

# A Semantically-Enhanced Modelling Environment for Business Process as a Service

Knut Hinkelmann, Sabrina Kurjakovic, Benjamin Lammel, Emanuele Laurenzi  
FHNW University of Applied Sciences and Arts  
Northwestern Switzerland  
Olten, Switzerland  
<firstname>.<lastname>@fhnw.ch

Robert Woitsch  
BOC Asset Management  
Vienna, Austria  
robert.woitsch@boc-eu.com

**Abstract**— In this paper we present a hybrid modeling approach which supports the continuous alignment of business and IT in the cloud. Business Process as a Service provides the end-to-end cloud support for business processes instead of single applications. A graphical modelling environment allows non-technical modelers to design business processes and to specify requirements in human-interpretable way. Via semantic lifting, the graphical models can be annotated with classes and values from an enterprise ontology. The BPaaS Ontology contains the relevant classes for the smart Business and IT-Cloud alignment. It supports the modeler in using a standard terminology and thus ensures consistent modeling.

**Keywords**—semantic lifting, enterprise modelling, business process as a service, cloud computing

## I. INTRODUCTION

It is a challenge for today's enterprises to *continuously align business and IT in a rapidly changing environment*. According to Gartner [15] enterprises are facing a new era of enterprise IT, which is "characterized by deep innovation beyond process optimization, exploitation of a broader universe of digital technology and information, more-integrated business and IT innovation, and a need for much faster and more agile capability".

However, due to missing technical expertise, many enterprises, in particular small and medium enterprises (SMEs), have difficulties to deeply integrate business and IT and thus to gain appropriate benefit from digitalisation. To overcome these difficulties, a hybrid modelling approach has been developed which supports both *machine interpretation* and *human interpretation* of enterprise models (see Figure 1). The approach builds on the principles of model-driven enterprise engineering [34] and has been used for the alignment of business and IT in the cloud.

- A modeling environment provides graphical modelling tools for designing domain specific business processes, specifying requirements and describing properties of workflows and cloud services. These graphical models can easily be created and understood by humans.

- To identify appropriate cloud services and to determine needs for adaptations the information about the business processes, the service requirements and the workflows are represented in machine-interpretable format [19]. Semantic lifting makes the semantics of graphical models explicit [20] such that the analysis, adaptation and evaluation of models can be done by a machine. The semantics of the models is specified by an ontology.

The concept of business process as a service (BPaaS) provides the flexibility of mixing different delivery models and focusing on the end-to-end business processes instead of single applications [28]. Alignment between the business and IT level can be achieved by mapping business-oriented process models to technical models that can drive the allocation and execution of business processes in the cloud.

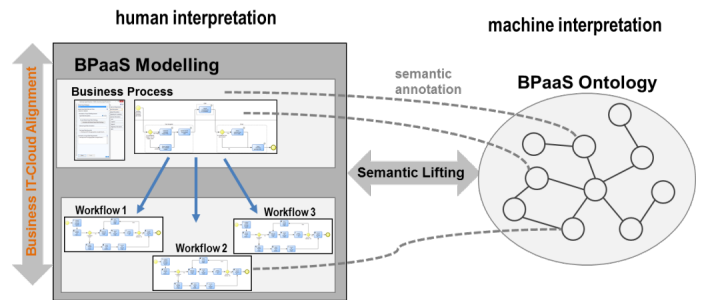


Figure 1. Hybrid Modelling with Semantic Lifting

After the literature review (Section II) we give an overview of the BPaaS Design Environment in Section III. In Section IV we describe the research methodology. The *human-interpretable, graphical* modelling is presented in Section V. The machine-interpretation is supported by the BPaaS Ontology, which is described in Section VI. The integration of these two interpretation types is achieved by the semantical lifting of the graphical models (see Section VII).

## II. LITERATURE REVIEW

BPaaS represents an initial field of research. Most of the research work proposed focuses on how to define BPaaS and the respective candidate architectures to realise it [2]. Some work has concentrated on dealing with security aspects (e.g., anonymization-based protocols for BPaaS fragments [4]). Initial work has been conducted on how elasticity can be realised for BPaaS through a specific formal model and a respective elasticity framework [23].

Modelling the business processes, workflows and services in CloudSocket is part of enterprise modelling - the description and definition of the processes, structure, information and resources of an enterprise. According to Fox and Gruninger [14] an enterprise model must supply the information and knowledge necessary to support the operations of the enterprise. Enterprise modelling techniques are developed in several fields such as business process modelling, information modelling, systems modelling, and enterprise architecture.

OMG has developed several specialized modelling languages for enterprise modelling, for example Business Process Model and Notation (BPMN) [25], Case Management Model and Notation (CMMN) [26], the Decision Model and Notation (DMN) [27]. The primary purpose of these graphical modelling languages is to support communication between human stakeholders, although there do exist execution engines for BPMN and decision tables.

Adding formal semantics to business processes enables machine reasoning and allows exploiting the full potential of process models. This semantic lifting can be achieved by representing a model with ontologies. De Nicola et al. [10] already mention the use of semantic lifting for the alignment of business and IT. Other applications for semantic lifting are process automation [18], process mining [4] or learning [12].

