Context-driven Business Process Modelling

Matthias Born[1], Jens Kirchner[1], and Jörg P. Müller[2]

[1] SAP Research, Karlsruhe, Vincenz-Priessnitz-Strasse 1, 76131 Karlsruhe
{mat.born, jens.kirchner}@sap.com
[2] Department of Informatics, Clausthal University of Technology
joerg.mueller@tu-clausthal.de

Abstract. Business professionals create business process models in order to get an understanding of who is involved and which resources are needed for the execution of a business process. Such business process models are required as a basis for knowledge transfer, quality purposes, regulations, communication between internal and external collaborative partners, or documentation in general. Current process modelling approaches and tools are still very costly, error-prone, and often used in an insouciant manner. For example, changes affecting similar process artefacts need to be executed manually in each embedding model. We aim to increase the reusability of business process knowledge and propose the integration of a context-driver principle for business process modelling.

1 Introduction

Business process modelling is the first phase of the business process management lifecycle [1]. The actual reasons why organisations model their processes may be quite different [2]. Business process models depict and describe activities that contribute to the production of goods or delivery of services taking place in an enterprise. This work focuses on the conceptual business process modelling phase which depicts processes on the level of activities, sub-processes, and the control-flow between them [3]. The target group comprises professionals with business knowledge who typically do not have a strong technical background. We concentrate on how business process models are used as a way of documentation, communication, and collaboration of business needs. Hence, the actual purpose of a process model in our work is to provide various information elements to its users. Such business process models can be required for quality purposes or compliance, with regulations to document processes explicitly. For example, a business process model contains information about what activities are composed in a process, who is responsible for performing certain activities, when and where they are executed, how and why they are performed, and what business objects, and which data is manipulated. Different business process modelling techniques are available and each of those techniques highlights different information aspects to answer these questions [4]. In order for enterprises to benefit from process modelling efforts, business analysts require better support in creating process models. However, current process modelling tools provide very little guidance to the user and, thus, resemble a scratchboard rather than a technical approach. If any guidance is provided at all, it is on a purely syntactical level.

This paper is organised as follows. Next, we present an overview about the problem statement, starting by an analysis of the requirements (Section 2.1) and following by a running example (Section 2.2). Based on this, we describe the application of a context-driver principle for business process modelling (Section 3). We will then present relevant related work in Section 4 and conclude the paper with Section 5.

2 Problem statement

2.1 Requirements analysis

In order to gain a deeper understanding of the requirements and problem areas of current business process modelling approaches, we have arranged several workshops with different target groups covering a variety of industry sectors. Amongst others, we have talked to business analysts, business consultants and technicians, SAP internal groups and its customers, covering retail, real-estate, telecommunication, and high-tech industries, as well as public administration. The workshops have been organised in a semi-structured way and included open discussions about general and industry related problems. Based on this analysis, we argue that current process modelling approaches and tools are still very costly and error-prone since users are not guided in any sensible way. This statement is supported by an analysis of Gartner [5] that shows that business process modelling tools are too complex for the average user and that the major modelling tool with a market share of 30% is Microsoft Visio [6]. Vendors still construct their own model representations and tools, which are mostly not interchangeable, and yet most business analysts are using office tools (like Microsoft PowerPoint or Microsoft Excel) to describe their process landscape [7, 8]. Instead of reusing existing process models or parts of process models, they are usually replicated and modelled from scratch. In addition to the low level of reusability, a further drawback is that there is no common understanding of business process models and their involved objects of the business world. Modellers are free to name and describe process artefacts in any arbitrary way [7, 8].

