On the Contribution of Reference Modeling to e-Business Standardization – How to Apply Design Techniques of Reference Modeling to UN/CEFACT's Modeling Methodology

Birgit Hofreiter and Jan vom Brocke

University of Liechtenstein,
Martin Hilti Chair of Business Process Management,
Institute of Information Systems, Principality of Liechtenstein
{birgit.hofreiter, jan.vom.brocke@hochschule.li}

Abstract. Reference Modeling has evolved as a strong discipline especially driven by the German speaking community. Great achievements have been made in finding ways to leverage the potentials of reuse in business modeling. However, the perception of reference modeling as such is still limited to a rather small group of scholars. This is surprising as the phenomenon of "re-use" is very much in the center of various current topics in the international information systems and business process management discipline. In this paper, we set out exploring the contribution of findings in the field of reference modeling for business standardization. We show that in particular design techniques for reference modeling perfectly apply to solve conflicts of globalization and localization in business standards. As an example, we study the UMM as a specific standard developed within the United Nations CEFACT group.

Keywords: Reference Modeling, Re-use, Construction Concepts, Standardization, UN/CEFACT Modeling Methodology (UMM).

1 Introduction

With its roots in the early 1990s reference modeling has predominantly been driven by the German speaking IS community. The essential idea behind it may be seen in "not wanting to re-invent the wheel" whenever engaging in a new modeling process. It is intended to provide some kind of "reference" to start with in order to increase both the efficiency and effectiveness of modeling processes [1, 2, 4, 8, 10]. However, despite of this kind of common understanding, we can observe quite an intensive discussion on the very characteristics of reference models and reference modeling respectively. (For an overview see e. g. [3, 13]).

Considering this discussion, we hold that putting the concept of re-use at the core of reference modeling enables the use of synergies by combining them with corresponding fields of research in IS. Accordingly, reference models are referred to as

special information models that serve to be reused in the design process of other information models [19, 20] as outlined in Fig. 1.

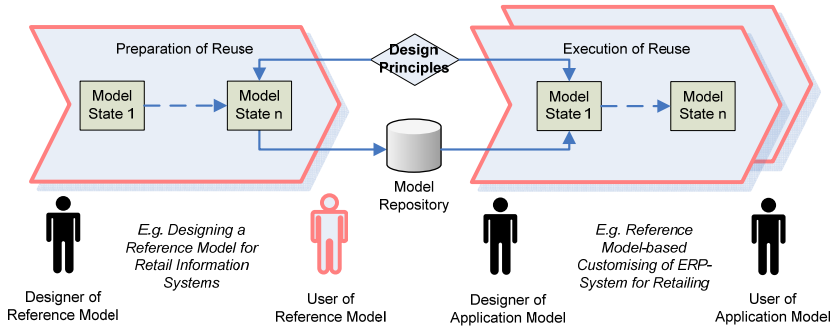

Fig. 1. Reuse-oriented Concept of Reference Modeling [20]

In this paper we study the potential of reference modeling and its phenomenon of "re-use" in the field of business standardization. In particular we focus on the standardization of inter-organizational business processes. In this case business standardization has to focus on the business activities that require communication between the partners. This results in standardized interface processes that allow an easy integration of new business partners or the easy creation of virtual enterprises. Instead of describing these inter-organizational business processes for a specific platform like Web Services or ebXML we prefer to specify platform independent models that are later transformed to platform specific processes.

Such an approach is envisioned by the United Nation's Centre for Trade Facilitation and e-business (UN/CEFACT) by the UN/CEFACT Modeling Methodology (UMM) [17], which we have co-authored. We have defined the UMM as a UML profile specifying a set of stereotypes, tagged values and constraints that put UML in a very strict corset resulting in well-defined artifacts. These artifacts may be transformed to BPEL [5, 6] or BPSS [7] in a next step.

