CEUR-WS.org/Vol-2673/paperDA2.pdf

Business Process Architecture Design Based on Domain Models (Extended Abstract)

Fernanda Gonzalez-Lopez

Pontificia Universidad Católica de Valparaíso, fgonlop@uc.cl

Abstract. This thesis proposes a novel Business Process Architecture (BPA) design method for overcoming limitations of currently available alternatives. The proposal – named domain-based BPA (dBPA) method – uses domain models as a starting point for building BPA models. The evaluation of the dBPA method revealed that it was perceived as useful and likely to be used in practice.

Keywords: Business Process Architecture · Domain Model · Method.

1 Introduction [1]

The thesis focuses on Business Process Architecture (BPA) design. A BPA consists of an organized overview of business processes and their relations. An analysis of the literature of the BPA research field revealed problems regarding BPA design methods (see Sect.3) that ultimately challenge the most relevant quality metrics – and predictors of use – of BPA methods, namely, ease of use and usefulness.

The aforementioned issues led to identifying the need for a new BPA design method: the domain-based BPA (dBPA) method. The research, which followed the Design Science Research (DSR) paradigm [17], focused on the following research goals:

- **G1 (Artifact design):** Develop the dBPA method so that it reuses structured business knowledge; considers composition, specialization, trigger, and resource flow process relationships; and uses a suitable industry-standard language.
- **G2 (Instrument design):** Develop research instruments to validate the dBPA method in terms of requirement satisfaction, effects, and trade-offs.
- **G3 (Knowledge):** Assess the performance of the dBPA method and compare it to the performance of currently available BPA design methods.
- **G4 (Prediction):** Predict the usage of the dBPA method and compare it to the predicted usage of currently available BPA design methods.

The qualitative phase of the work was concerned with the following: (i) designing a conceptual framework (see G1 - G4), and (ii) developing the method itself and analyzing industry standards for the BPA models (see G1). The quantitative phase focused on validating the proposed method (see G2 - G4).

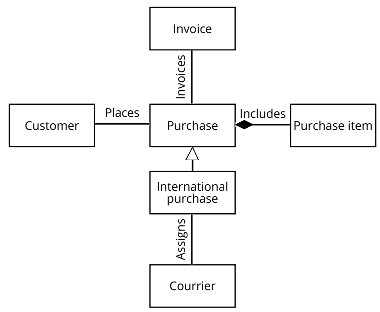
[1] Materials reported in this chapter are partially published in [1].

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

2 Models [2]

Data models represent information handled by organizations. A domain model represents the structure of a domain by showing its main concepts and how they relate to one another as classes and associations, respectively (see Fig. 1). An object lifecycle (OLC), can be built for a class to represent the dynamic aspects of data, namely states and transitions. Unlike traditional activity-centric approaches to process modeling (e.g. [10]), the dBPA method considers data as a first-class citizen, in line with the entity-centric paradigm [13]. Accordingly, it is based on structure and dynamic models of business entities (BEs), i.e. real-world concepts handled by the organization.

Fig. 1. Domain model



3 Related Work [3]

The analysis of currently available BPA design methods based on a Systematic Literature Review (SLR) following [11], revealed the following limitations:

- **Lack of structured business knowledge inputs for BPA design.** BPA design methods usually rely exclusively on unstructured inputs (e.g. domain expert knowledge). This feature challenges the ease of use of the methods.
- **Limited consideration of business process relations within BPA design.** Though relationships between business processes are key elements of BPAs, most BPA design methods support one or two types. This issue jeopardizes the completeness of the resulting BPA models.
- **Restricted use of industry-standard languages in BPA models.** The use of industry standards is very low among BPA design methods. This adds difficulty to the understandability of the resulting BPA models.

4 Language

The selection of a suitable architecture description language for the dBPA method was based on requirements regarding abstract syntax and cognitive effectiveness of Business Process Model and Notation (BPMN) [14] and ArchiMate [16]. The assessment revealed that only ArchiMate complied with abstract syntax requirements. However, to improve cognitive effectiveness, its viewpoint mechanism was used for defining the BPA viewpoint, exemplified in the BPA model in Fig. 2.

[2] Materials reported in this chapter are partially published in [7, 3].
[3] Materials reported in this chapter are partially published in [5].

5 Foundations [4]

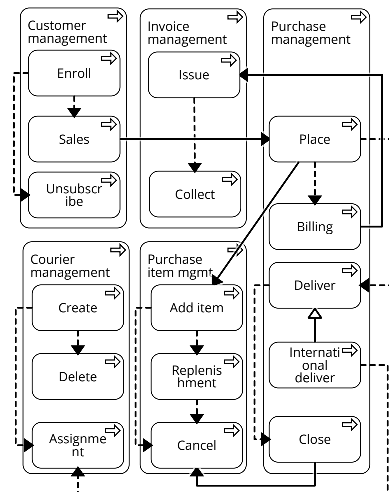
The dBPA method is based on a number of foundations. First, regarding the identification of business entities and relationships:

- Each domain model class defines, at most, one business entity.
- Each domain model association defines one or more relationships between the OLCs of the partaking classes.

Second, integration of structure and dynamic data models involves specializing the domain model classes with dynamic hierarchies such that:

- A dynamic hierarchy is built by specializing the original class with sub-classes that hold a bi-univocal relationship with the OLC states.
- The properties of the hierarchy are defined in a way that they are consistent with the organization of OLC states.

Fig. 2. BPA model



Third, formal definitions used for the method include the concepts of: association a, BPA model Θ, business entity b, business process φ, class c, data dictionary dd, domain model w, and object lifecycle l.