2.2 Motivation

Let us assume that there exist two departments, namely the financial department and the sales department within a company, which are using their own terminology in order to model and describe the processing of an invoice (compare Figure 1). As the naming of business process artefacts is often more art than science [9] the modellers may apply their own naming conventions and name business artefacts in arbitrary ways. Figure 1 shows that both processes actually describe the same process, while both departments use their own terminology. Apart from using a different terminology, both process models have a slightly different order of the activities. In this example, the financial department has to update the order after the invoice was created. Besides that the sequence of both process flows is the same. One further distinction is that the task "Bill Creation" does not need "Outgoing goods" as an input. This example highlights that a reuse of business artefacts is difficult. We distinguish between two basic levels of

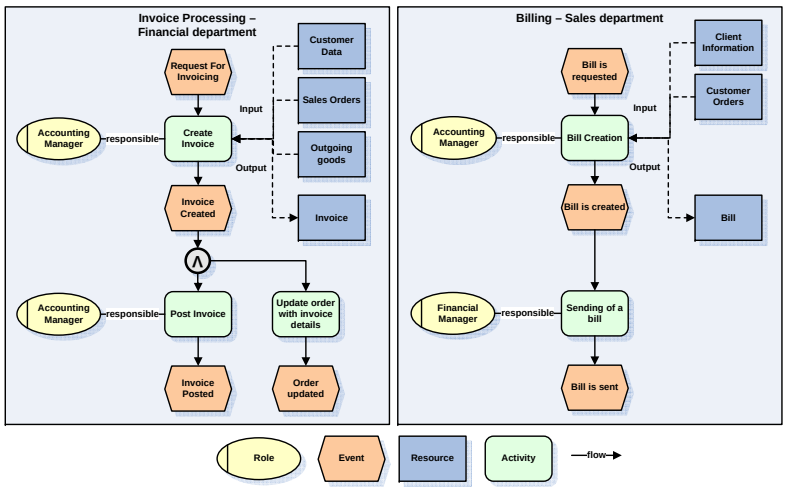


Fig. 1. Simple example of an Invoice Processing process

reusability: On the one hand, reusability of the semantic representation and on the other hand, reusablity of the structural characteristics of business process artefacts.

(a) Reusability across process models, i.e., parts of a process model, e.g., the activity "Create Invoice" in the "Invoice processing" process model in Figure 1, may also occur in other process models (e.g. sales process). However, different creators of process models typically come up with different labels for the same activity (e.g., another modeller might call the activity "Creation of an invoice").
(b) Reusability of structural dependencies, i.e., elements of the process model are typically linked to one or more elements. For example, the process step "Post Invoice" is linked to the responsible role "Accounting Manager". However, in the second process model the same process step has not only a different label but is also linked to a different responsible role, namely "Financial Manager".

Our work aims to overcome the issue of reusability for both cases. We argue that we can achieve a higher consistency and reusability of process knowledge by introducing a context-driver principle into business process modelling approaches.

3 Context-driven Business Process Modelling

Context awareness plays an essential role for the dynamic appearance of structures and the dynamic linguistic representation of components in our framework. The facets of context awareness within this work are influenced by the context-driver idea of CCTS [10–12]. A process model is defined by its process artefacts, which have a unique semantic representation, and the context in which it is used. The general idea is to create a standardised, consistent, and understandable description of every process artefact using

controlled vocabularies and to link each process artefact to a specific business context. We want to give a brief example. Let us assume a modeller creates an activity "create invoice", which is already available in the repository. Now, the idea is to avoid creating a second instance of this activity, but rather to adjust the valid business context of the activity "create invoice". Logically, a modeller would first need to define, in which business context he is modelling.

We will explain the concepts of the context-driver principle in the next sections and show how it can improve reusability of process artefacts. In general, the conceptual framework is designed to be flexible and extensible.

3.1 Context-driver Principle

Context defines the environment in which a business process artefact is used. On the one hand, the context-driver principle must be sufficiently discrete in order to enable semantically unambiguous precision. In other words, a semantic meaning of a business process artefact is always unambiguous, when considered in a specific context. For example, an input resource *Bank* in an *Issue Management* is not precise if this entity is defined without context. "Bank" describes different objects in the industry area of "Finance" and "Marine". Therefore, it is not possible to define only one "bank", which can be used everywhere. Rather, the context in which "bank" is used adds further semantic meaning. This is similar to the previous approach of Stuhec [10].

On the other hand, a business process specifies the sequence of activities. This information, however, is of less semantic importance and thus only the structural information will be relevant. In this case, the context-driver principle of [10] was extended to support such structural differences. In general, the context-driver principle allows to identify, store, and represent a business process artefact only once while specifying the differences depending on specific context categories (e.g., business process, industry, country, etc). In our example process (Figure 1), each activity has only one unique semantic conceptual instance in the process repository, however, there may be a structural difference (a different predecessor or successor) or even different representations (e.g. synonyms, abbreviations, etc) depending on the context.