In this paper we use the UMM as our modeling language of choice. In order to guarantee acceptance of the UMM, it must be both effective and easy to understand for business process modelers and software architects. Re-use is vital for efficiently managing the complexity of standard models on different levels of application. In particular, techniques of model re-use that have been subject to intensive research in reference modeling may well be applicable here. Thus, we set out to demonstrate and further analyze the synergies between reference modeling and e-business standardization by means of UMM. We introduce some well accepted design techniques in reference modeling in section 2. In section 3 we analyze the potential of these design techniques when creating a UMM model. The conclusion in section 4 summarizes the benefits that reference modeling offers to UMM.

2 An overview of Design Techniques in Reference Modeling

In order to facilitate re-use, various design techniques are subject to research in reference modeling (see Fig. 2). These techniques provide rules describing the way in which the content of one model is reused in constructing another model. The rules describe ways of taking over contents as well as of adapting and extending it in the resulting model. With each design techniques, specific sets of rules are differentiated.

In former studies on reference modeling, the design of configurative reference models was focused on, in order to support the derivation of multiple variants of a reference model for a certain application [1, 10, 13]. This work intends to encounter all relevant variants of prospective applications during build-time of the model in order to facilitate adaptability by means of choices [18]. Considering the variety of requirements to be faced in today's business engineering, the design techniques of configuration illuminates specific limitations. In particular, it is increasingly hard to take into account the various requirements that may be relevant, and to incorporate them in the reference model.

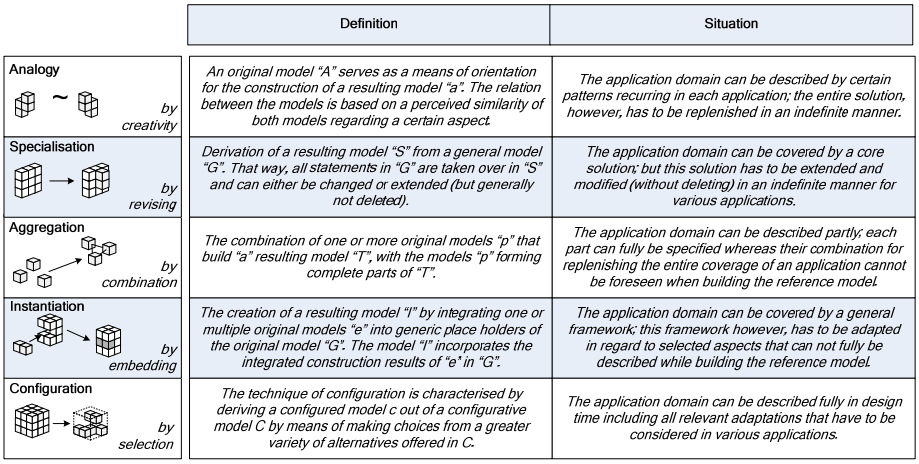

Fig. 2. Design Techniques of Reference Modeling according to [20]

Hence, supplementary design techniques have been developed in order to enlarge the "tool-kit" of reference modeling. Particularly referring to software-engineering [9, 12] the techniques aggregation, specialization, instantiation, and analogy are presented in this paper [19, 20]. According to instantiation, general aspects of a domain are designed as a framework providing generic placeholders for plugging in models considering special requirements of an application. Specialization enables the takeover of the entire contents of a general model into a specific model allowing for modification and individual extending. Aggregation enables the take-over of contents delivered by various part models that are composed in and extended according to special requirements of application. Analogy, finally, employs seemingly similar solutions in a creative way in order to tackle new problems.