The purpose of ontologies in enterprise modelling is to formalize and establish the sharing, reuse, assimilation and dissemination of information across all organizations and departments within an enterprise. Developing enterprise ontologies started in the 1990s with TOVE [12], The Edinburgh Enterprise Ontology [33] and the organizational memory [1]. More recent work is the Context Based Enterprise

Ontology [22]. Den Haan [11] has used an enterprise ontology to realize a Model-Driven Enterprise Engineering.

In this research we use an enterprise ontology which is based on the concepts of the ArchiMate Standard [31] and extend it with concepts for BPaaS. ArchiMate is an integrated modeling language for enterprise architectures (EA). It is consistent with the TOGAF [29] framework. The overall enterprise architecture comprises a set of closely inter-related architectures: Business Architecture, Information Systems Architecture, and Technology Architecture. Another well-known EA framework is the Zachman Framework, a two dimensional matrix, in which the cells contain models [35].

Hinkelmann et al. [19] combine Enterprise Architecture modelling and Enterprise Ontologies for continuous alignment of business and IT. They show the potential of having both graphical and ontological representations in one environment. In this research we build on that approach and show how ontologies can be used to guide the modeller.

## III. OVERVIEW OF THE BPaaS DESIGN ENVIRONMENT

The BPaaS Design Environment comprises two modelling components - the BPaaS Modelling Environment and the BPaaS Ontology - and the inference engine for the smart alignment (see Figure 2). The BPaaS Modelling Environment encompasses the meta-model for the human-interpretable, graphical modelling languages, i.e. Business Process Model and Notation (BPMN) [25].

The graphical models can be semantically annotated with the ontological concepts, which are defined in the BPaaS Ontology. This means, that both ontology and meta-model development have to be synchronized in the sense that the ontology contains class definitions describing the intended semantics of the elements of the graphical modelling language. This approach provides the possibility of modelling business processes and annotating elements modelled with corresponding functional and non-functional service specifications, such as business, technical and compliance requirements.

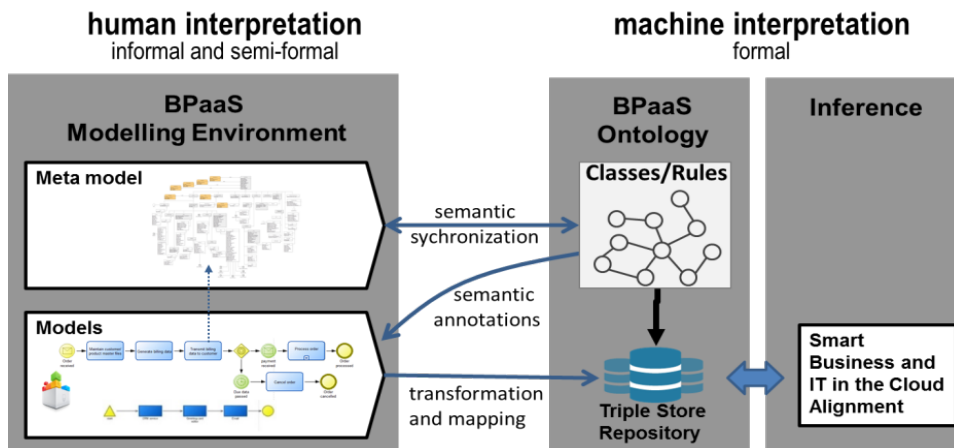


Figure 2. Elements of the BPaaS Design Environment

#### IV. METHODOLOGY

The development of the BPaaS Design Environment was supported by the OMiLAB LifeCycle, which is the basis of Agile Model Method Engineering [21] and has been developed and successfully used in the Open Models Initiative (<http://www.openmodels.at>).

To determine the scope of the modelling framework we sketched a list of questions that the system should be able to answer. These questions are called competency questions. They have been introduced by Gruninger and Fox [16] as a method for enterprise engineering and ontology scope determination [32]. This approach is widely known and was amongst others adopted by De Leenheer & Mens [9], De Bruijn [8] and Cardoso [6].

A focus of the modelling lay in the specification of business process requirements and the description of cloud service and workflow properties. We analysed several real-world business scenarios. This was done in workshops with SMEs and cloud brokers. The business scenarios served as a starting point, since they represent real situations as they occur in enterprises. For the determination of non-functional requirements and specifications we additionally analyzed the Cloud Service Level Agreement Standardisation Guidelines [7].

Based on this analysis both the metamodels of the graphical model environment and the concepts of the enterprise ontology were developed. To demonstrate and evaluate our approach, we implemented a cloud realization of a social media campaign process.

#### V. BPaaS MODELLING METHOD

Models are representing part of reality or a vision in an agreed modelling language. The BPaaS meta model defines the (a) domain-specific business layer and (b) the IT-Cloud relevant technical layer, as well as the interaction between them (see Figure 3).