We have adapted the meta-model from CCTS [11] in order to provide a flexible mechanism for creating new context categories. In CCTS, business contexts are classified into eight context categories. However, our meta-model is not restricted to a fixed set of context categories. Figure 2 depicts our approach which we will explain in more detail hereafter.

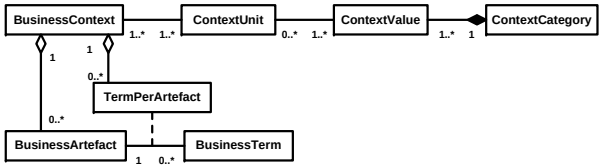


Fig. 2. Meta Model of the adapted Context-driver Principle

Context Categories allow the classification of different context environments, e.g. “Industry”, “Geography”, “Period”, etc. Within this work, we do not focus on the definition of suitable context categories. However, our meta-model allows to add any additional classification. Every context category refers to multiple *Context Values*. For example, the context category “Period” may have the context values “1st Year Quarter”, “2nd Year Quarter”, or “3rd Year Quarter”. A set of arbitrary context values describes the valid environment for a business process artefact. Such a context value set is linked to a *Context Unit*. Multiple context units can be linked to a *Business Context*. Finally, each *Business Artefact* can be described by multiple *Business Terms*, and has exactly one business context assigned. A business term represents the natural language representation for a business artefact. The association between business artefacts and business terms is also linked to a business context and thus specifies the valid linguistic representation for a business artefact in a given business context. Let us recall our running example (Figure 1) where two departments use a different terminology to describe the same business case. Introducing context awareness allows both departments to use their own terminology while using the same conceptual business artefacts. We want to point out that context awareness does not change the semantic meaning of the artefacts but instead delivers a mechanism that different representation can be linked to one common conceptual business artefact.

In our work, we define a business context *as the formal description of a specific business circumstance*.

3.2 Logical definition

Business contexts can be based on mathematical sets for their representation. We will provide an insight of the usage of the set theory within the approach by [12]. The following sets (Listing 1) represent three context categories. Set I represents the industry sector which contains context values such as “Automotive”, “Finance”, “Sales”. Set G represents geographical values which can be the ISO codes for countries (ISO 3166) such as “DE” for Germany, “GB” for the United Kingdom, and “US” for the United States of America. Finally, Set P represents processes which are numbered.

$$I = \{\text{Automotive}, \text{Finance}, \text{Sales}\}; G = \{\text{DE}, \text{GB}, \text{US}\}; P = \{\text{P1}, \text{P2}, \text{P3}, \text{P4}\} \quad (1)$$

Let us assume, we have defined a business artefact and we want to define a business context which specifies that the artefact is valid in the sales and finance sector within Germany and the United Kingdom. It should be also applicable for process P1 and P4. In order to build a valid business context, we need to define a context unit which builds a Cartesian product of subsets of the context categories. These subsets are listed as set I', set G', and set P' ($I' \subseteq I \wedge G' \subseteq G \wedge P' \subseteq P$). The valid business context, which we name BC, is a Cartesian product that is shown in Listing 2.

$$BC = (I' \times G' \times P') = (\{\text{Sales}, \text{Finance}\} \times \{\text{DE}, \text{GB}\} \times \{\text{P1}, \text{P4}\}) \quad (2)$$

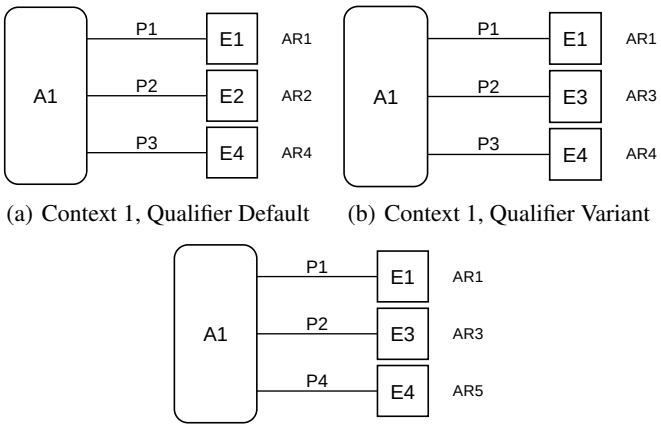
As we have now defined the logical principle behind the context-driver approach, we want to continue with describing how this approach can be utilised into business process modelling techniques.