Studies show that each of the technique has different economic effects in regard to context situation of the modeling process. Looking at the derivation of costs, it becomes apparent that configuration and analogy form two opposite techniques for reference modeling. Whereas configuration implies that most of the work on reusing content is done by building the reference model, this work is left for the application using analogy. Consequently, configuration comes along with relatively high costs for building the reference model on the one hand, but with low costs for applying it on the other. The principle of analogy, on the contrary, causes a minimum cost for building the model and yet a maximum for applying it. The other techniques gradually lie in between the two first ones. Applying instantiation, prospective applications do not need specification entirely while building the reference model. Saying that, a minimum of certain generic aspects have to be identified and specified for embedding special solutions. In aggregation, only certain parts of the application domain definitively require description that may be combined and extended in various ways. However, modifications of each part model to be aggregated are not provided. This is possible thanks to the principle of specialization giving way to rather flexible modifications without eliminating parts of the reference model. With the techniques of analogy, finally, unlimited ways of adapting the content are given.

Assumingly, these techniques may well be used in order to manage different variants of a standard within a global business standardization initiative. This will be further analyzed in the next chapter.

3 Applying Design Techniques to the UMM

The UMM is a methodology that guides the modeler through a development process of well defined tasks in order to create a UMM compliant model. The methodology itself and the resulting artifacts are structured in three main phases and views, correspondingly. The first phase gathers domain knowledge and existing process knowledge of the business domain under consideration. The resulting artifacts are captured in the business requirements view and its subviews. Based on these requirements, the second phase is used to describe the resulting inter-organizational business processes as a choreography from an observer's perspective. Consequently, the artifacts are kept in the business choreography view, which comprises the subviews business transaction view, business collaboration view and business realization view. The last phase is used to model the business document types which are used in the exchanges between business partners. Its artifacts are part of the business information view.

Due to space limitations we are not able to demonstrate how the design techniques of reference modeling apply to all of the UMM artifacts. Thus, we limit ourselves to the task of specifying an agreed choreography between the business partners, which is also the core focus of the UMM. Consequently, we focus on the artifacts of the business choreography view. In the following two subsections we analyze the potential of reference modeling for the subviews business transaction view and business collaboration view.

3.1 Business transaction view

The basic building blocks of a UMM choreography are business transactions. The goal of a business transaction is synchronizing the business entity states between two parties. Synchronization of states is either required in a uni-directional or in a bi-directional way. In the former case, the initiator of the business transaction informs the other party about an already irreversible state change the other party has to accept - e.g., the notification of shipment. It follows, that responding in such a scenario is neither required nor reasonable. In the latter case, the initiating party sets a business entity to an interim state and the responding party decides about its final state - consider a *request for quote* that the responder might either refuse or accept when returning a *quote*.

The synchronization takes place by exchanging business information. According to the definitions above, an exchange always takes place between exactly two parties. It is either a uni-directional exchange or a bi-directional exchange including a response. Due to this strict setting, business transactions no matter what business goal they try to achieve always follow the same basic structure. Thus, the design technique of analogy is appropriate for constructing business transactions.

Exploring Analogy

The design technique of analogy is realized in UMM by providing a basic pattern for UMM transactions which has to be used when creating a new business transaction. An activity diagram stereotyped as business transaction always includes two partitions, one for each participating role. Each partition includes exactly one activity, the requesting business activity and the responding business activity, correspondingly. The activity diagram starts off with the requesting business activity.

Exactly one requesting business document is exchanged. This is notated by an object flow from the output pin of the requesting business document to the input pin of the responding business activity. In case of a one-way business transaction, no document is returned, i.e. no object flows are modeled in the reverse direction. In case of a two-way transaction at least one document is exchanged from an output pin of the responding business transaction to an input pin of the requesting business transaction. If alternative responses are allowed, additional object flows are added to the reverse direction.

A business transaction must always lead to a well-defined business entity state. The requestor is able to recognize the resulting state by receiving the responding document or by an acknowledgment in case of a one way transaction. As a consequence, the resulting business entity states are model by transitions from the requesting business activity to the end states. There must be at least one business entity state. Further alternative business entity states may be added, but must be well distinguished by mutually exclusive transition guards.

