaire, mieke.jans,
an.caris, koen.vanhoof}@uhasselt.be

Abstract. A resource typically executes a particular activity on a series of cases. When a resource performs an activity on several cases simultaneously, (quasi-) sequentially or concurrently, this is referred to as batch processing. Given its influence on process performance, batch processing needs to be taken into account when modeling business processes for performance evaluation purposes. This paper suggests event logs as an information source to gain insight in batching behavior. It marks a first step towards a more thorough support for the retrieval of batch processing knowledge from an event log by (i) identifying different types of batch processing and (ii) briefly outlining a method to generate event log insights.

Keywords: Batch processing • Process mining • Event log knowledge

1 Introduction

A business process is typically described as a series of interconnected activities executed by resources. Resources, e.g. machines or staff members, are assigned to activities and typically carry them out on a series of cases such as products or files. Stating that arriving cases will be handled immediately presents a simplified view on resource behavior. Machines can deal with multiple products together or an employee might deem it more efficient to accumulate invoices and treat the entire stack later. This is referred to as batch processing [7].

Batch processing influences process performance as it can e.g. result in longer waiting times for particular cases [8]. Consequently, it should be considered when modeling a business process for performance evaluation purposes. This is also shown by van der Aalst et al. [7] by illustrating the effect of batch processing on flow times.

While the presence of batch processing might be observable for machines, it is less straightforward to observe the organization of work by humans. Moreover, given the mere fact of being observed, the Hawthorne effect can cause a discrepancy between observed and actual behavior [4]. In this respect, the increasing presence of process-aware information systems, such as ERP systems, can be useful. Such systems record process execution information in event logs. Retrieving insights on batch processing from an event log is indicated as a research challenge in Martin et al. [3].

In process mining research, only Nakatumba [5] proposes a method to identify batch processing. In this method, all resource actions, i.e. executions of activities, are placed on a timeline and grouped in so called chunks. A new chunk is started when the elapsed time between the end of an action and the start of the following action exceeds one hour. When a period such as a working day is composed of multiple chunks, Nakatumba [5] states that batch processing occurs. This work can be leveraged in several ways. Firstly, the one-hour delay required to start a new chunk implies that no actions can be recorded for that resource during this arbitrary timespan. Secondly, the proposed method abstracts from the specific activities performed by a resource, assuming that a resource can batch all activities. Consequently, batch processing is equated to the alternation of periods of activity and periods of total inactivity. Moreover, some activities might be more eligible for batch processing than others. Thirdly, no verification is included to check if the period of inactivity is caused by batch formation or by the absence of cases to process.

This paper marks a first step towards a more thorough support for the retrieval of batch processing knowledge from an event log. Next to the identification and definition of several batch processing types in process mining terminology, a preliminary method to extract insights on this matter from an event log is presented.

2 Batch Processing: Definition and Types

In a production setting, often studied in operations management and operations research literature, batch processing is defined as processing a number of cases simultaneously [6]. The same can hold for service processes, e.g. an information session that is organized for a group of customers instead of individually [2]. However, Baptiste [1] states that the duration of processing a batch can also be determined by the sum of the processing times of all separate cases. This hints at a form of batch processing in which cases are processed sequentially.

From the previous, it follows that different types of batch processing can be distinguished. In general, this paper defines batch processing as the simultaneous, (quasi-) sequential or concurrent execution of an activity on distinct cases by the same resource. Consequently, three types of batch processing are considered, which are illustrated using the example in Figure 1 where an activity is always executed by the same resource for both cases:

- **Simultaneous batch processing.** Instances of an activity are in a simultaneous batch when they are executed by the same resource for distinct cases at the exact same time. For instance: several car parts that need to be painted in the same color can be put together in a spray booth. In Figure 1, the two instances of B are a simultaneous batch as both start and completion times correspond across instances.
- **Sequential batch processing.** Instances of an activity are in a sequential batch when they are executed by the same resource for distinct cases immediately or almost immediately after each other. For example: employees only handle their e-mails twice a day when they are treated sequentially, where a delay of a few seconds might be present between processing two different e-mails. The two instances

of A in Figure 1 form a sequential batch as the start timestamp for the second case corresponds to the complete timestamp of the first case.

- **Concurrent batch processing.** Instances of an activity are in a concurrent batch when they are executed by the same resource for distinct cases partially overlapping in time. For example: a clerk can already start booking a second invoice when additional information is required to finalize the first one. In Figure 1, the instances of activities C, D, E, F and G present types of concurrent batches.

The above batch processing types are largely supported by Wu [8], where simultaneous and sequential batch processing are consistent with the concepts of parallel and serial process batches, respectively.

