A Generic Process Data Warehouse Schema for BPMN Workflows

Thomas Benker(✉)

Chair of Business Information Systems – Systems Engineering
Otto-Friedrich-University of Bamberg, An der Weberei 5, 9047 Bamberg, Germany
thomas.benker@uni-bamberg.de

Abstract. Companies in dynamic environments have to react to certain market events. Reactions can be short-term and influence the behavior of running process instances or they can be mid-term or long-term and cause the redesign of the process. In both situations, insights into the process flow are necessary and provided by Process Data Warehouse Systems. This paper proposes to derive the data warehouse structures from the meta model of the BPMN (Business Process Model and Notation), the actual de-facto standard of workflow languages. The resulting data structure is generic in order to be portable between application domains and to be stable in case of changing workflows.

Keywords: Process data warehouse system · Data warehouse schema · Business Process Model and Notation · Workflow

1 Introduction

Data Warehouse Systems (DWH systems) are established in most companies. They are used to support strategical and tactical decisions at managerial level by providing relevant multidimensional information. The multidimensional data model enables the aggregation and analysis of quantitative measures (e.g., number of sales) along qualitative dimensions (e.g., region, customer group or time). The schema design is aligned to certain areas of business decisions and analysis. The measures are provided aggregated and abstracted from concrete business transactions or processes.

Companies in highly competitive and dynamic markets often have to react appropriately and with an adequate latency to changing conditions [1]. Process Data Warehouse Systems (PDWH systems), a specialization of subject-oriented DWH systems, are appropriate to optimize business processes and operations on a daily or intraday basis [2]. They can be used to support the identification of changing conditions as well as the design of adequate reactions by providing insights into the processes. The difference to the subject-oriented DWH concept is shown in Fig. 1. A PDWH system focuses on a certain process type while subject-oriented DWH systems are providing data abstracted from process type information. This is also true of the data instances. A PDWH system has a lower aggregation level and explicitly provides data that is related to a process and its behavior. Processes are executed by Workflow Management Systems (WfMS). These systems also keep track of the execution and enable some runtime monitoring and

W. Abramowicz et al. (Eds.): BIS 2016, LNBIP 255, pp. 222–234, 2016.
DOI: 10.1007/978-3-319-39426-8_18

restricted analytical functions at a technical level. But they are restricted in case of business level analysis in order to support the following PDWH scenarios: The behavioral (operational) scenario describes short-term reactions that influence the behavior of running process instances. The structural scenario describes the mid-term or long-term redesign of the process structure based on data of finished instances [3]. The process relationship is the common basis as well as the multidimensional data structuring. E.g., [4, 5] motivate multidimensional structures for the structural scenario. For the behavioral scenario the multidimensional data model is proposed to analyze process events in a historical context [6, 7]. Of course, the data structure has to be extendable with scenario-specific information.

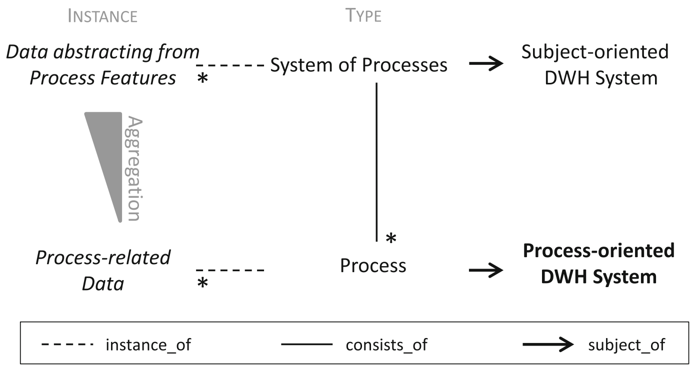

Fig. 1. Distinction of subject- and process-oriented DWH systems

The term *process* is rather unspecific in the context of PDWH systems. The term *workflow* is more precise and better suited. A workflow is understood as a special kind of process that is designed to be executed by humans or WfMS. Basically, it is described by activities and their relationship [8]. Workflow languages are close to the executing systems and for this appropriate as basis to derive multidimensional structures. The Business Process Model and Notation (BPMN) [9] is one of the dominant workflow modeling languages [8]. The OMG (Object Management Group) standard is widespread and accepted by modelers and tool developers [10]. But the BPMN has not been considered so far in multidimensional process data structures.

