A Process Warehouse for Process Variants Analysis

Lisana Berberi[1[0000−0002−7632−2466]]

Karlsruhe Institute of Technology, Hermann-von-Helmholtz-Platz 1, 76344 Karlsruhe, Germany
lisana.berberi@kit.edu
https://www.scc.kit.edu

Abstract. Process model variants are collections of similar process models evolved over time because of the adjustments that were made to a particular process in a given domain, e. g.,order-to-cash or procure-to-pay process in reseller or procurement domain. These adjustments produce some variations between these process models that mainly should be identical but may differ slightly. Existing approaches related to data warehouse solutions suffer from adequately abstracting and consolidating all variants into one generic process model, to provide the possibility to distinguish and compare among different parts of different variants. This shortcoming affects decision making of business analysts for a specific process context. This paper addresses the above shortcoming by proposing a framework to analyse process variants.
The framework consists of two original contributions: (i) a novel meta-model of processes as a generic data model to capture and consolidate process variants into a reference process model; (ii) a process warehouse model to perform typical online analytical processing operations on different variation parts thus providing support to decision-making through KPIs; The framework concepts were defined and validated using a real-life case study.

Keywords: Process variant · process warehouse · business process analysis.

1 Introduction

Process model variants, as collections of similar process models, may evolve over time because of the adjustments made to the same business process in a given domain, e. g.,order-to-cash or procure-to-pay process in reseller or procurement domain. These adjustments produce some variations between these process models. Surely, between business processes across department of the same organization, or across companies in a given industry many common activities are frequently found. For example, typical process *procure-to-pay* often consists of a business process that starts from the moment a procure invoice is received from a vendor after a customer places an order and fulfilled if the vendor has received the corresponding payment. All these *procure-to-pay* processes include activities

related to receiving, invoicing and payment. They may look the same but they slightly differ from each other. Especially, of great importance is having an information on management of the work progress between different parts of different variants and then select the most efficient one. Dedicated technologies lack on effectively manage the information on processes encoded in process models and process execution records [17]. For more than a decade process-oriented data warehouse have been introduced as a solution on analysing effectively process activities of an organization. Process Warehouses [2, 8, 16] are an appropriate means for analysing the performance of business process execution using well established data warehouse technology and on-line analytical processing (OLAP) tools. A way to manage these variants is expressing all the variants in a single process definition with the excessive use of XOR-Splits. The resulting processes are large, difficult to understand and overloaded, and new process definitions still comprise of all the past processes definitions they should replace.
To address these shortcomings we propose a process warehouse model which allows to express a generalization hierarchy of processes to adequately capture process variants. This generalization hierarchy can be generated from a metamodel of business process models which introduces the notion of generic activities which generalize a set of activities (e. g., pay by credit card, by check, or by third-party (PayPal) could all be generalized to an activity payment). Based on these given hierarchies of activities we can define generalization hierarchy of processes for the "process" dimension of a process warehouse. This hierarchy can then be used to roll-up or drill down when analysing the logs of the executions of the various process variants and it makes it much easier to compare key-performance indicators between different variants at different levels of genericity. In this context, the main research question this paper addresses is:
RQ: How can a family of process variants be effectively and efficiently analysed using a process warehouse approach?

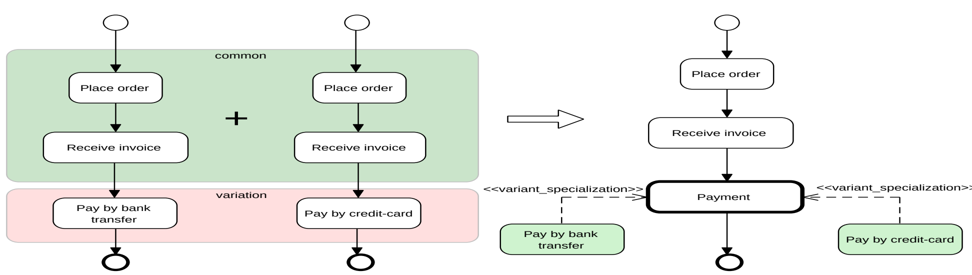

Fig. 1: An example of reference(global) process model abstracting multiple process variants

This research question specifies the interoperability between business process modelling, enactment and data warehouse research areas with the aim of analysing different variants in a multidimensional perspective. To identify how effective (e. g. measure customer satisfaction for a product or process) and efficient (e. g. measure time, cost and resource utilization) a business process is, a process performance analysis is crucial. Moreover it helps in estimating process improvement efforts. To understand how a reference or global process model is constructed, let us consider a concrete example that refers to customer invoice payments after ordering his/her goods or services. Figure 1 shows two variants of the order-to-pay process represented Business Process Modelling Notation (BPMN)[6]. These variants reflect two possibilities to pay: the first pay by bank transfer (filling a bank statement), the other, pay by credit-card (check customer balance). We show how a reference can be constructed by identifying the commonalities and variability among them. The choice between pay by credit-card or pay by bank transfer represents a variability in this process: depends on different drivers such as type of invoice, type of goods etc. The two variant activities are integrated to a new generic (abstract) activity named *Payment* as shown on the right-hand side of the figure. We use a stereotype named ≪*variant_specialization*≫ assigned to the generic connector between the generic activity and the specialized activities. We present our method to deal with process variants, specifically we design a meta-model to adequately capture process variants by introducing two new notions of generic activities and generic processes and to define specialization/generalization relationships between them. This meta-model is an extension and alteration of this work [4] we published years ago. The remaining of this paper is as follows: chapter 2 gives the algorithm developed to generate the process variant hierarchy, chapter 3 reviews the literature and chapter 4 finally draws some conclusions.