This analogy has to be followed whenever a new business transaction is created, no matter whether *quote requests*, *purchase orders*, *notification of shipment*, etc., are modeled. Fig. 3. depicts an example of a business transaction for *quote requests*.

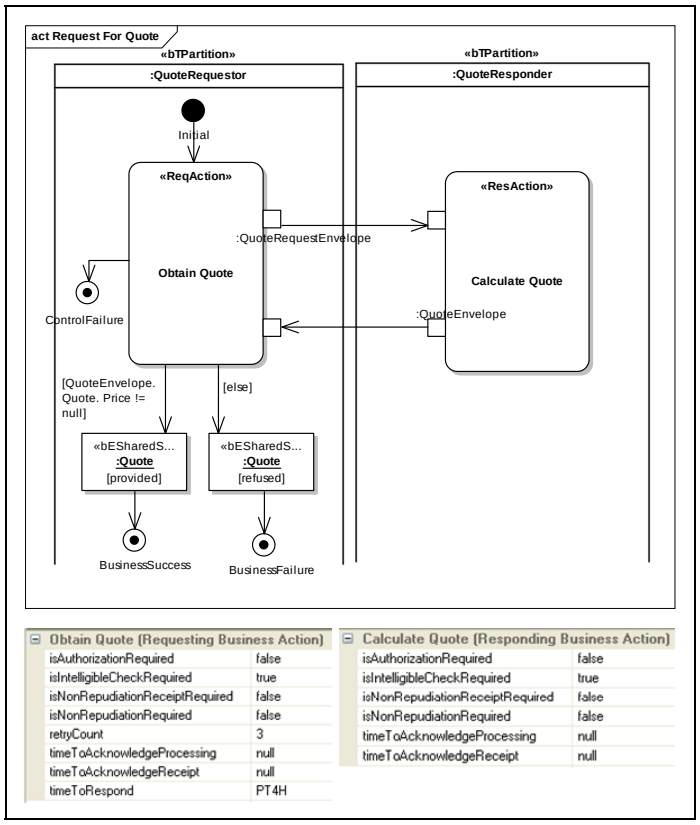

Fig. 3. Business Transaction: Request For Quote

Since we talk about standardized business transactions, a model library should only include one business transaction for a specific business goal, i.e. there should be only one business transaction for *quote requests*. However, the UMM concept of a business transaction comes also with a set of tagged values. The requesting/responding business activities are described by values for *is authorization required*, *is non-repudiation required*, *time to perform*, *time to acknowledge receipt*, and *time to respond*. The values for *is non-repudiation of receipt required* and for *retry count* are only defined for the requesting business activity. Most of these attributes are self-explanatory. An *acknowledgment of receipt* is usually sent after grammar validation, sequence validation, and schema validation. However, if the *is intelligible check required* flag is set to false, the acknowledgment is sent immediately after receipt without any validation. An acknowledgment of processing is sent after validating the content against additional rules to ensure that the content is processable by the target application. *Retry count* is the number of retries in case of control failures.

In practice, it is realistic that an existing standardized business transaction, e.g. for *quote requests*, fits to needs of a specific business partnership. However, it is rather unlikely that the values for the tagged values are acceptable for all business partner-

ships using this business transaction. For example, one partnership expects the quote to be returned (time to respond) within four hours and the other one within a day. To overcome this problem of different tagged values the design technique of configuration may be used.

Exploring Configuration

The design technique of configuration is realized for UMM business transactions by providing a master business transaction for each business goal, e.g. for a request for quote in the model library. This master business transaction does not specify any predefined values for the tagged values. If this master business transaction is used in a specific scenario for a to-be created partnership, a new business transaction is created based on the master business transaction, but which sets the tagged values. Thereby, it is guaranteed that standardized process and document interfaces are re-used, but with different timings and security parameters.