The goal of this paper is to present a multidimensional data structure in order to realize the following requirements: First, the data structure should support decision making in the behavioral and the structural PDWH scenario. Second, it should support the workflow language BPMN. And finally, the multidimensional structure should be flexible in order to react on dynamically changing workflow schemas. The strategy to realize these requirements is to derive the multidimensional data structure based on the BPMN meta model. Of course, it has to be enhanced with business information. Further, the data structure will be specified to be generic. This means that it is independent of the workflow schema. First, this enables the application in heterogeneous business domains (e.g. sales or human resources). Second, the redesign of a workflow schema due to changing market conditions does not imply the redesign of the DWH schema.

The benefit is that existing reports are repeatable and results are reproducible. This is a key feature of DWH systems (e.g. in case of compliance checking). The redesign of DWH structures often has a negative influence on repeatability and reproducibility.

To introduce the generic PDWH schema, the paper is structured as follows: First, the relevant basics of the BPMN are introduced. Section 3 is a discussion of related work. Section 4 presents the concept of the generic PDWH schema that is applied and demonstrated in Sect. 5 on a real-world case study. Finally, Sect. 6 summarizes and reflects the concept and gives an outlook on future work.

2 The Business Process Model and Notation

PDWH systems focus on process behavior and interaction between processes. The BPMN provides *process* (workflows) and *collaboration diagrams* for this purpose [11]. The other diagram types are not relevant for the research problem. The BPMN also defines conformance classes in order to determine the conformance between modeling tools and the specification. To control the complexity of the BPMN in this work, the focus of the presented concept is reduced to the elements of the *descriptive process modeling conformance* class. This class defines the basic concepts for process and collaboration modeling. Section 6 shortly explains the extension of the concept to support the *full process modeling conformance*. The meta model for the conformance class is based on the BPMN specification [9] and shown in Fig. 2.

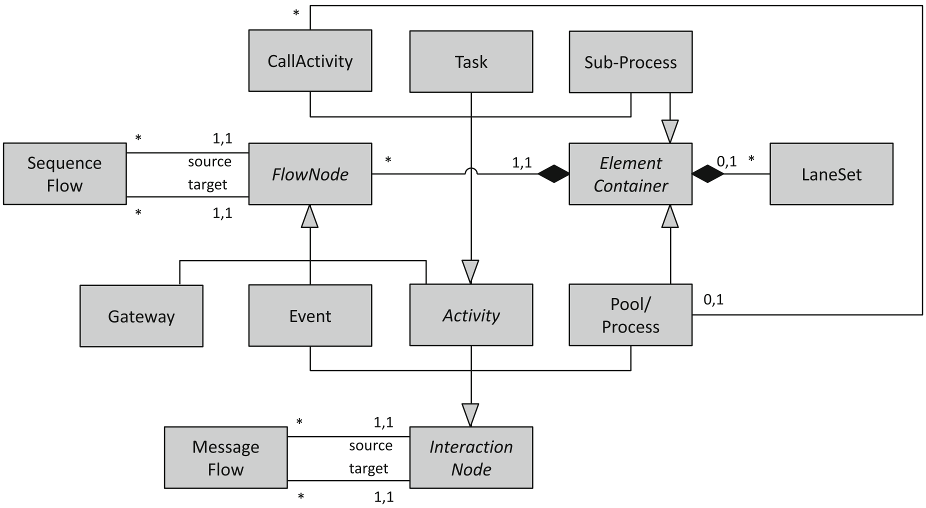

Fig. 2. The relevant part of the BPMN meta model (representation based on [9])

Flow Nodes [9, 11]: Flow Nodes are the building blocks of BPMN process models. *Activities* are used to model working steps. A *Task* represents an atomic working step that can be described more closely as *user*, *service*, *send* or *receive task*. A *Sub-Process* is a non-atomic working step and encapsulates a process itself. A *Call Activity* is used to model a callable and reusable process. *Gateways* control the flow of the process. An *exclusive* gateway splits/joins alternating sequence flows. *Parallel* gateways create/synchronize parallel sequence flows. *Events* of a process are differentiated by their position (*start*, *intermediate*, *end*) and their trigger. Relevant events are the *empty start/end event*, the *message start/end event*, the *timer start event* and the *terminate end event*. Sub-types of *Task*, *Event* and *Gateway* are not shown in Fig. 2.