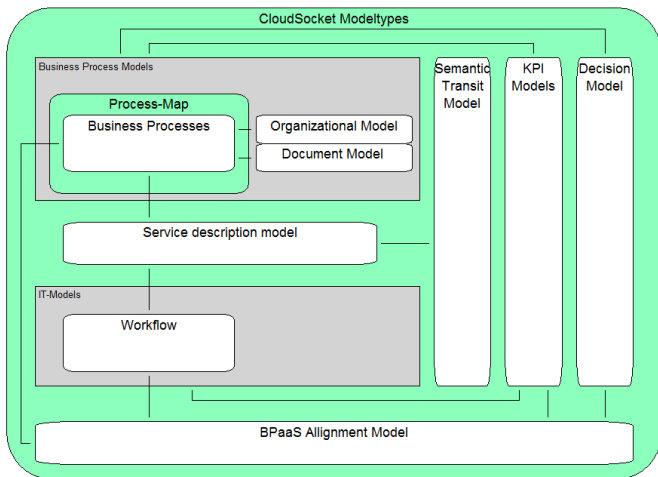


Fig. 3. The BPaaS Model Stack

The business process layer includes business process, organisation and design models. An *organization model* can be

built to illustrate a detailed structure of a working environment for the business process. *Document models* represent documents (templates), which are utilized in the processes as input and output to activities.

The IT layer consists of workflow models, which represent possible implementation of business. It contains technical details and descriptions of cloud services.

The interaction between the domain-specific business layer and the IT layer is done with the Service Description Model. Within this model type there are elements to specify business process requirements and workflow descriptions.

For both business and IT layers, there are links to the key performance indicator (KPI) and the decision models. The KPI model allows to model operational and strategic goals of cloud realizations. Such goals can be quantified by performance indicators.

The Semantic Transit Model is used for semantic lifting of graphical models.

The lines between the model types of Figure 3 indicate that there are references between elements of the different model types. In the following we focus only on business process models, workflow models and service description. An emphasis is on references to business process requirements and workflow descriptions.

##### A. Business Process Models

For the business process modelling we use the BPMN 2.0 standard from OMG. In order to specify the requirements, we added the possibility to assign requirements to parts of a process.

This is performed by making a group for those tasks, which should be implemented by a cloud service and make a reference to a business process requirement specification. In the process of Figure 4, there are two groups "User part" and "Social Media Execution Part". For each of these groups there can be references to Business Process Requirements specifications.

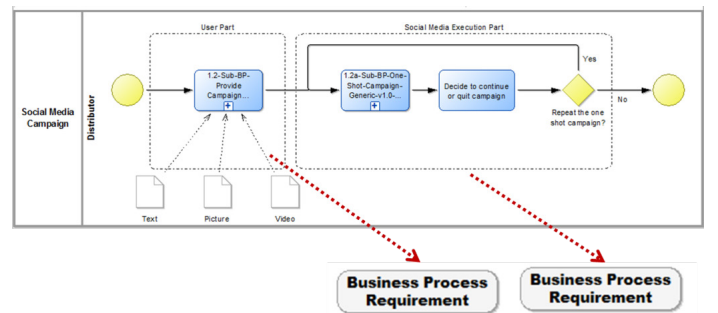


Figure 4. Business Process Model with References from Groups to Requirements

## B. Workflow Models

A workflow model reveals technical aspects of the business process and is therefore meant to be designed and understood by technical people. In particular, in contrast to business processes, workflows do not have requirements. Instead there is a reference of workflow models to descriptions of functional and non-functional properties. Since in workflows the cloud services are actors, we provide references from the lanes to the workflow descriptions (see the two dotted red lines heading to Workflow Description in Figure 5). The workflow description is an aggregation of the properties of the cloud services.

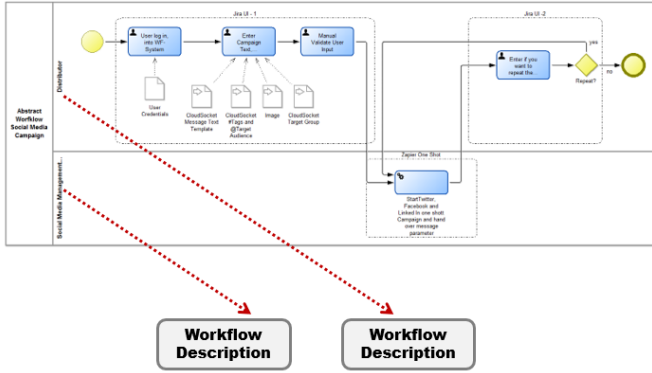


Figure 5. Workflow Model with Reference from Lane to Description

## C. Service Descriptions

In the service description there are to model elements

- functional and non-functional requirements of business processes
- functional and non-functional descriptions of workflows.

Each Business Process Requirement instance contains:

- a description
- functional requirement specifications
- non-functional requirement specifications

Each Workflow Description instance contains:

- a description
- functional description
- non-functional description

### 1) Functional Specifications

The functional requirements specify the functionality of a task or a group of tasks. There are two ways to specify the functionality:

- by assigning hierarchies from the APQC Process Classification Framework [3]
- by assigning an action and an object from a predefined taxonomy, which corresponds to the convention of BPMN to name activities by a verb and a noun [29].

The verb corresponds to the action and the noun to the object.

Figure 6 depicts these two ways of specification applied on the Social Media Campaign process. With respect to the APQC hierarchy, the category “3.3.4.2 Develop Marketing Messages” was chosen as the most appropriate for the activity “1.2-Sub-BP-Provide Campaign Content”, while “3.3.4.5 Execute promotional activities” categorizes the two other tasks. Section II.B describes how the modeling environment supports the semantic annotation with APQC categories.