6 Method [5]

The dBPA method has six steps that are – generally speaking – sequential:

1. **Prepare domain model.** Some transformations are applied to the original domain model to produce a semantically-equivalent domain model w where all associations are directed and power types are replaced by hierarchies.
2. **Identify business entities.** The set of business entities B corresponds to the set of all classes $c \in w$ that are not sub-classes.
3. **Identify states of domain model classes.** The set of states for each business entity $b \in B$ is identified by using two sources of information: the domain model w and its complementary information (e.g. its data dictionary dd), and domain expert knowledge.

[4] Materials reported in this chapter are partially published in [2].
[5] Materials reported in this chapter are partially published in [4].

4. **Build OLC for each business entity.** The OLC l for each business entity $b \in B$ is built by using domain expert knowledge. In every OLC l, a transition corresponds to one business process φ which changes the state of the respective business entity. Also, each association a between domain model classes is mapped into relations between the corresponding OLCs.
5. **Build BPA model.** The BPA model Θ is built by mapping information in each OLC l and in the domain model w.
6. **Improve domain model expressiveness.** The domain model w is transformed to a semantically equivalent one by mapping OLC states to subclasses of dynamic hierarchies.

7 Evaluation [6]

The conceptual evaluation of the proposed method was done using a novel framework based on the works by [9, 15]. This highlighted some differentiating aspects of the proposal. A user study was conducted for the empirical evaluation of the dBPA method based on the Method Evaluation Model (MEM) [12]. The study assessed the use of the dBPA method and another BPA design method (i.e. BPTrends Associates [10]) in terms of ease of use, usability, and intention to use with promising results in favor of the proposal.

8 Conclusions

G1 (Artifact design): The dBPA method was designed for overcoming issues of currently available BPA design methods. The reuse of structured business knowledge was implemented as using domain models as an input for the method. The method also considers four types of process relations that, though mentioned in the literature, are rarely considered together in BPA design methods. Including them allows building more complete BPA models. The use of an industry-standard language was done by proposing an ArchiMate viewpoint that would improve cognitive effectiveness of the resulting BPA models. The proposed method has some limitations that could be addressed in future research: (i) it works with a simplified version of the domain model; (ii) it assumes that the granularity of all OLC transitions is such that it can constitute one business process; and (iii) it is domain model dependent. Future works include: partial automation of the method, and development of tools for BPA analysis.

G2 (Instrument design): Two instruments were tailored for evaluation of the proposed method. The framework for qualitative analysis was found useful for the description, evaluation, and comparison of (other) BPA design methods at a conceptual level. Future lines of work might include validating this framework with a panel of experts. An instantiation of the MEM was successfully used for quantitative analysis. Future research could include improving some issues

[6] Materials reported in this chapter are partially published in [6, 8].

inherited from the MEM, as well as some issues of the instantiation. Also, the use of the MEM could be further developed and tested.

G3 (Knowledge): The evaluation provided significant evidence that the proposal was perceived as useful and not easy to use. The user study also provided preliminary evidence that the proposed method is an improvement regarding ease of use, usefulness, and intention to use. These results are consistent with the main design decisions of the proposed method. In the end, however, further research is needed to gather significant evidence.

G4 (Prediction): In terms of adoption in practice, the study provided preliminary evidence in favor of the proposal. Again, future research is needed in this regard. Also, the study provided significant evidence stating that usefulness seems to be the main predictor of adoption in practice followed by ease of use.

References

1. Gonzalez-Lopez, F.: Research proposal for business process architecture from domain models. In: Doctoral Consortium, BPM 2016 (2016)
2. Gonzalez-Lopez, F., Bustos, G.: Integration of entity-centric models for business processes. In: 2nd Workshop STI, Chillán (2015)
3. Gonzalez-Lopez, F., Bustos, G.: Business process architecture in the context of enterprise architecture: a survey. In: 8th ICPR – Americas 2016, Valparaíso (2016)
4. Gonzalez-Lopez, F., Bustos, G.: Business process architecture baselines from domain models. In: BPM 2016, LNBIP 281. pp. 118—130. Springer (2017)
5. Gonzalez-Lopez, F., Bustos, G.: Business process architecture design methodologies – a literature review. Bus. Process Manag. J. **25**(6), 1317–1334 (2019)
6. Gonzalez-Lopez, F., Bustos, G.: Evaluating methodologies for business process architecture design – a pilot study. In: ZEUS 2019, CEUR 2339 (2019)
7. Gonzalez-Lopez, F., Bustos, G.: Integration of business process architectures within enterprise architecture approaches: A literature review. Eng. Manag. J. **31**(2), 127–140 (2019)
8. Gonzalez-Lopez, F., Pufahl, L.: A landscape for case models. In: BPMDS 2019, LNBIP 352. pp. 87–102. Springer (2019)
9. Green, S., Ould, M.A.: A framework for classifying and evaluating process architecture methods. Softw. Process Improv. Pract. **10**(4), 415–425 (2005)
10. Harmon, P.: Business process change: a business process management guide for managers and process professionals. Elsevier (2014)
11. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering Version 2.3 (2007)
12. Moody, D.L.: The method evaluation model: A theoretical model for validating information systems design methods. In: 11th ECIS. pp. 1327–1336 (2003)
13. Nigam, A., Caswell, N.S.: Business artifacts: An approach to operational specification. IBM Syst. J. **42**(3), 428–445 (2003)
14. OMG: Business Process Model and Notation (BPMN): Version 2.0. (2011)
15. Pesantes, M., Mitre, H.A., Lemus, C.: Evaluation of process architecture design methods. In: 13th ICCSA, Ho Chi Minh City. pp. 144–159 (2013)
16. The Open Group: ArchiMate® 3.0.1 specification (2017)
17. Wieringa, R.: Design science methodology for information systems and software engineering. Springer (2014)