3.3 Context-driven activity structures

Business process models contain a set of activities which are processed in a sequential order. The sequential order constitutes the backbone of a process. Having a closer look at the activities, some business process modelling notations, such as the extended Event-driven Process Chain (EPC), enable the visualisation of entities that are involved in the execution of a process. These entities can be roles that execute the activity, or roles that are responsible, or resources which are required for the execution of an activity, such as input or output data or documents.

We want to recall our running example in Figure 1. The activity "Create Invoice" is linked to one responsible role "Accounting Manager" and has several input resources, e.g. "Customer Data" or "Sales Order", and one output resource, namely "Invoice". Thus, we can see that an activity has several relations to other business artefacts which we also call entities in the following. A set of entity-relations can be different regarding the context in which an activity is represented. Our example shows that the sales department does not have the input resource *Outgoing goods*. Now, we will show how this scenario can be reflected using the context-driver principle.

The solution will be presented in an abstract way with variables which represent process artefacts, such as *activities* (*A*) and *entities* (*E*). Entities reflect basic process artefacts, such as roles, resources, data, documents, etc. *Predicates* (*P*) represent the semantical meaning, e.g. "is responsible", of the relations which are named *Activity Relations* (*AR*). Figure 3 represents a scenario of one activity which appears in three structural forms and is processed in two different process models. Although the relation-structures look different, it is always the same activity with one unambiguous semantic meaning.



(a) Context 1, Qualifier Default (b) Context 1, Qualifier Variant

(c) Context 2, Qualifier Default

Fig. 3. Abstract Context-Driven Activity Scenario

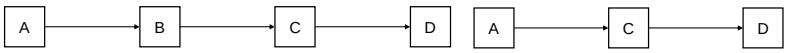
One major goal of our approach is the semantic formalisation of process model components, which should be represented with low redundancy and therefore to enable and encourage the reusability of process model knowledge. If we compare these three activity structures, we can determine relations which are used in both contexts (*AR1*), in only one context (*AR4, AR5*), and also relations which are even different within a single business context (*AR2, AR3*). The latter is labelled as *Qualifier Default* and *Qualifier Variant*. On the contrary to the context-driver approach by Stuhec [12], instances of one semantic activity can also have different relations within the same business context. For example, our process model from Figure 1 may contain the activity "Post Invoice" twice, however both activities may be linked to different roles. To overcome this issue, we introduce the concept of qualifiers, which can be seen as a kind of artificial context. This artificial context will not be visible to a modeller but it is needed in order to identify duplicate activities and relate them to the correct artefacts. One other possibility would be to create a copy of the activity and store all relations for each instance of it. However, this will lead to an uncontrolled number of instances which will make the reuse of existing relations problematic. As we have already mentioned, we only want to keep one instance of the activity and build the dependencies based on a certain context setting. Hence, a modeller has to set his context before creating and updating a process model. With this solution it is possible to define relation variations of an activity structure within a context.

3.4 Context-driven process structures

In general, a business process model is based on activities which are performed in a specific order. Within our framework a business process is linked to a *Process Structure*. A process structure contains the sequence flow of activities represented by their *Activity Structures* which were introduced in Section 3.3. Activities define the executional part of a business process. In order to present the sequential flow of a process model we introduce the concept of a *Process Step*. A process step is a special kind of wrapper object that can contain activity structures or logical coordinators, such as XOR, OR, or AND, or other relevant artefacts, such as sub-processes. In addition, a process step not only contains an activity or other entity but also contains a qualifier. The reason was discussed in the Section 3.3. An activity might have two different structures within one process. Thus, a process step contains also information about an artificial context value which is used to achieve a unique identification for such duplicate activities. Finally, process steps are linked using a flow relation from one step to another.