3.2 Business collaboration view

The main artifact of the business collaboration view is the business collaboration protocol. Whereas the business transaction models a simple inter-organizational business process exchanging one document and optionally returning another one, the business collaboration protocol models complex business collaboration usually including many activities between the involved parties. For the purpose of re-use, a business collaboration protocol uses the design technique of aggregation.

Exploring Aggregation

It follows that a business collaboration protocol is built by aggregating existing models, either existing business transactions or other existing business collaboration protocols which are nested within the to-be created business collaboration protocol. Accordingly, a business collaboration protocol includes two kinds of actions. A business transaction action calls the sub-process of an existing business transaction, whereas a business collaboration action calls the sub-process of another existing business collaboration protocol. The call of sub-processes is noted in UML in general by the rake symbol in the action.

In Fig. 4 we show the example of a simple business collaboration protocol for *order from quote*. The resulting activity graph is built by two business transaction actions. The first one calls the sub-process of the business transaction *request for quote* which we have outlined in Fig. 3. The second one is a call to the business transaction *place order* which we omit to depict. The transitions in the business collaboration protocol are guarded by entity states. Each business entity state of a business transaction must lead to a transition starting from the business transaction action in the business collaboration protocol For example, the business transaction *request for quote* sets the state of the business entity quote either to provided or refused. Accordingly, the business transaction action *request for quote* has two outgoing transitions: If the quote is provided the business collaboration protocol continues with *place order* and if the quote is refused the business collaboration terminates.

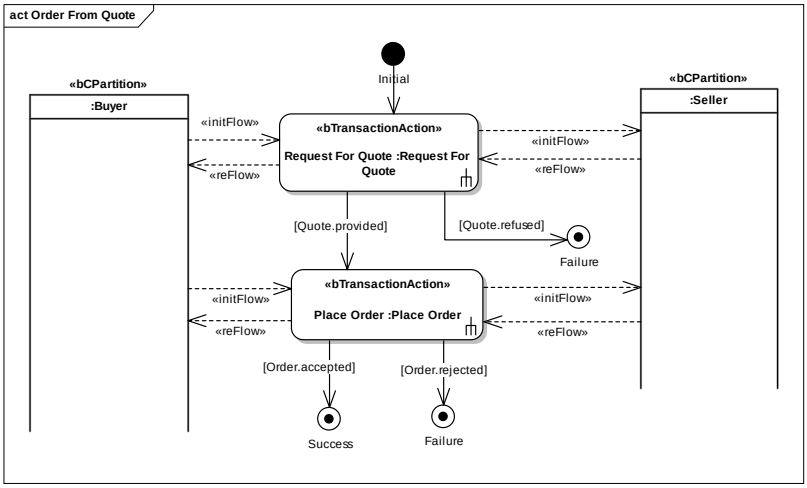

Fig. 4. Fig. 4: Business Collaboration Protocol: Order From Quote

In order to foster the re-use of an existing business transaction in different business collaboration protocols, it is not desired to mandate that the roles on the two different levels of abstraction must be identical. In other words, one and the same business transaction may be re-used in different business collaboration protocols each performed by a different set of partners. In order to cope with this requirement UMM has to consider the design technique of specialization.

Exploring Specialization

In order to allow the specialization of roles, one has to use the concept of partitions in the business collaboration protocols. However, the partitions are not used to cover the actions – which are by definition inter-organizational ones. The partitions serve the purpose of role mapping. This is rather unusual in UML, but this concept is adopted from BPMN [11]. A business transaction action is linked with the partitions of the involved parties by the concept of init flow and in case of an underlying two way transaction also by a re-flow. Thereby, the role of the partition that is the source of an init flow performs the initiating role of the called business transaction. Vice versa, the role of the partition that is the target of an init flow performs the responding role of the called business transaction.

For a better understanding, consider our example. The business transaction *request for quote* is performed between the roles *quote requestor* and *quote responder*. The business collaboration protocol defines the collaboration between a *buyer* and a *seller*. In case of the business transaction action *request for quote*, the init flow starts from the buyer's partition and leads to the seller's partition. Consequently, the buyer plays the *quote requestor* (initiating role) and the seller acts as *quote responder* (responding role) in the underlying business transaction.