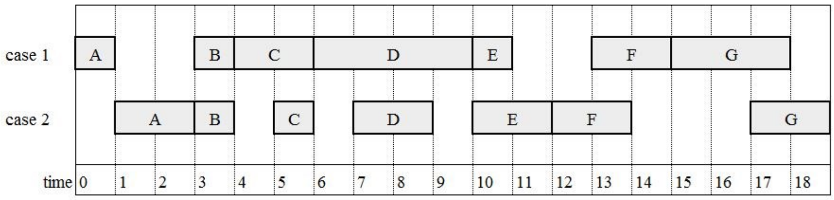

Fig. 1. Schematic representation of two cases, in which an activity is always executed by the same resource for both cases.

3 Batch Processing Identification in Event Logs

To identify batch processing in an event log, the approach visualized in Figure 2 is suggested. This method takes an event log, containing both start and complete events on the one hand and resource information on the other hand, as a starting point. This event log is restructured in a resource-activity matrix (RAM) in which each cell contains a list of events associated to a particular resource-activity combination. The event log is reorganized in a RAM because batch processing stems from the way in which a resource organizes activity execution across cases.

From the RAM, a batching matrix is retrieved for each of the batch processing types defined in Section 2. Each of these matrices has a similar structure as it represents the batching behavior for the batch processing type under consideration. A cell in a batching matrix contains a list of case sets, where each case set is composed of cases handled in batch. To determine whether cases are batched, the definitions outlined in Section 2 are used. Note that case sets containing a single case are also recorded, indicating that this particular case is not part of a batch. Including the latter can be helpful to place case sets containing multiple cases into perspective.

Using the information in the batching matrix as a starting point, future work will define and extract useful batch processing metrics such as the occurrence frequency of batch formation and the average batch size on different levels of analysis.

The retrieval of batching matrices has been implemented and applied to an artificial event log in which batch processing is explicitly included. Solely using this event log, preliminary experimentation shows that the suggested approach is able to discover batch processing.

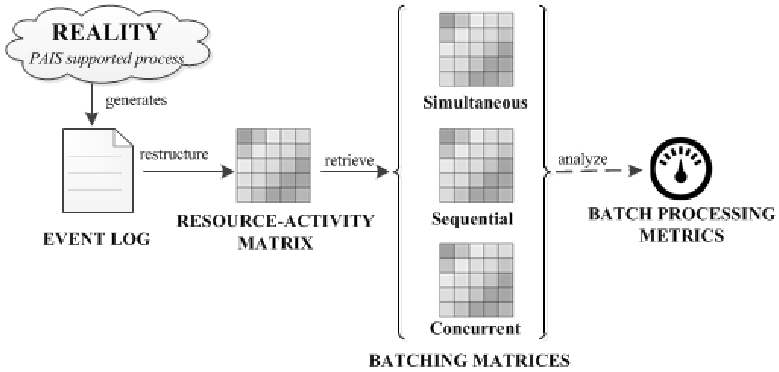

Fig. 2. Overview of batch processing identification method.

4 Conclusion and Future Work

Given its implications on process performance, batch processing needs to be considered when modeling business processes for performance evaluation purposes. To gain insight in batching behavior, this work is a first step towards a more thorough support for the retrieval of batch processing knowledge from an event log by (i) identifying and defining several types of batch processing and (ii) briefly outlining a method to generate insights from event logs.

As indicated in Section 3, future work will leverage these efforts by defining and calculating valuable batch processing metrics. Moreover, a distinction between sequential batch processing and regular queue handling will be introduced as both leave the same trail in an event log. Other possible extensions include discovering batches of activity sequences instead of single activities and distinguishing batching behavior using case attributes as particular cases might be batched while others are not.

5 References

1. Baptiste, P: Batching identical jobs. Mathematical methods of operations research, 52(3), 355-367 (2000)
2. George, M. L.: Lean six sigma for service. McGraw-Hill, New York (2003)
3. Martin, N., Depaire, B., Caris, A.: The use of process mining in business process simulation model construction: structuring the field. Business & Information Systems Engineering (forthcoming)
4. McBride, D.M.: The process of research in psychology. Sage, Thousand Oaks (2016)
5. Nakatumba, J.: Resource-Aware Business Process Management Analysis and Support. PhD thesis. Eindhoven University of Technology (2013)
6. Pinedo, M. L.: Scheduling: theory, algorithms, and systems. Springer, Heidelberg (2012)
7. van der Aalst, W.M.P., Nakatumba, J., Rozinat, A., Russel, N.: Business process simulation. In: vom Brocke, J., Rosemann, M. (eds) Handbook of business process management. Springer, Heidelberg (2010)
8. Wu, K.: Taxonomy of Batch Queueing Models in Manufacturing Systems. European Journal of Operational Research, 237, 129-135 (2014)