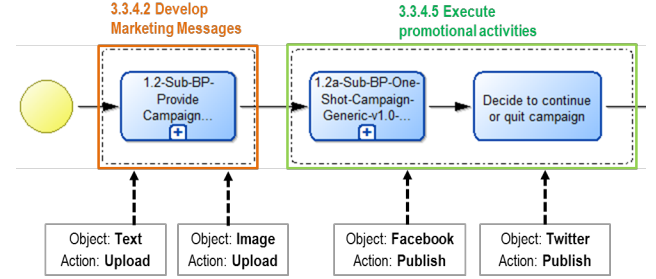


Figure 6. Annotation of Social Media Campaign with APQC and Object/Action Taxonomy

The bottom of Fig. 6 shows objects and actions chosen for the two activity groups, i.e. the first one has “Text” and “Image” as objects and “Upload” as action, while the second activity group is assigned to the two objects “Facebook” and “Twitter”, with “Publish” as action. Assigning actions and objects add additional information to the assigned APQC categories reflecting a domain-specific augmentation of the latter. Additionally, actions and objects allow specifying semantics of non-business-related activities for which no appropriate APQC classification can be found. Likewise, the modeling environment supports the modeler to consistently specify actions and objects by referring to the Functional Description Ontology (see Section VI).

The same objects are used to specify the functionality of a workflow and a service.

### 2) Non-Functional Specifications

The non-functional requirements are grouped into four categories: Data Security, Performance, Support Service and Payment aspects. In each group there are a number of attributes which can be different for business process requirements and workflow descriptions. Business process requirements use business-oriented attributes while workflow descriptions contain more technical knowledge. Figure 7 shows the performance attributes on the business level of the Social Media Campaign process, while Figure 8 shows them in the workflow level. Both screenshots were taken from the modelling environment.

More in detail, performance attributes in the business layer are as follows:

- Downtime refers to the time in a defined period the service is not available. For the business layer we propose downtime in minutes per month;

- Media type refers to the types of media that the user would like to upload in the business process;

Figure 7. Performance Attributes in the Business Layer

- Number of process executions refers to the number of times a process will be executed in a defined period;
- Number of simultaneous users refers to the number of separate cloud service customer users that can be using the cloud service at one time;
- Response Time refers to the time interval between a call of a cloud service (stimulus) and a response by the cloud service provider.

Performance attributes in the workflow layer are as follows:

- Uptime describes the time in a defined period (i.e. month in our case) the service was available, over the total possible available time, expressed as a percentage;
- Capacity is expressed in terms of both hardware and network. For hardware capacity we consider the maximum available data storage. For network capacity, the consider both the maximum simultaneous connections and maximum simultaneous users;
- Response time – refers to the time that is required between the requests from the stakeholder to the system and the receipt, elaboration, execution and response of the request.

Hence, stakeholders from business and IT can model in their own language. For example, to specify the data capacity the business modeller can simply mention the media type he/she wants to upload (e.g. Video) and the number of process executions in a defined time (e.g. 50 social campaign per month). The workflow description specifies the size of the storage. The mapping between these two languages is done by the smart alignment component.

Figure 8. Performance Attributes in the Workflow Layer

## VI. BPaaS ONTOLOGY

The BPaaS Ontology<sup>1</sup> is an extension of the ArchiMEO ontology (see Figure 9). ArchiMEO includes a top-level ontology, which contains general concepts, e.g. for location or time. Additionally, it contains an Enterprise Upper Ontology with the concepts of the ArchiMate modelling language [31] as well as classes which represent the modelling elements of standard modelling languages like BPMN.

<sup>1</sup> The BPaaS Ontology is retrievable under the link <https://github.com/BPaaSModelling>