Connecting Objects [9, 11]: The *Sequence Flow* is used to model the process behavior. A Sequence Flow element always runs from a start to a target *Flow Node*. The interaction between processes is modeled by *Message Flows* with optionally annotated message descriptions.

Lanes [9, 11]: A process is contained within a *Pool* that represents a participant of a collaboration. A Pool can be hierarchical structured by *Lanes*. Typically, they are used to model roles or responsibilities within a process.

The *Data Objects* of the BPMN specification are left out of this work. The BPMN semantics of the term *Data* are not precise enough and because of this not operable. The element *Data* is not only used to annotate input/output data. Even more, it is used to assign physical objects and products. Further, the BPMN does not provide the possibility to specify data structures like UML Class Diagrams or Entity Relationship Models. But such data structures could be important to identify business dimensions. Because of this, Sect. 5.2 demonstrates the derivation of business dimensions based on object-oriented operational structures.

3 State of the Art

A number of publications are presenting multidimensional data structures for workflow schemas. A first group of publications base their concepts on informal [12] or proprietary [4, 13] workflow specifications. The authors of [14] present a generic multidimensional schema. The used workflow specification is kept abstract. The proprietary workflow specifications are often incomplete, restricted and only for theoretical usage. A second group of concepts is defined for certain application domains. Multidimensional data structures for surgery workflows are presented in [5]. In [15] parts of multidimensional data structures for service and sales processes are introduced. Because of their domain-specific contexts, the portability to other application domains is restricted. The authors of [16] base their concept on the workflow specification of the WfMC (Workflow Management Coalition). The multidimensional data structures are not specified in general terms. They are designed to support assumed queries. The data structure of [17] abstracts from application domain and process language. It is designed for the identification of information requirements based on generated process data. Due to the abstraction from language specifics it is limited for tracking and analyzing relevant processes data at runtime.

The presented approaches support heterogeneous workflow concepts as dimensions (Table 1). Only few approaches support the analysis of the hierarchical structures or of the sequence flow of the workflows. Data usage is mostly considered rudimentary or only domain-specific. The dimensions *Time*, *Actor* and *Instance* (Assignment of node instances to workflow instance) are supported in all proposals. Yet, the BPMN has not been considered in related work. Compared to the BPMN, the used workflow specifications and multidimensional structures are restricted. E.g., the interaction between participants/processes is not subject of any of the concepts. Furthermore, the related multidimensional structures are not defined based on a formal specification and, for this, could not be checked for completeness. A further feature of the BPMN is its clearly defined and human readable graphical syntax. This is important for process redesign in order to localize workflow elements that, for example, have been identified to be inefficient. The related publications do not discuss workflow specifications with an appropriate graphical syntax compared to the BPMN.

Table 1. Supported dimensions in related concepts

	[4]	[5]	[12]	[13]	[14]	[15]	[16]	[17]
Instance	X	X	X	X	X	X	X	X
Hierarchy		X	X				X	X
Sequence		X			X			X
State		X	X	X	X		X	X
Time	X	X	X	X	X	X	X	X
Actor	X	X	X	X	X	X	X	X
Organization	X		X	X		X	X	X
Data		dom	X	X	dom	dom		X
Events		X	X					

X → *feature is realized*; dom → *domain-specific realization*

4 The Generic Process Data Warehouse Schema

The PDWH schema is derived and justified based on the meta model of the BPMN (Fig. 2). The data instances of a PDWH schema should be provided not aggregated as they are available at the source system (workflow engine) [18, 19] in order to flexibly realize ad-hoc queries. Nodes (Task, Gateway, Event) are the low-level elements of the BPMN and, for this, the lowest level for dimensions of the multidimensional structure. Figure 3 shows the resulting dimensions that are derived based on the element *Flow Node* of the meta model. For representing the multidimensional schema, the *Semantic Data Warehouse Model* (SDWM) [20] is used. A dimension is shown with all of its hierarchy levels and the aggregation relationships between these levels. The highest level (*Total*) means a full aggregation of the dimension hierarchy. Normally, there is a n:1-aggregation relationship between a hierarchy level and the next higher one. As graphical representation crow's feet are used at the n-side. The SDWM assumes that a hierarchy level is implicitly described by ID and a semantic attribute.