2 Generate a process variant hierarchy

In this section, we show how to apply these transformation sets and afterwards represent them in a consolidation hierarchy. We develop an algorithm to generate all activity steps of concrete processes derived from generic processes of the Reference process model after applying substitution of each generic activities with respective activity specializations. The steps of this algorithm are as follows: -Firstly, filter (procedure FILTER_STEPS() is left out due to space limitation) only some specific occurrences after applying the sequential order of steps from concrete and generic processes; -Secondly, we generate all steps of concrete processes derived from generic processes after applying direct and non-direct specializations of GAs as described in Algorithm 1 . For each direct specialization of generic activities we obtain respective activities from concrete processes as bounded activities. Whereas, for non-direct specializations we use a breadth-first search strategy to explore other activities starting from the specialized activity up to the last activity of a concrete process. -Thirdly, after configuring the genericity levels of the hierarchy by ranking rows according to *lvl*-(GAs absolute level) values. Detail explanations of this algorithm is given in the publication in [1]

[1] Report:https://doi.org/10.5445/IR/1000146946

Algorithm 1 Derive process variants hierarchy after applying direct and non-direct specializations of GAs from generic processes

Input: $all\_Filtered\_Steps \leftarrow$ FILTER_STEPS()

Output: A multiset $PV\_Hierarchy$ with tuples $\{(act\_id, ga\_id, process\_id, lvl)\}$

Variables

$step\_ID = \pi_{StepId}\ (all\_Filtered\_Steps)$

$process\_ID = \pi_{ProcessId}\ (all\_Filtered\_Steps)$

$activity\_ID = \pi_{ActivityId}\ (all\_Filtered\_Steps)$

$lvl = \pi_{lvl}\ (all\_Filtered\_Steps)$ ▷ **step level**

$isGeneric = \pi_{isGeneric}\ (all\_Filtered\_Steps)$ ▷**isGeneric=1 i.e. step is a GA**

procedure DERIVE_PV_HIERARCHY()
    $PV\_Hierarchy \leftarrow \emptyset$ ▷ multiset of process variants specializations

    foreach $step\_ID \in all\_Filtered\_Steps$ **do**
        if $activity\_ID$ *is not null* **then** ▷ skip control element steps
            if $isGeneric = true$ **then** ▷ check if $step\_ID$ is a GA
                ▷ get direct GA's specialization i. e., an EA or SP
                $substituted\_step \leftarrow \pi_{ActivityId}\ \sigma_{GenericActivityId=step\_ID}(a\_is\_spec\_of\_ga)$
                $bounded\_step\_a \leftarrow \pi_{S\_Bound.act\_id,\mathbf{step\_ID},\mathbf{process\_ID},lvl}(\rho_{Sub\_S}(substituted\_step) \times \rho_{S\_Bound}(\text{GETBOUNDEDSTEP\_OF\_A}(Sub\_S.ActivityId)))$
                ▷ insert into multiset $PV\_Hierarchy$ with current tuples
                $PV\_Hierarchy \leftarrow PV\_Hierarchy \cup bounded\_step\_a$
                ▷ get indirect GA's specialization using breadth-first-search strategy
                $indirect\_step\_a \leftarrow \pi_{BFS.act\_id,\mathbf{step\_ID},\mathbf{process\_ID},lvl}(\rho_{S\_Bound}(bounded\_step\_A) \times \rho_{BFS}(\text{BREADTH\_FIRST\_SEARCH}(S\_Bound.act\_id)))$
                ▷ insert into multiset $PV\_Hierarchy$ with current tuples
                $PV\_Hierarchy \leftarrow PV\_Hierarchy \cup indirect\_step\_a$
                ▷ derive and store process specializations
                get $concretePId\_of\_a$ for $bounded\_step\_a$
                get $concretePId\_of\_ind\_a$ for $indirect\_step\_a$
                $p\_is\_spec\_of\_gp \leftarrow p\_is\_spec\_of\_gp \cup \{(concretePId\_of\_a, process\_ID)\} \cup \{(concretePId\_of\_ind\_a, process\_ID)\}$
            end if
        end if
        get next $step\_ID$ from $all\_Filtered\_Steps$
    end foreach
    ▷ rename $PV\_Hierarchy$ attributes set
    $PV\_Hierarchy \leftarrow \pi_{act\_id,\rho_{ga\_id/step\_ID},\rho_{process\_id/process\_ID},lvl}(PV\_Hierarchy)$
    return $PV\_Hierarchy$

end procedure

Algorithm 1 as described below generates all steps of concrete processes derived from generic processes after applying direct and non-direct specializations of GAs. As a result, from all these process specializations after substitution operations we derive process variant hierarchy. Accordingly, *Process Variant* dimension in our process warehouse model stores records of this hierarchy. Detailed information the interested reader may find in [3] thesis.