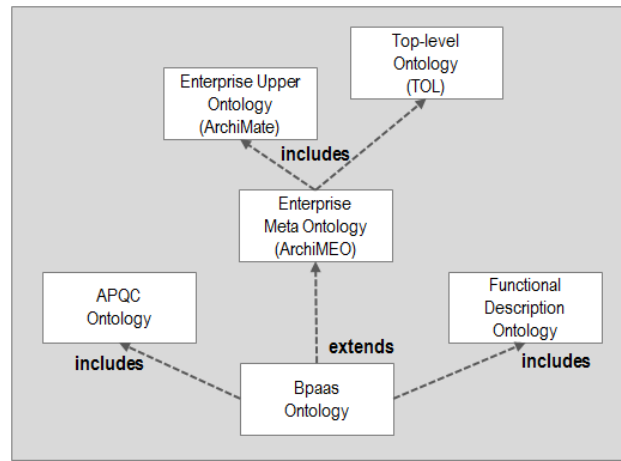


Figure 9. BPaaS Ontology

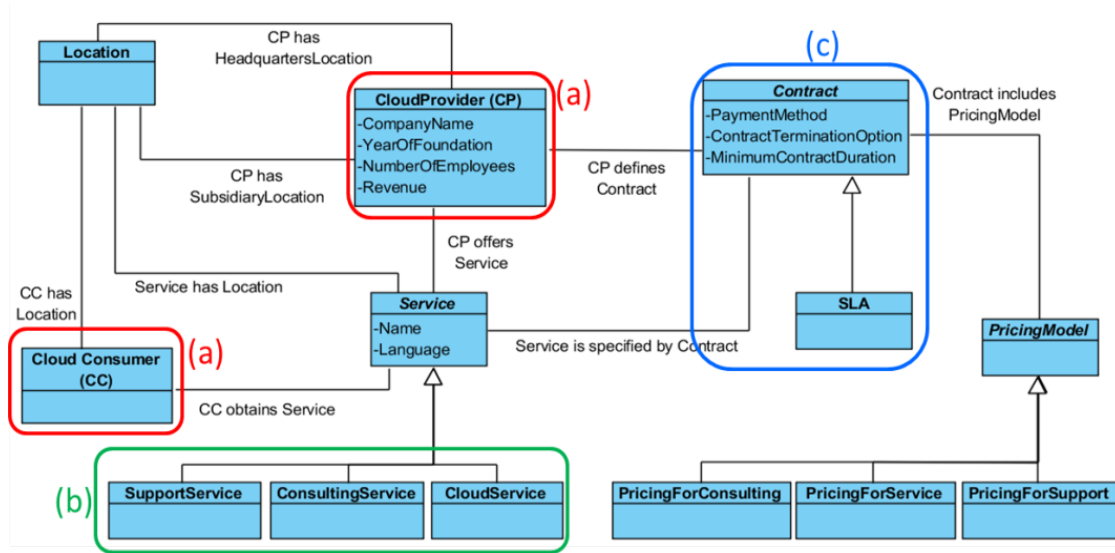


Figure 10. Conceptual Model of Cloud concepts in the business perspective

The BPaaS Ontology includes all concepts required to describe cloud specific requirements in order to achieve a smart alignment of business and IT. The cloud-specific extensions were determined from the analysis of the business scenarios and the competency questions. These also drove the choice for the representation formalism of our BPaaS Ontology. We started with a language as less complex as possible and then adapted according to the expressions needed to represent the artefacts and make the required inferences. Hence, we stick with the RDFS ontology language.

The BPaaS Ontology extends the ArchiMEO ontology according to the BPaaS requirements. Figure 10 depicts the class diagram of the overall conceptual model for the business perspective. The classes are integrated in the class hierarchy of ArchiMEO. The class diagram highlights in red and labelled with (a) the classes of Cloud Provider (CP) and Cloud Consumer (CC). The ontology hierarchy in left part of Figure 11 shows that these classes are modelled as sub-classes of the ArchiMate concept BusinessRole.

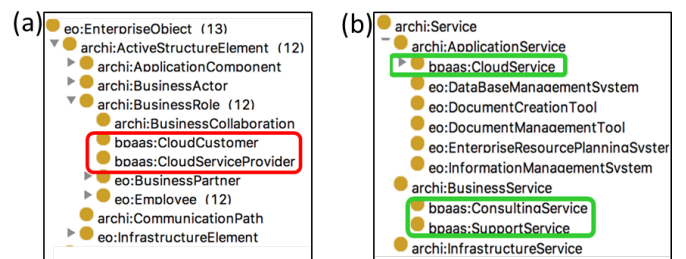


Figure 11. Embedding cloud concepts into ArchiMEO

The right part of Figure 11 shows the different kinds of services labelled with (b). Support and consulting services are classified as business services while cloud services represent a specialization of application services. A third part of the conceptual model (c) shows the (Cloud) Service Level Agreement (SLA), which is defined as sub-concept of Contract, which itself is a Business Object in ArchiMate.

## VII. SEMANTIC LIFTING

Semantic annotations allow the human modeller to categorize and classify elements as well as adding attribute values by using classes and knowledge defined in the BPaaS Ontology. The overall context of this mechanism is outlined in Figure 12.

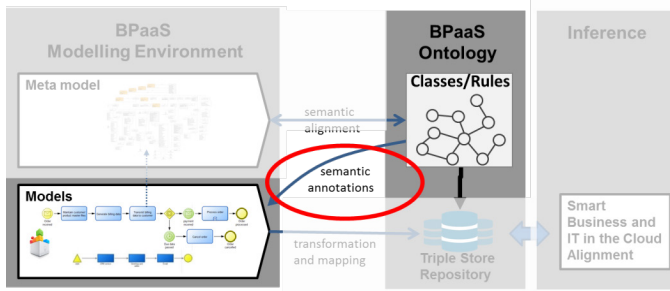


Figure 12. Semantic Lifting of Models by Semantic Annotations

The semantic annotation is implemented with a web service. The web service creates and maintains the link to the ontology. The modelling environment calls the web service for a concept by providing the context from which a query to the ontology is created. The web service returns the resulting classes and instances, which are retrieved from the ontology by evaluating the query. The communication flow is shown in Figure 13.

1. The modelling environment receives at the beginning the link to the web server. While the human models the diagram, the properties can be set. Properties that need to receive a semantic annotation, provide the possibility to make a call to the web service
2. The web service receives the request containing contextual information and queries the ontology accordingly.
3. The result set is returned and processed by the web service according to the interface description to the modelling environment.
4. Once the processing is done, the web service returns the enriched result set to the modelling environment that makes sure that the annotation is placed at the right place in the meta data of the modelling element.