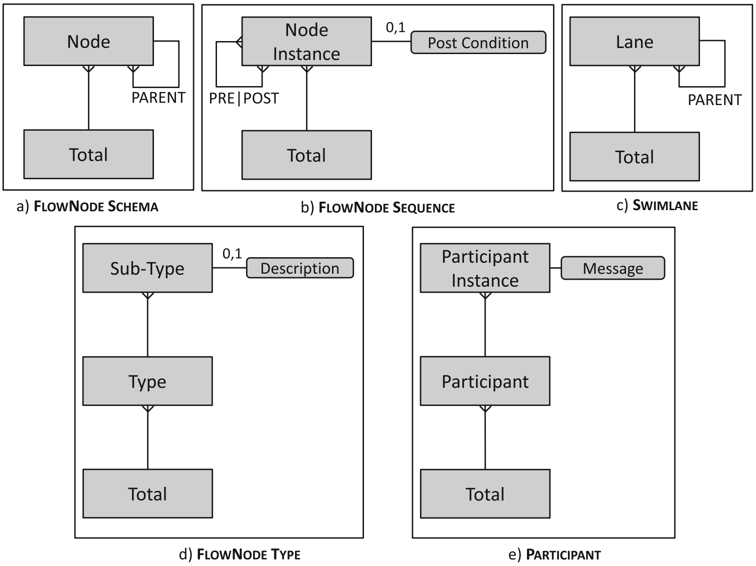

Fig. 3. BPMN-specific dimensions

Scope of the PDWH schema is a *BPMN process* (workflow). It is embedded within a *collaboration diagram* to show the interaction with other participants. A multidimensional data structure covers the process of one participant. The other participants are regarded as black boxes. Their processes can be subject of separated PDWH schemas.

FlowNode Schema: The dimension *FlowNode Schema* (Fig. 3a) is used to make the workflow and nodes analyzable at schema level. At the lowest level of this dimension the semantic of Node is its name within the workflow schema. Nodes may be hierarchically structured (*Parent*-relationship). E.g., a Task may be part of a Sub-Process and the Sub-Process may also be part of a higher Sub-Process itself. So the dimension is unbalanced at the instance level. For the *FlowNode Schema* dimension, the *Total* level means the aggregation up to the whole workflow.

FlowNode Sequence: The behavior of a certain workflow instance is tracked and analyzable through the dimension *FlowNode Sequence* (Fig. 3b). A *Node Instance* can have zero or multiple pre- and successors (pre/post-relationship). Multiple successors are, for example, the result of parallel gateways. This enables the aggregation along the behavioral path of a workflow instance. An aggregation over the whole instance is modeled by the *Total* dimension level. The *Post Condition* is a dimensional attribute that is especially useful to understand gateway behavior.

Swimlane: The dimension *Swimlane* (Fig. 3c) is derived from the meta element *LaneSet*. It enables the analysis of organizational structures within a participants workflow schema. In a collaboration diagram, a node is assigned to a *Lane*. *Swimlanes* are hierarchically structured with an a priori unknown depth. This is considered by the parent relationship.

FlowNode Type: The dimension *FlowNode Type* (Fig. 3d) is a generic dimension to consider the node types and enables the aggregation along the BPMN-type hierarchy (generalization of *Flow Node* for *Event*, *Activity*, and *Gateway* in Fig. 2). The instance level of the dimension is common for all BPMN workflows. The aggregation of the *Sub-Types Exclusive* and *Parallel* Gateway to the *Type Gateway* is an example for the instance level. The attribute *description* could be used (0,1-relationship) to provide further information (e.g., *message* as trigger of a *Start Event*).

Participant: The dimension *Participant* (Fig. 3e) is necessary to analyze the interaction within a collaboration between the instance of a workflow and of related participants. It is closer described by the message exchange. The dimension enables the aggregation of participant instances to types and the *Total* level.

To enable the aggregation of all workflow instances, a dimension *Process Instance* is needed that is not derived from the meta model. This also applies to the obligatory dimension *Time* of DWH schemas, which enables the analysis of workflow executions on the time axis.

Finally, measures have to be identified for the multidimensional schema. Following the argumentation of Boehnlein [20], measures of a DWH system are defined to assess certain objectives. The objectives for the structural DWH scenario (process design) that could be assessed in a workflow schema are (i) short process runtime, (ii) low process costs, and (iii) high process quality [21]. The measures *time* and *cost* are appropriate to assess objective (i) and (ii). For the assessment of the process quality a number of publications propose static procedures [22]. A PDWH system could be used additionally to analyze the true behavior at runtime. The analysis of normal and exceptional behavior may be an example. So, a counting measure (*number* of executions) is necessary to track the executed activities. Based on these measures, further case-specific measures could be derived. Measures for the behavioral scenario are application-domain dependent and cannot be specified in general terms.