3.3 Extension to the business collaboration view

All the design techniques we have described so far do not require any changes to the UMM meta model. In order to develop a UMM compliant business collaboration view it is required that the call behavior of all business transaction actions and business collaboration actions is exactly defined. There exists no flexibility in calling the underlying business transaction or nested business collaboration protocol, respectively. However, when creating a reference model for a business collaboration protocol it may be desired that this call behavior is not exactly defined in order to reach some flexibility for variants and at the same time still guaranteeing a common base.

For this purpose the UMM meta model has to be extended in order to provide some placeholders in the business collaboration protocol. The placeholders may be used in reference models where the calling behavior is not exactly known at design time of the reference model. The placeholders have to be replaced later on by business transaction actions and business collaboration models when creating a specific business collaboration protocol. For this replacement on may use the design techniques of instantiation or of configuration.

Exploring Instantiation

Accordingly, the concept of instantiation is used when the structure of a called business transaction or nested business collaboration protocol is not known at all at the design time of the reference model. Whereas the concept of configuration is applied in case different alternative already existing business transactions or nested business collaboration protocols may be used. Evidently, we need new UMM stereotypes to cope with the two kinds of placeholders. In analogy, to the design technique we call these stereotypes *instantiation action* and *configuration action*. Since a configuration action must later on be replaced one out of a set of alternative business transactions / nested business collaboration protocols, it is required that UML dependencies are established between the configuration action and the members of this set.

In order to demonstrate the need for placeholders in UMM reference models we continue our *order from quotes* example which finally results in the business collaboration protocol of Fig. 5. At the design time of the reference model is known that after an order is accepted, the process continues with tracking and tracing the shipment of the order as well with the billing, which may occur in parallel. However, it is not yet known how tracking and tracing as well as billing are realized exactly. In case of billing one may consider two alternatives, either classical invoicing or the self-billing approach. Thus, billing is modeled in the business collaboration protocol of Fig. 5 as a configuration action. The dependencies of this configuration action to the two alternative business transactions must be conceptually defined, but are not part of the business collaboration protocol. It should be noted that classical invoicing is usually initiated by the seller, whereas self-billing is initiated by the buyer. Thus, it does not make sense to specify init flows and re-flows in the reference model. Since the structure of tracking and tracing is not defined in any ways, i.e. no pre-defined alternatives exist, it is modeled by an instantiation action.