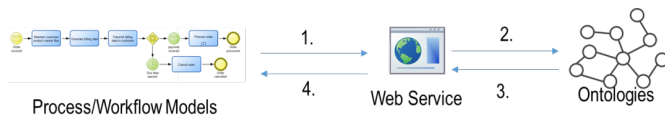


Figure 13. Modelling Environment - Web Service Communication

The notebook is the user interface for maintaining the meta data of modelling elements. Figure 10 shows a notebook of the Business Process Requirement Element. We distinguish two kinds values that can be retrieved from the ontology:

1. results of a query to the ontology
2. sequences of queries

### A. Retrieving attributes values from the Ontology

As an example for a simple semantic annotation we look at the non-functional requirement of storage capacity. On the business level the modeler can specify the media types and the estimated number of items. The possible media types are defined in the BPaaS Ontology. When the user clicks on the button to set the media type, the web service is called (step 1 in Figure 14) and the ontology is queried with the appropriate class, i.e. MediaType. As a result the instances of the class are retrieved and presented to the modeler. The modeller can navigate through the instances choose the a value, e.g. Video mp4 (step 2). The value will appear on the related textbox (step 3).

Retrieving values from the ontology ensures consistent modeling and adaptability of the system. When new media types are needed, the ontology is extended accordingly and immediately the new value is offered to the modeler - without any change of the modeling environment.

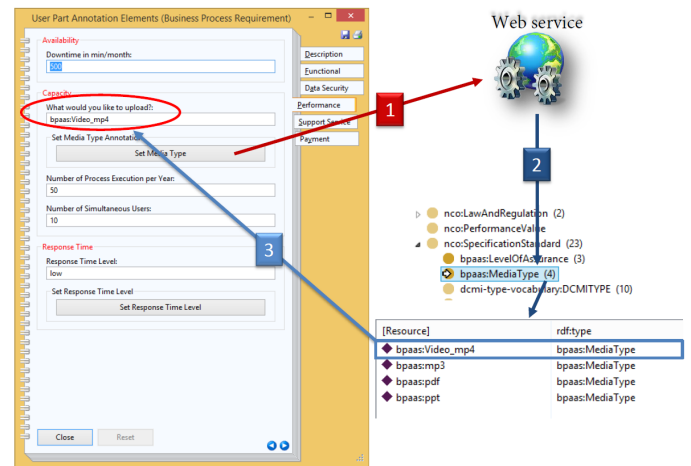


Figure 14. Semantic Alignment by querying the ontology

### B. Sequence on Ontology Queries

The interface not only allows to directly access required values from the ontology but also supports sequences of queries. Figure 15 shows part of the notebook for specifying functional requirements using APQC process classification.

The APQC Process Classification Framework [3] comprises five levels which start from 13 generic business process categories on top and goes down to particular tasks. Since the user might not know the APQC process names by heart, the user clicks on the “Set APQC” button and triggers the web service request. The request includes the context, which is in this case APQC.

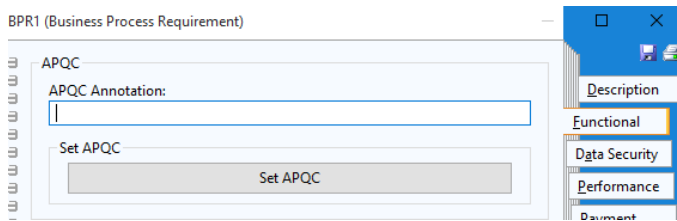


Figure 15. Model Element Notebook - Unannotated

The web service receives the request and queries the ontologies for sub-concepts (first level) of “AmericanProductivityAndQualityCenter” at the namespace “apqc”. The result set of the query is converted into the interface requirements of ADOxx and returned to the modelling environment.

ADOxx receives the results and converts it into the selection box. The user can select the item that fits to the requirement that is being specified. The selection is outlined in Figure 16.

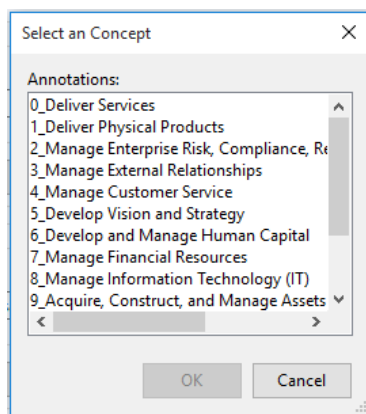


Figure 16. Selection Box with the Results of the APQC Web Service Request

After the user selected the best fitting APQC categorization, the modelling environment asks the user for querying the next level. This question is shown in Figure 17.

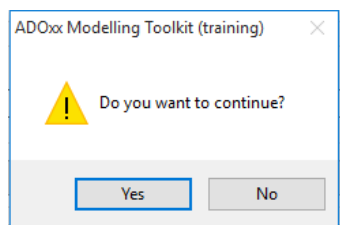


Figure 17. Multi Level Selection User Dialog

If the user decides to increase the granularity of the annotation, “yes” needs to be clicked, if not, the user can stop on the current level by choosing “no”.

If yes is clicked, the modelling environment calls the web service again and takes the current selection as new context. If no is selected, the latest selection is transferred as value into the “APQC Annotation” attribute of the Business Process Requirement notebook. The annotation result is shown in Figure 18.