5 Application of the Process Data Warehouse Schema

Section 5 demonstrates the application of the generic multidimensional data structure on an extract of a real-world case study (Sect. 5.1). Section 5.2 shows the extension of the data structure by business dimensions to support the structural DWH scenario.

5.1 Application of the Generic Data Structure

The generic PDWH schema is demonstrated by application to the sales process of an e-car (electronic car) rental company. The sales process is specified according to the case study presented in [23]. Figure 4 shows an extract of the BPMN sales workflow.

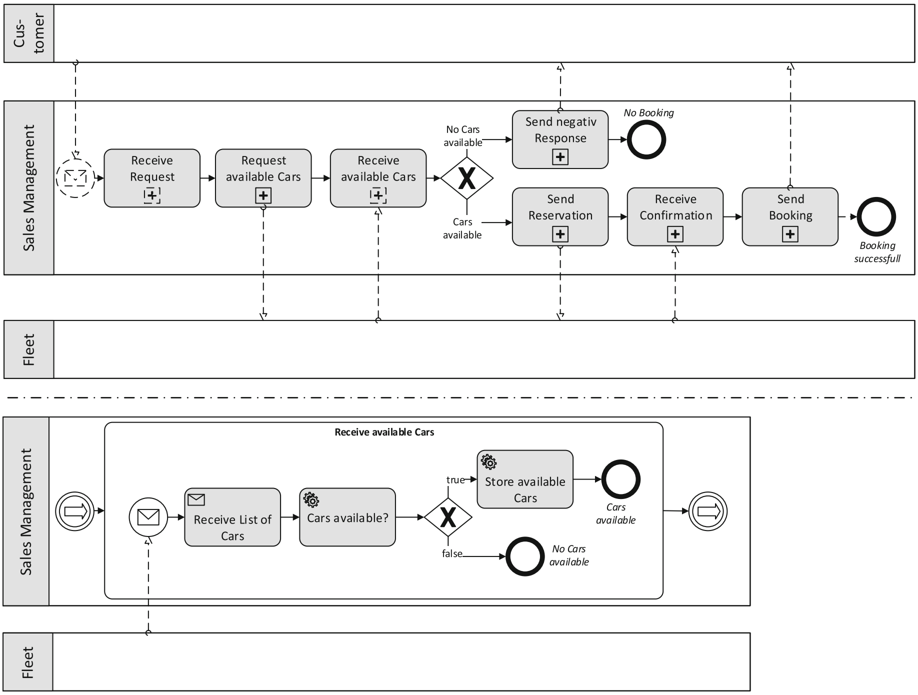

Fig. 4. Sales management workflow for e-Car rental

The upper part of Fig. 4 shows the overall process starting with the receipt of a customer request and ending with successful or non-successful booking. The availability of e-Cars is the crucial success factor. The process is specified within a collaboration diagram and interacts with the black box participants *Customer* and *Fleet Management System*. The main activities are modeled as embedded and collapsed Sub-Processes. The lower part of Fig. 4 shows the insight of the Sub-Process *Receive available Cars*. The *Fleet* participant sends a list of available cars. Only if cars are available, they are persisted locally and the Sub-Process ends up in the final state *Cars available*. Otherwise, the final state is *No Cars available*.

Figure 5 illustrates the application at instance level using tables according to the star schema paradigm of relational DWH systems. The *Facts* table combines the dimensions by foreign key relationships and has the attributes for the measures. The example shows the measure *n* (number of executions). The dimensions for process instance (*process*) and time (*timestamp*) are not shown explicitly. They are only considered by the attributes *instance* and *timestamp*.

FlowNode Schema		
NODE-ID	NODE-DESC	PARENT
[...]	[...]	[...]
2	Request available Cars	-
3	Receive abailable Cars	-
4	Decision Cars available	-
301	Event Receive List of Cars	3
302	Receive List of Cars	3
303	Cars available?	3
304	Decision Cars available	3
305	Store available Cars	3
306	Event Cars available	3
[...]	[...]	[...]