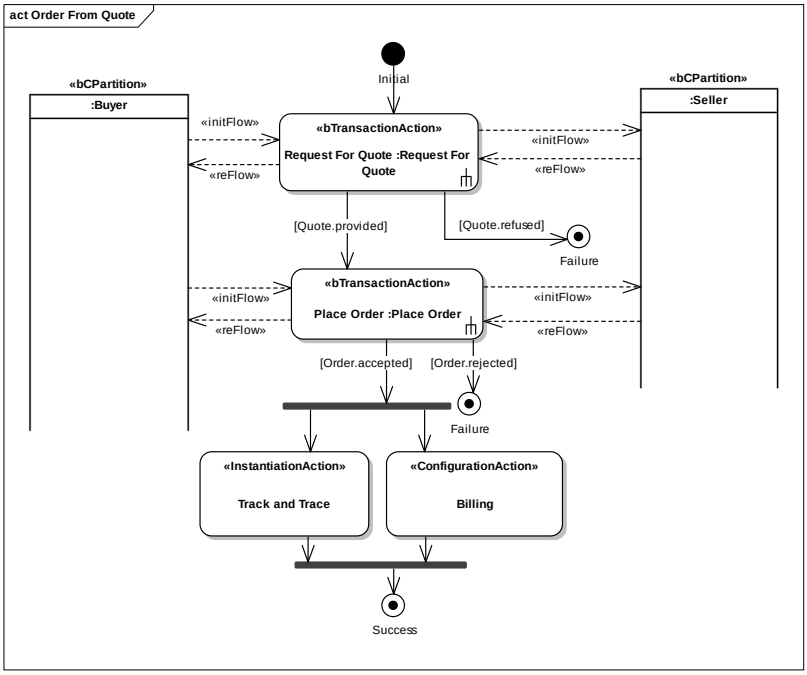

Fig. 5. Business Collaboration Protocol: Order From Quote (incl. new stereotypes)

4 Conclusion

When developing a new model one may start from scratch, or one may have a look on an already existing model that was developed in a "similar" context to speed up the development process and to leverage quality. Reference modeling is an IS field that intends at providing consistent design techniques for developing reference models and customizing them when creating new models. In this paper we evaluated how the standardization of inter-organizational business process models may benefit from the concepts of reference modeling. These concepts seem to be of particular interest since it is rather unlikely that all companies as well as public administrations will be able to work with a single standardized process. It is much more likely that different variants of the same base process are accepted. For example, there does not exist a one-for-all procurement process that serves the needs of all companies and agencies all over the world in any industry sector. However, it is expected that many different inter-organizational procurement processes share a common ground that may be modeled in a reference model.

For our evaluation we use the UN/CEFACT modeling methodology (UMM) which is a common approach to model inter-organizational business processes from an observer's perspective. We investigated how the five design techniques of reference modeling – configuration, instantiation, specialization, aggregation, and analogy –

may be adopted when developing UMM models. In particular, we demonstrated the use of these techniques for the business choreography view which comprises business transactions and business collaboration protocols. It became evident that UMM business transactions are already based on the concept of analogy, because all UMM transactions are analogous to each other by being based on a similar transaction pattern. In addition, business transactions will benefit from the concept of configuration of their tagged values for timing and security parameters, since these may differ for different variants of the same business transaction. This concept is not yet used in the development of UMM models, but the UMM meta model is capable of handling these configurations - it requires only an adaptation in the methodology for creating business transactions.

UMM business collaboration protocols are indeed based on the concept of aggregation, because they are built by aggregating business transactions and nested business collaboration protocols. Furthermore, UMM supports the concept of specialization for the roles participating in a business collaboration protocol with respect to the roles in the underlying business transactions. Additionally, business collaboration protocols may profit from reference modeling if the current UMM meta model is going to be extended. Currently, the strict aggregation mechanism requires that all the aggregated business transactions and business collaboration protocols are known when constructing a business collaboration protocol. However, when designing a reference model of a business collaboration protocol they may not be known. Accordingly, UMM may be extended by allowing actions in a business collaboration protocol that are instantiated later on for a specific model or that provide a set of alternatives that are configured later on in a specific model.

One may argue that we investigated only in the artifacts of UMM's business choreography view. However, this view covers all important aspects of an inter-organizational process choreography. The business requirements view, which precedes the business choreography view, targets at capturing the requirements of to-be constructed collaboration. Thus, this view is always context specific and will not benefit much from more general, context-independent reference model. In contrary, the business information view models the document types being exchanged that may differ for different variants of business partnerships. Thus, this view may benefit as well from reference modeling. Whereas this paper concentrates on the process, future work will address the structural aspects of the business documents.