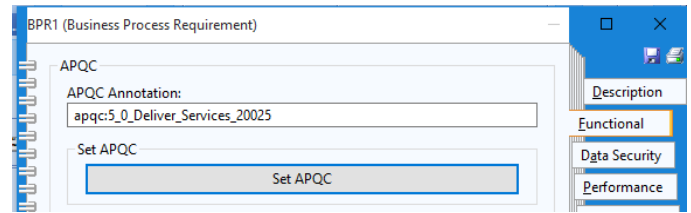


Figure 18. Model Element Notebook - Annotated

While the selection is focused on human interpretation, the annotation is technical by showing the category, name of the process and the process ID.

## VIII. CONCLUSION

In this paper we present a hybrid modeling approach which supports the continuous alignment of business and IT in the cloud. Business Process as a Service provides the end-to-end cloud support for business processes instead of single applications.

Business scenarios were analysed and competency questions were derived in order to determine the scope of the modelling framework.

A graphical modelling environment allows non-technical modelers to design business processes and to specify requirements in human-interpretable way. The same environment supports technical people to specify cloud services and workflows. The BPaaS modelling method was implemented in the ADOxx meta-modelling platform.

The model types were extended with algorithms and mechanisms for semantic lifting to connect the graphical models with the BPaaS ontology. The BPaaS Ontology contains the relevant classes for the smart Business and IT-Cloud alignment. The ontology offers a shared conceptualization for creating domain-specific models. It supports the modeler in using a standard terminology and thus ensures consistent modeling.

The “Social Media Campaign” scenario was introduced to underpin the first evaluation phase. Namely, we demonstrated how our novel conceptual approach for the smart alignment of IT and Business in the cloud can be implemented and used. In the next evaluation phase, we will consider qualitative research methods aiming at understanding both the utility (also known as perceived usefulness) of the prototype and the cognitive effort (also known as perceived ease of use) with respect to the domain-specific models and their annotation. At first, interviews and questionnaires will be addressing people within the European project consortium. Then, we will use the internal feedback as a lesson learned to refine the adopted qualitative



techniques and extend them to outsiders, i.e. cloud brokers as well as modeling and ontology experts.

Future work goes into two directions aiming at improving the presented prototype. First of all, we will iteratively improve both functional and non-functional specifications. This will be mainly done while applying our approach on further cloud application scenarios as we will gather more and more experience. This will imply some adjustment in both our BPaaS ontology and the related library of the ADOxx metamodeling environment. Secondly, we intend to work toward the automatic inheritance of attribute values in the workflow layer. For this purpose, we will (a) derive cloud services specifications, (b) extend the BPaaS ontology according to the derived specifications, (c) develop and implement inferencing rules for automatically inheriting the most appropriate values to workflow attributes.

#### ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Framework Programme for Research and Innovation HORIZON 2020 (ICT-07-2014) under grant agreement number 644690 (CloudSocket). A special thank goes to Kyriakos Kritikos for his valuable comments on the research.