Although process structures are different to activity structures, the application of context awareness in process structures is also reasonable. Figure 4 depicts an example scenario with two abstract business process models in two different business contexts. The process steps of these models are marked with capital letters.

We recognise that both business process models are similar. In fact they are the same, apart from the aspect that process step *B* is omitted in business context *C2*. Without context awareness we would have to keep two separate business process model representations. However, with the usage of context awareness we are able to have one business process model representation which is able to consider differences regarding its appearance in a specific business context, which is depicted in Figure 5.



(a) Context *C1* (b) Context *C2*

Fig. 4. Abstraction of a Context-Driven Process Flow Scenario

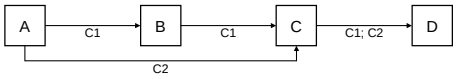


Fig. 5. Combined Abstraction of a Context-Driven Process Flow Scenario

One of our goals is to provide an intuitive way of maintaining business process artefacts. Once the process modeller enters a name into the label of a model element within his hitherto process modelling environment, a software agent validates this term against the current business context, compares the entered details with available knowledge in the repository, and finally selects the appropriate business artefact. As a result, the usage of proper business artefacts and terminology can be supported.

4 Related Work

A major facet which this work addresses is the context awareness aspect of business process models. There are already initial approaches of the consideration of contexts within business process models [13–15]. In [16], the authors concentrate on the context-relevant adjustment of configuration variants of the technical execution of business processes. The authors of [17] contemplate on the situations which affect the flow of business process models. In their work, they do not focus on structural differences and how context methods can actually change the flow of these models. Their focus is set on the formalization of these situations in the form of process contexts. However, these aspects primarily concentrate on business processes within companies and they primarily focus on their context-driven flow. The context awareness aspects which are described in this paper consider the context-driven differences in the structural appearance of business process models and focus on various levels as well as context-driven terminologies for business artefacts. Furthermore, our solution does not focus on any specific application of contexts and neither on any specific context category. It rather contemplates context-driven differences in general on an abstract level. With the development of CCTS [11], the specification already introduced the idea of context awareness among business-related data objects and data types. The concept of business contexts and core components depicts the concrete characteristic of context awareness in CCTS. Based on the specification, business contexts are classified into eight context categories. Within the work of Stuhec [10], the context aspects of CCTS were implemented into an SAP prototype called *CCTS Modeler Warp 10*[3]. Currently, the approach relates only to data

[3] *CCTS Modeler Warp 10* is a Semantic Web ontology-based data integration, modelling and mapping tool that leverages the semantics of meta data by implementing the semantic-based approach described in ISO 15000-5 Core Component Technical Specification.

modelling but we have successfully adapted those principles for business process modelling. The approach was chosen as it has been proven efficient and practical for business data modelling.

Another related research stream is the work of [18]. The authors propose an extension of EPCs, called the aggregate EPC (aEPC), which can be used to describe a set of similar processes with a single model. In contrast to this, our work not only focuses on the process flow but on all relations between business artefacts. Furthermore, the context aware process models from our framework could be used in order to create such aggregated EPCs.

5 Conclusion

In this paper, we have presented the adaptation of a context-driver principle for business process modelling approaches. We argue that we can achieve a higher consistency and reusability of process knowledge by introducing such a formalised context aware specification. Our current implementation assumes the existence of suitable context categories which potentially are created by knowledge engineers. A remaining open question is, what context categories might be appropriate for business process modelling and who will define them.

To be able to prove the conception, the semantic representation, which would be stored in the process repository, has been reproduced in a relational form. The relational data model was used for this purpose because all relations can be stored in a relational database. Furthermore, this was also advantageous for the validation of the structural relationships and dependencies, which could be proven by means of query statements. Different real-life business processes from SAP where used in order to show the various aspects of process models. The context awareness approach enables the usage of context-driven vocabulary and also the context-driven appearance of activity structures and process structures in order to consider business situation dependent variations. Test queries have emulated possible functional queries of software agents. All entities and structures could be stored and retrieved in the expected way. This part of the validation also showed that relational databases are an appropriate technology for an implementation. Currently, we are implementing the solution prototypically into a web-based business process modelling tool.