FlowNode Sequence			
INSTANCE	PRE	POST	P.COND.
[...]	[...]	[...]	[...]
2-1	[...]	3-1	-
3-1	2-1	4-1	-
4-1	3-1	[...]	[...]
301-1	-	302-1	-
302-1	301-1	303-1	-
303-1	302-1	304-1	-
304-1	303-1	305-1	true
305-1	304-1	[...]	[...]
[...]	[...]	[...]	[...]

FlowNode Type			
TYPE-ID	SUBTYPE	DESC.	TYPE
T1	start_event	message	event
T2	embedded_sub_process	-	sub_process
T3	exclusive_gateway	-	gateway
T4	receive_task	-	task
T5	service_task	-	task
T6	send_task	-	task
[...]	[...]	[...]	[...]

Participant			
P-ID	INSTANCE	P-MESSAGE	PARTICIPANT
P1	Fleet-App	List of Cars	Fleet
P2	[...]	[...]	[...]

Facts						
NODE-ID	INSTANCE	PROCESS	TYPE-ID	P-ID	TIMESTAMP	N
[...]	[...]	[...]	[...]	[...]	[...]	1
2	2-1	1	T2	-	2015-12-13T13:15:20	1
3	3-1	1	T2	-	2015-12-13T13:15:50	1
4	4-1	1	T3	-	2015-12-13T13:18:45	1
301	301-1	1	T1	P1	2015-12-13T13:15:50	1
302	302-1	1	T4	-	2015-12-13T13:15:52	1
303	303-1	1	T5	-	2015-12-13T13:16:40	1
304	304-1	1	T3	-	2015-12-13T13:17:50	1
305	305-1	1	T5	-	2015-12-13T13:17:57	1
[...]	[...]	[...]	[...]	[...]	[...]	1

Fig. 5. The process data warehouse schema of the sales management workflow

A node instance (data set of the *Facts* table) is identified by the attribute *instance* and is assigned to a process instance by the attribute *process* (e.g., instance = 301-1). The table *FlowNode-Type* shows for instance 301-1 (type-id = T1) that it is of the Sub-Type *message start event* and could be aggregated to the Type *event*. For the same node instance the interaction with the participant instance *Fleet-App* (p-id = 1) is associated in table *Participant*. The message exchange is described as *List of Cars*. Node instance 301-1 is further defined by the schema element 301. The table *FlowNode Schema* shows the semantic description of the event *Receive List of Cars* as well as its integration in the workflow hierarchy. It is a child of node 3, the Sub-Process *Receive available Cars*. The behavior of process instance 1 is shown in table *FlowNode Sequence*. E.g., node instance 304-1 is an exclusive gateway. The decision of this gateway in process instance 1 is true, meaning cars are available and node instance 305-1 is executed next.

It is also possible to manage loops in a workflow schema. Each time a node is accessed in a workflow instance, a new node instance is created and put in the *Facts* table. The timestamp and the counter within the node instance description enable to keep track of the execution sequence of the node instances within the loop. The presented data structure is, due to its generic character, able to deal with changes within the workflow schema. Changes in workflow hierarchy and flow node semantics are realized by new instances within the dimension *FlowNode Schema*. Changes in the sequence of the workflow schema are tracked with the workflow instances in the dimension *Flow-Node Sequence*.

5.2 Domain-Specific Extension of the Multidimensional Data Structure

For comprehensive analysis it is necessary to add business-specific dimensions and measures. Customer or sales dimensions would be useful for the presented case study. For example, the sales process could be executed by call center or web shop. The sales dimension would allow to compare performance measures between these types. A customer dimension could show differences in the sales process between young and older customers. Of course, these dimensions could not be derived from the meta model of the BPMN. As mentioned in Sect. 3, the BPMN *Data* semantics are not useful for the derivation of dimensions. According to [20], we suggest to identify business dimensions based on object-oriented diagrams. As mentioned yet, data structures are not provided by the BPMN and have to be specified additionally. The author of [20] gives hints in order to reveal business-specific dimensions in an object-oriented data schema. On the left side Fig. 6 shows an example of the object-oriented structures of the case study. As stated, a *Sales Channel* could be an *e-Shop* or a *CallCenter*. A *CallCenter* is the composition of *CallCenter Employees*.