#### REFERENCES

- [1] Abecker, A., Bernardi, A., Hinkelmann, K., Kühn, O., & Sintek, M. (1998). Toward a Technology for Organizational Memories. *IEEE Intelligent Systems and Their Applications*, 13 (3), 40–48.
- [2] Amziani, M., Melliti, T., & Tata, S. (2012). *A generic framework for service-based business process elasticity in the cloud*. In Proceedings of the 10th international conference on Business Process Management (BPM), Alistair Barros, Avigdor Gal, and Ekkart Kindler (Eds.). Springer-Verlag, Berlin, Heidelberg, 194–199.
- [3] APQC (2014) Process Classification Framework Version 6.1.1. Available at <http://www.apqc.org/pcf> [Access December 23, 2015].
- [4] Azzini, A., Braghin, C., Damiani, E., & Zavatarelli, F. (2013). Using Semantic Lifting for improving Process Mining: a Data Loss Prevention System case study. In *3rd International Symposium on Data-driven Process Discovery and Analysis (SIMPDA 2013)* (pp. 62–73). Riva del Garda, Italy: CEUR Workshop Proceedings, Vol-1027. <http://ceur-ws.org/Vol-1027/paper5.pdf>
- [5] Bentounsi, M., Benbernou, S., & Atallah, M. J. (2016). Security-aware Business Process as a Service by hiding provenance. *Comput. Stand. Interfaces*, 44 (C), 220–233.
- [6] Cardoso, Y.C. (2010). Creation and Extension of Ontologies for Describing Communications in the Context of Organizations. Universidade Nova de Lisboa.
- [7] Cloud Service Level Agreement Standardization Guidelines, EC Cloud Select Industry Group (C-SIG), European Commission, 2014.
- [8] De Bruijn, J. (2003). *Using Ontologies - Enabling Knowledge Sharing and Reuse on the Semantic Web*, Tech. Rep. DERI-2003-10-29, Digital Enterprise Research Institute (DERI), Galway, Ireland.
- [9] De Leenheer, P. & Mens, T. (2008). Ontology Evolution. In M. Hepp et al., eds. *Ontology Management - Semantic Web, Semantic Web Services, and Business Applications*. Springer Science + Business Media Inc., pp. 131–176.
- [10] De Nicola, A., Di Mascio, T., Lezoche, M., & Taglino, F. (2008). Semantic lifting of business process models. In *Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOC* (pp. 120–126). <http://doi.org/10.1109/EDOCW.2008.55>
- [11] Den Haan, J. (2009, October 15). *An Enterprise Ontology based approach to Model-Driven Engineering*. TU Delft, Delft University of Technology. Retrieved from <http://repository.tudelft.nl/view/ir/uuid:e2093132-9db7-4cba-bc68-9355f93cb9e3/>
- [12] Emmenegger, S., Hinkelmann, K., Laurenzi, E., Thönsen, B., Witschel, H. F., & Zhang, C. (2016). Workplace Learning - Providing Recommendations of Experts and Learning Resources in a Context-sensitive and Personalized Manner. *MODELSWARD 2016*, Special Session on Learning Modeling in Complex Organizations. Rome.
- [13] Fox, M. S., Barbuceanu, M., & Grüninger, M. (1996). An organisation ontology for enterprise modelling: Preliminary concepts for linking structure and behaviour. *Computers in Industry*, 29(1-2), 123–134. doi:10.1016/0166-3615(95)00079-8
- [14] Fox, M. S., & Gruninger, M. (1998). Enterprise Modeling. *AI Magazine*, 19(3), 109.
- [15] Gartner (2014). “Taming the Digital Dragon: The 2014 CIO Agenda.” Gartner, pp. 1–12.
- [16] Gruninger, M. & Fox, M.S. (1995). Methodology for the Design and Evaluation of Ontologies. *Industrial Engineering* (1995),
- [17] Hepp, M. (2005). A Methodology for Deriving OWL Ontologies from Products and Service Categorization Standards. In *ECIS*, 2005.
- [18] Hepp, M., Leymann, F., Domingue, J., Wahler, A., & Fensel, D. (2005). *Semantic Business Process Management: A vision towards using semantic Web services for business process management*. ICEBE 2005: IEEE International Conference on E-Business Engineering, 535–540.
- [19] Hinkelmann, K., Gerber, A., Karagiannis, D., Thoenssen, B., van der Merwe, A., & Woitsch, R. (2015). A new paradigm for the continuous alignment of business and IT: Combining enterprise architecture modelling and enterprise ontology. *Computers in Industry*. Vol. 80. <http://doi.org/10.1016/j.compind.2015.07.009>
- [20] Hrgovic, V., Karagiannis, D. and Woitsch R. (2013), Conceptual Modelling of the Organisational Aspects for Distributed Applications: The Semantic Lifting Approach. in *COMPSACW 2013, 37th Annual Computer Software and Applications Conference Workshops*, IEEE, pp. 145–150.
- [21] Karagiannis, D. (2015). Agile Modeling Method Engineering. *Proceedings of the 19th Panhellenic Conference on Informatics*, 5–10.
- [22] Leppänen, M. (2005). A Context-Based Enterprise Ontology. In G. Guizzardi & G. Wagner (Eds.), *Proceedings of the EDOC International Workshop on Vocabularies, Ontologies and Rules for the Enterprise (VORTE'05)* (pp. 17–24). Enschede, Netherlands.
- [23] Lynn, T., O'Carroll, N., Mooney, J., Helfert, M., Corcoran, D., Hunt, G., Van Der Werff, L., Morrison, J., & Healy, P. (2014). *Towards a Framework for Defining and Categorising Business Process-as-a-Service (BPaaS)*. 21st International Product Development Management Conference.
- [24] Noy, N. F., & McGuinness, D. L. (2001). *Ontology development 101: A guide to creating your first ontology*. Palo Alto: Stanford Knowledge Systems Laboratory Technical Report KSL-01-05.
- [25] OMG (2011). *Business Process Model and Notation (BPMN) Version 2.0*. Needham, MA: Object Management Group OMG. Available at <http://www.omg.org/spec/BPMN/2.0/PDF/>
- [26] OMG (2013). *Case Management Model and Notation (CMMN)*, Version 1.0. Needham, MA: Object Management Group OMG. Available at <http://www.omg.org/spec/CMMN/1.0/PDF>
- [27] OMG (2015). *Decision Model and Notation Version 1.0*. Needham, MA: Object Management Group OMG. Available at <http://www.omg.org/spec/DMN/1.0/PDF>
- [28] Papazoglou, M. P., & van den Heuvel, W. J. (2011). Blueprinting the cloud. *IEEE Internet Computing*, 15(6), 74
- [29] Silver, B. (2011). *BPMN Method and Style*, Second Edition (Second Edition). Aptos, CA: Cody-Cassidy Press.
- [30] The Open Group. (2011). *TOGAF® Version 9.1*. Van Haren Publishing.
- [31] The Open Group. (2012). *ArchiMate 2.1 Specification*. The Open Group. <http://pubs.opengroup.org/architecture/archimate2-doc/> [Access November 3, 2015]

- [32] Uschold, M. & Gruninger, M., (1996). *Ontologies: Principles, Methods and Applications*. Technical Report, University of Edinburgh Artificial Intelligence Institute AIAI, (191).
- [33] Uschold, M., King, M., Moralee, S., & Zorgios, Y. (1997). The Enterprise Ontology. *The Knowledge Engineering Review*, 13.
- [34] Völter, M., Stahl, T., Bettin, J., Haase, A. and Helsen, S. (2013) *Model-Driven Software Development: Technology, Engineering, Management*. John Wiley & Sons.
- [35] Zachman, J. A. (2008). *John Zachman's Concise Definition of The Zachman Framework*. Zachman International. Available at <http://zachmaninternational.com/>