In the future, we want to investigate the impact of the context-driver principle into business process modelling activities. In order to verify the actual benefits and acceptance of such a context aware modelling system, we want to conduct a case study. Our analysis aims to reveal if the concepts support or hinder business process modelling activities. One critical aspect is the performance of our approach. Having a huge set of context values and resulting business context settings might lower the search functionality tremendously. In order to show the influence of the different context settings, we are currently preparing a test environment.

Overall, the goal is to enhance the quality of process models from the very beginning and enable a better reuse of existing process knowledge, by establishing a common understanding and terminology of business process artefacts.

References

1. van der Aalst, W.M.: Business process management demystified: A tutorial on models, systems and standards for workflow management. In: Lectures on Concurrency and Petri Nets. Volume 3098., Springer-Verlag, Berlin (2004)
2. Reijers, H.: Design and control of workflow processes : business process management for the service industry. Springer-Verlag New York, Inc., Springer (2003)
3. List, B., Korherr, B.: An evaluation of conceptual business process modelling languages. In: SAC '06: Proceedings of the 2006 ACM symposium on Applied computing, New York, NY, USA, ACM (2006) 1532–1539
4. Curtis, B., Kellner, M.I., Over, J.: Process modeling. Commun. ACM 35 (1992) 75–90
5. Rosser, B.: Taking advantage of user-friendly business process modeling. Gartner Research G00156919 (2008)
6. Blechar, M.: Magic quadrant for business process analysis tools, 2h07-1h08. Gartner Research (2007)
7. Bandara, W., Indulska, M., Sadiq, S., Chong, S.: Major issues in business process management: An expert perspective. In: in proceedings of the European Conference on Information Systems (ECIS 2007), June 7-9, St. Gallen, Switzerland, School of Information Technology and Electrical Engineering (2007)
8. Filipowska, A., Kaczmarek, M., Kowalkiewicz, M., Zhou, X., Born, M.: Procedure and guidelines for evaluation of BPM methodologies. Business Process Management Journal 15, issue 3 (2008)
9. Mendling, J., Recker, J.: Towards systematic usage of labels and icons in business process models. In: CAiSE 2008 Workshop Proceedings - Twelfth International Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD 2008). (2008)
10. Stuhec, G.: How to solve the business standards dilemma - CCTS key model concepts. Technical report, SAP Developer Network (SDN) (2006)
11. UN/CEFACT United Nations Centre for Trade Facilitation and Electronic Business: Core components technical specification - part 8 of the ebXML framework (November 2003) Version 2.01.
12. Stuhec, G., Yu, H.: Context Driven Approach. Technical report, SAP AG (December 2007)
13. Rosemann, M., Recker, J.C., Flender, C., Ansell, P.D.: Understanding context-awareness in business process design. In: In: 17th Australasian Conference on Information Systems, December 6-8, 2006, Adelaide, Australia. (2006)
14. Rosemann, M., Recker, J.: Context-aware process design: Exploring the extrinsic drivers for process flexibility. In Latour, T., Petit, M., eds.: 18th International Conference on Advanced Information Systems Engineering. Proceedings of Workshops and Doctoral Consortium., Luxembourg, Namur University Press (2006) 149–158
15. Saidani, O., Nurcan, S.: Towards context aware business process modelling. In: In: Proceedings of the 8th Workshop on Business Processes and Support Systems: Requirements for flexibility and the ways to achieve it, BPMDS'07. (2007)
16. Hallerbach, A., Bauer, T., Reichert, M.: Context-based Configuration of Process Variants. In: 3rd International Workshop on Technologies for Context-Aware Business Process Management (TCoB 2008). (2008)
17. Rosemann, M., Recker, J., Flender, C.: Contextualization of business processes. International Journal of Business Process Integration and Management (IJBPIM) 3(1) (2008) 47 – 60
18. Reijers, H.A., Mans, R.S., van der Toorn, R.A.: Improved model management with aggregated business process models. Data Knowl. Eng. 68(2) (2009) 221–243