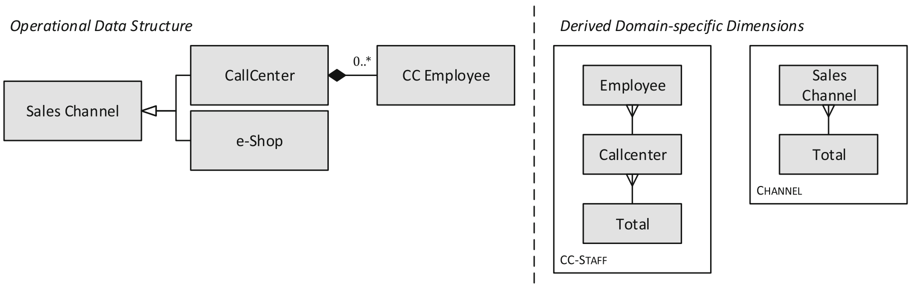

Fig. 6. Domain-specific dimensions of the case study

The right side of Fig. 6 shows possible business dimensions. The generalization of *CallCenter* and *e-Shop* to Sales Channel results in a dimension *Channel*. E.g., it enables to differentiate performance measures of activities/workflows by the sales channel type (*e-Shop* vs. *CallCenter*). The dimension CC-Staff results from the composition relationship of *CallCenter* and *Employee*. Measures could be analyzed at the grain of single employees or for different engaged call center providers.

Generalizations or compositions in object-oriented data structures are just hints for the modeler to identify dimension structures. Of course, domain knowledge and experience of the modeler are important for this task.

6 Summary, Reflection and Future Work

This paper presents a generic Data Warehouse Schema to support process-related decision making. *Generic* means that it is suitable for BPMN processes independent of its application context and stable in case of changing workflow schemas.

As mentioned initially, the work is restricted to the *descriptive modeling conformance* class. Only one additional attribute of the *Facts* table (*association*) is necessary to support complex BPMN elements of the *full process modeling conformance* as *Compensations.* A number of elements are just supported by the presented data structure, e.g. further sub-types of gateways (*complex*, *event-based*) and events (*intermediate*). Other elements are understood as alternating representations of supported modeling scenarios (e.g., looping activities).

According to [24], process warehousing approaches can be reflected by their support of the process modeling perspectives of [25]: *functional*, *behavioral*, *organizational*, and *informational*; subsequently, the presented approach is also reflected based on these perspectives. The initial classification of the BPMN in [26] is used as basis. The *functional* perspective is realized by the dimension *FlowNode Schema*, showing the hierarchical structure and the semantical names of the Tasks. The dimension *FlowNode Sequence* and the dimension *FlowNode Type* build up the *behavioral* perspective. Using Pools and Lanes to specify responsibilities enables the *organizational* perspective. The differentiation between *Service Tasks* and *User Tasks* is also understood as an organizational aspect, as well as the consideration of the interaction between participants. The message exchange as well as Events with their semantics are contributing to the *informational* perspective. The derivation of dimensions from operational data structures (UML class diagram) complements the BPMN-based dimensions and the *organizational* perspective. To summarize, the presented concept can provide comprehensible insights into a workflow.

For future work, the presented schema is planned to be integrated in a holistic development method that enables two features: (i) the integrated development of operational workflow and PDWH structures; (ii) the interpretation of multi-dimensional workflow data in business semantics. The idea is to interpret conceptual process models as business views on more technically oriented workflow schemas.

References

1. Dorn, C., Burkhart, T., Werth, D., Dustdar, S.: Self-adjusting recommendations for people-driven ad-hoc processes. In: Hull, R., Mendling, J., Tai, S. (eds.) BPM 2010. LNCS, vol. 6336, pp. 327–342. Springer, Heidelberg (2010)
2. Panian, Z.: Delivering actionable business intelligence through service-oriented architecture. In: Cordeiro, J., Filipe, J. (eds.) Proceedings of the 10th International Conference on Enterprise Information Systems, pp. 191–194 (2008)
3. Benker, T.: Konzeption einer Komponentenarchitektur für prozessorientierte OLTP- and OLAP-Anwendungssysteme. In: Bamberger Beiträge zur Wirtschaftsinformatik und Angewandten Informatik, vol. 97. Bamberg (2015)