References

1. Becker, J., Delfmann, P., Dreiling, A., Knackstedt, R., Kuropka, D.: Configurative Process Modeling - Outlining an Approach to Increased Business Process Model Usability. In: Information Resources Management Association Conference. New Orleans (2004)
2. Fettke, P., Loos, P.: Classification of reference models - a methodology and its application. In: Information Systems and e-Business Management, vol. 1(1). Springer, Heidelberg, pp. 35-53 (2003)
3. Fettke, P., Loos, P. (eds.): Perspectives on Reference Modeling. In: Reference Modeling for Business Systems Analysis. Hershey, PA, USA, pp. 1-20 (2007)

4. Frank, U.: Evaluation of Reference Models. In: Fettke, P., Loos, P. (eds.) Reference Modeling for Business Systems Analysis. Hershey, PA, USA, pp. 118-140 (2007)
5. Hofreiter, B., Huemer, C.: Transforming UMM Business Collaboration Models to BPEL. In: OTM Workshops. LNCS, vol. 3292. Springer, Heidelberg, pp. 507-519 (2004)
6. Hofreiter, B., Huemer, C., Liegl, P., Schuster, S., Zapletal, M.: Deriving executable BPEL from UMM Business Transactions. In: International Conference on Services Computing. IEEE Computer Society, Salt Lake City, Utah, USA (2007)
7. Huemer, C., Kim, J.: From an ebXML BPSS choreography to a BPEL-based implementation. In: ACM SIGecom Exchanges, vol. 5(2), ACM, pp. 1-11 (2004)
8. Huschens, J., Rumpold-Preining, M.: IBM Insurance Application Architecture (IAA). An Overview of the Insurance Business Architecture. In: Bernus, P., Mertins, K., Schmidt. G. (eds.) Handbook on Architectures of Information Systems, 2nd ed. Springer-Verlag, Berlin et al., pp. 669-692 (2006)
9. Karhinen, A., Ran, A., Tallgren, T.: Configuring designs for reuse. In: 19th international conference on Software engineering. Boston, Massachusetts, United States (1997)
10. Meinhardt, S., Popp, K.: Configuring Business Application Systems. In: Bernus, P., Mertins, K., Schmidt, G. (eds.) Handbook on Architectures of Information Systems, 2nd ed. Springer-Verlag, Berlin et al., pp. 705-721 (2006)
11. OMG - Object Management Group: Business Process Modeling Notation Specification 1.0 (2006)
12. Peterson, A.S.: Coming to terms with software reuse terminology: a model-based approach. In: SIGSOFT Software Engineering Notes, vol. 16(2), pp. 45-51 (1991)
13. Recker, J., Rosemann, M., van der Aalst, W.M.P., Mendling, J.: On the Syntax of Reference Model Configuration. In: First International Workshop on Business Process Reference Models (BPRM´05). Nancy, France (2005)
14. Scheer, A.-W.: Business Process Engineering: Reference Models for Industrial Enterprises. Springer-Verlag, New York, Inc (1994)
15. Scheer, A.-W., Nüttgens, M.: ARIS Architecture and Reference Models for Business Process Management. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management - Models, Techniques, and Empirical Studies. Berlin et al., pp. 376 389 (2000).
16. Thomas, O.: Understanding the Term Reference Model in Information System Research. In: First International Workshop on Business Process Reference Models (BPRM´05). Nancy, France (2005)
17. UN/CEFACT: UN/CEFACT's Modeling Methodology (UMM), UMM Meta Model - Foundation Module. Technical Specification – Candidate for V2.0, http://www.untmg.org/wp-content/uploads/2009/01/specification_umm_foundation_module_v20_implementationdraft_20090130.pdf (2006)
18. van der Aalst, W.M.P., Dreiling, A., Gottschalk, F., Rosemann, M., Jansen-Vullers, M.H.: Configurable Process Models as a Basis for Reference Modeling. In: First International Workshop on Business Process Reference Models (BPRM´05). Nancy, France (2005)
19. vom Brocke, J.: Reference Modeling, Towards Collaborative Arrangements of Design Processes (in German). Berlin (2003)
20. vom Brocke, J.: Design Principles for Reference Modeling. Reusing Information Models by Means of Aggregation, Specialisation, Instantiation, and Analogy. In: Fettke, P., Loos, P. (eds.) Reference Modeling for Business Systems Analysis. Hershey, PA, USA (2007)