3 Related Work

This section provides an overview of current process-oriented data warehouses approaches to analyse business processes through the most important analysis parameters which are elicited from the generic meta-model of business processes. [2] proposes to derive a generic data warehouse structures from the meta model of the BPMN, whereas [11] proposes a Sequence Warehouse (SeWA) architecture and OLAP tools to analyse data stemming from workflow logs but a conceptual model for DW is missing. Approaches based on goal-oriented methodology for requirement analysis in order to design a data warehouse were proposed by [10, 15, 19]. A multidimensional data modelling for business process analysis was proposed by [7, 13, 14, 16] During our research work we have found a number of relational and multidimensional data warehouse design used for process mining analysis as well. According to approach in [1], process cubes notion is presented to organize events and mined process models using different dimensions. Authors in [18] introduced an event cube as a basis for process discovery and analysis. A framework is further developed to realize a process cube allowing for the comparison of event data in [12]. A hierarchy level was defined only in the time dimension. For instance, multidimensional process mining can be used to analyse the different versions of a sales process, where each version can be defined according to different dimensions such as location or time, and then the different results can be compared as proposed in [5]. Furthermore, authors in [20] partition event logs into groups of cases called sublogs with homogeneous features in a dynamic and flexible way, in order to manage comparisons between models. Whereas, authors in [9] proposed an Abstract Argumentation Framework (AAF) to support the “high-level” analysis that business analysts are used to reason.

4 Conclusions

This paper proposed a process warehouse approach to efficiently and effectively analyse a family of process variants. Current business process management systems and traditional process warehouses lack on adequately abstracting and consolidating all variants into one generic process model, to provide the possibility to distinguish and compare among different parts of different variants. As a summary, based on the consumption of PW in many business intelligence development and solutions, a framework that allows process variants to be efficiently analysed can significantly improve the state-of-art.

References

[1] van der Aalst, W.M.P.: Process cubes: Slicing, dicing, rolling up and drilling down event data for process mining. Springer Verlag (2013)

[2] Benker, T.: A generic process data warehouse schema for bpmn workflows. Springer International Publishing (2016)

[3] Berberi, L.: Analysis of Process Variants with a Process Warehouse Approach. Ph.D. thesis (2021)

[4] Berberi, L., Eder, J., Koncilia, C.: A process warehouse model capturing process variants. EMISA Journal (2018)

[5] Bolt, A., van der Aalst, W.M.P.: Multidimensional process mining using process cubes (2015)

[6] BPMN(Spec.): About the business process model and notation specification version 2.0. Tech. rep., OMG, https://www.omg.org/spec/BPMN/2.0/PDF

[7] Casati, F., Castellanos, M., Dayal, U., Salazar, N.: A generic solution for warehousing business process data. VLDB Endowment (2007)

[8] Eder, J., Olivotto, G.E., Gruber, W.: A data warehouse for workflow logs. Springer (2002)

[9] Fazzinga, B., Flesca, S., Furfaro, F., Pontieri, L.: Process mining meets argumentation: Explainable interpretations of low-level event logs via abstract argumentation (2022)

[10] Giorgini, P., Rizzi, S., Garzetti, M.: Goal-oriented requirement analysis for data warehouse design. ACM (2005)

[11] Koncilia, C., Pichler, H., Wrembel, R.: A generic data warehouse architecture for analyzing workflow logs. Springer (2015)

[12] Mamaliga, T.: Realizing a Process Cube Allowing for the Comparison of Event Data. Master's thesis, TU Eindhoven (2013)

[13] Mansmann, S., Neumuth, T., Scholl, M.H.: Olap technology for business process intelligence: Challenges and solutions (2007)

[14] Neumuth, T., Mansmann, S., Scholl, M.H., Burgert, O.: Data warehousing technology for surgical workflow analysis. IEEE (2008)

[15] Niedrite, L., Solodovnikova, D., Treimanis, M., Niedritis, A.: Goal-driven design of a data warehouse-based business process analysis system. AIKED'07

[16] Pau, K.C., Si, Y.W., Dumas, M.: Data warehouse model for audit trail analysis in workflows. IEEE (2007)

[17] Polyvyanyy, A., Ouyang, C., Barros, A., van der Aalst, W.M.: Process querying: Enabling business intelligence through query-based process analytics. Decision Support Systems (2017)

[18] Ribeiro, J.T.S., Weijters, A.J.M.M.: Event cube: Another perspective on business processes (2011)

[19] Shahzad, K., Zdravkovic, J.: Process warehouses in practice: a goal–driven method for business process analysis. Journal of Software: Evolution and Process (2012)

[20] Vogelgesang, T., Appelrath, H.J.: A relational data warehouse for multidimensional process mining. Springer (2015)