4. Eder, J., Olivotto, G.E., Gruber, W.: A data warehouse for workflow logs. In: Han, Y., Tai, S., Wikarski, D. (eds.) EDCIS 2002. LNCS, vol. 2480, pp. 1–15. Springer, Heidelberg (2002)
5. Mansmann, S., Neumuth, T., Scholl, M.H.: OLAP technology for business process intelligence: challenges and solutions. In: Song, I.-Y., Eder, J., Nguyen, T.M. (eds.) DaWaK 2007. LNCS, vol. 4654, pp. 111–122. Springer, Heidelberg (2007)
6. Eckerson, W.: Best practices in operational BI: converging analytical and operational processes. TDWI Best Practices Report, TDWI (2007)
7. Russom, P.: Operational data warehousing. The integration of operational applications and data warehouses. TDWI Best Practices Report, TDWI (2011)
8. Pütz, C., Sinz, E.J.: Model-driven derivation of BPMN workflow schemata from SOM business process models. Enterp. Model. Inf. Syst. Architect. **5**, 57–72 (2010)
9. OMG: Business process model and notation (BPMN). Version 2.0. http://www.omg.org/spec/BPMN/2.0/PDF
10. Silver, B.: BPMN Method and Style: A Structured Approach for Business Process Modeling and Implementation Using BPMN 2. Cody-Cassidy Press, Aptos (2011)
11. Weske, M.: Business Process Management: Concepts, Languages Architectures. Springer, Berlin (2012)
12. Pau, K., Si, Y., Dumas, M.: Data warehouse model for audit trail analysis in workflows. In: Cheung, S.C., Li, Y., Chao, K.-M., Younas, M., Chung, J.-Y. (eds.) Proceedings of IEEE International Conference on e-Business Engineering. IEEE (2007)
13. Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.-C.: Business process intelligence. Comput. Ind. **53**, 321–343 (2004)
14. Casati, F., Castellanos, M., Dayal, U., Salazar, N.: A generic solution for warehousing business process data. In: Koch, C. et al. (eds.) Proceedings of the 33rd International Conference on Very Large Data Bases, pp. 1128–1137. ACM, New York, NY, USA (2007)
15. Kueng, P., Wettstein, T., List, B.: A holistic process performance analysis through a performance data warehouse. In: Proceedings of the American Conference on Information Systems, pp. 349–356 (2001)
16. List, B., Schiefer, J., Tjoa, A., Quirchmayr, G.: Multidimensional business process analysis with the process warehouse. In: Abramowicz, W., Zurada, J. (eds.) Knowledge Discovery for Business Information Systems, pp. 211–227. Springer, Boston (2000)
17. Sun, X.: Ein szenario- und prototypingbasiertes Konzept zur Informationsbedarfsanalyse für Business-Process-Intelligence-Systeme. Entwicklung und Evaluation. Eul, Lohmar (2014)
18. Baars, H., Sun, X.: Wo sind die Klippen im Prozess? Business process intelligence - das unterschätzte Potenzial. BI Spektrum **8**, 10–12 (2013)
19. Inmon, W.H.: Operational and informational reporting. http://www.information-management.com/issues/20000701/2349-1.html
20. Böhnlein, M.: Konstruktion semantischer Data-Warehouse-Schemata. Dt. Univ.-Verl, Wiesbaden (2001)
21. Fischermanns, G.: Praxishandbuch Prozeßmanagement. Schmidt, Gießen (2013)
22. Overhage, S., Birkmeier, D.Q., Schlauderer, S.: Quality marks, metrics, and measurement procedures for business process models. Bus. Inf. Syst. Eng. **4**, 229–246 (2012)
23. Leunig, B., Wagner, D., Ferstl, O.K.: E-Car-Szenario - Hochflexible Geschäftsprozesse in der Logistik als Integrationsszenario für den Forschungsverbund forFLEX. In: Sinz, E.J., Bartmann, D., Bodendorf, F., Ferstl, O.K. (eds.) Dienstorientierte IT-Systeme für hochflexible Geschäftsprozesse, pp. 15–38. University of Bamberg Press, Bamberg (2011)

24. Shahzad, K., Johannesson, P.: An evaluation of process warehousing approaches for business process analysis. In: Barjis, J., Kinghorn, J., Ramaswamy, S. (eds.) Proceedings of the International Workshop on Enterprises and Organizational Modeling and Simulation, pp. 1–14. ACM, New York, NY, USA (2009)
25. Curtis, B., Kellner, M., Over, J.: Process modeling. Comm. ACM **35**, 75–90 (1992)
26. Korherr, B.: Business Process Modelling - Languages, Goals and Variabilities (2007)