

# An Ontological Approach to Automate Interface Configurations

Frank Oemig<sup>1</sup>, Bernd Blobel<sup>2</sup>

<sup>1</sup>Agfa Healthcare, Bonn, Germany

<sup>2</sup>eHealth Competence Center, University Hospital Regensburg, Regensburg, Germany

## Abstract

**Background:** Working interoperability can only be demonstrated with running systems. A prerequisite is an installation and setup of the appropriate interfaces. Currently, this process is performed manually. **Objectives:** This paper aims at analyzing the background to migrate that process to an automated installation. **Methods:** The solution is based on the Generic Component Model (GCM) as an architectural framework and the Communication Standards Ontology (CSO) and Basic Formal Ontology (BFO) presented in previous papers. **Results:** An ontology-based approach to automatically design communication interfaces is proposed. **Conclusions:** The offered approach can be extended to Webservice adaptations, structural translation of documents or terminology services.

## Keywords

Generic Component Model, Basic Formal Ontology (BFO), Communication Standards Ontology (CSO)

## Correspondence to:

**Dr. Frank Oemig**

Agfa HealthCare GmbH

Konrad-Zuse-platz 1-3

53227 Bonn, Germany

E-mail: Frank.Oemig@agfa.com

EJBI 2012; 8(3):18-21

## 1 Introduction

Jurisdictional and user requirements enforce the utilization of communication and document standards in healthcare environments like HL7 v2, V3 and/or CDA [1, 2, 3]. However, a communication standard can be seen and treated as a language combining a grammar and the appropriate vocabulary. Both enable data and information exchange, but do not guarantee semantic interoperability. Despite those preconditions - the existence of communication standards -, applications are not directly enabled to “speak” or “understand” this language. But even if the applications are programmed to use such an interface language, no matter whether it is a real communication standard or not, thorough configurations are necessarily requiring a deep knowledge of both, the internals of the application and the communication language. This paper analyses the way, this knowledge can be represented and reused facilitating the use of ontologies.

## 2 Methods

The work presented must be based on an appropriate architectural framework for formally describing the real system in question. The representation of the architectural components of the system has to be based on appropriate ontologies. As appropriate architectural framework, the Generic Component Model (GCM, [4]) has been used. Our approach is supported by, and based on, the so-called communication standards ontology (CSO) [5]. Ontologies incorporate terminologies [4] with interrelated terms, which represent concepts, i.e. the terms are linked on a very granular level so that the semantics of the relation is clearly defined. This way ontologies describe real systems. On the other hand, a system can be described by ontologies on different levels of details resp. granularity. Therefore, a basic methodology to work with ontologies is the reuse (“import”) of other ontologies. One possible reuse is a mapping of those ontologies [4]. To support this

fundamental approach, developing new ontologies based on the same top-level ontology is not only helpful but facilitates reuse and harmonization. For this purpose, the Basic Formal Ontology (BFO) [6] has been introduced as a top-level ontology. Following, the foundations of this paper are introduced shortly.

## 2.1 The GCM as an Architectural Framework

The Generic Component Model is a system-theoretical architectural approach to any kind of system, providing the representation of the different aspects the system serves, i.e., the different specialties impacting the system's design and use (in the GCM called domains), the composition/de-composition of the system's components, and finally the development process for designing, specifying, implementing and deploying the solution. At the abstract level of an architectural model, the development process dimension is represented by viewpoints defined in ISO/IEC 10746 "Information technology – Reference Model – Open Distributed Processing (RM-ODP)" [7].

## 2.2 The Basic Formal Ontology (BFO)

From its origin, the Basic Formal Ontology (BFO) has some kind of philosophical orientation as can be seen below when explaining the highlevel concepts. Despite that - or just because of that -, it provides an upper ontology which can therefore be used as a foundation for other ontologies [6]. Numerous research groups are utilizing BFO this way. Gene Ontology and Protein Ontology are examples resulting thereof.

One of BFO's primary advantages is its size. It is very small and comprehensive, so that ontology developers can easily align their concepts to it.

BFO groups everything below "entity" which acts as the root to the whole tree. On the next level, everything is divided into entities that continue or persist through time (objects, qualities, functions) which are named "continuants" as the most generic concept. The other branch provides the entities for events or happenings (processes) in which continuants participate. It is named "occurrents".

## 2.3 The Communication Standards Ontology (CSO)

CSO is based on BFO and represents an upper ontology in itself as well because it provides the generic concepts special communication standards ontologies like for HL7 v2.x or V3 have to align with. Its primary purpose is to provide a reference architecture which is based on and aligned with the GCM, but it also provides the common foundational concepts for different communication standards. CSO has been used to align the concepts from different communication standards like HL7 v2.x and V3 with each other [3], and to provide a mapping between

them by the help of a mediator domain. For the latter, "Advancing Clinico Genomic Trials on Cancer" (ACGT) [13] has been selected.

CSO is used as the starting point for the approach described following.

## 3 Results

CSO [14] has been designed to formalize the concepts of communication standards and to align them with an upper ontology. As mentioned above, BFO has been the choice so far, but other formal top-level ontologies can be used as well under the premise that perhaps higher level structures must be changed.

Another main topic of the Thesis [13], where the presented issues have been comprehensively elaborated, was the mapping to a central mediator domain. For the sake of simplicity and to demonstrate the support of the guiding principles for the use of ontologies, ACGT [13] has been selected although essential concepts were missing. It is easier to introduce the missing concepts instead of creating/re-inventing a completely new ontology. In principle this is creating a new ontology on top of another one. A simple investigation reveals that BFO itself is constructed this way out of a set of ontologies. (E.g. BFO has no relations. They create an ontology for itself.)

The completeness of the concepts in CSO as a foundation and their relationships have been demonstrated in the Thesis and in other papers. Furthermore, it is explained how different ontologies can be mapped to each other by establishing a structured mapping of "sub-models" within the domain resp. ontologies under the premise that the granularity of the concepts is detailed enough. The foundation for this approach is an ontological representation of all necessary concepts up to an "atomic" granularity based on an architectural approach to that ontology, i.e. the concepts do not use pre-coordination.

The described mapping allows for linking two different ontologies no matter whether they refer to a communication standard, a mediator domain, business objects or a database.

### 3.1 Describing the Application

Consequently, as introduced above, an application can be described the same way, i.e. all concepts must be created and aligned with the predefined mediator ontology. It includes the relationship to other concepts, especially to build the important sub-models which are needed for a mapping.

At this point, a very specific choice can be made: The mapping was described in the Thesis on the basis of a mediator domain, for which ACGT has been selected. Hence we can either describe the mapping of the application ontology to this mediator ontology, or we can make our application ontology this mediator ontology, so that no explicit mapping is necessary any more. Both ways are

possible and the choice depends on the use cases.

### 3.2 Use of Knowledge

A central question is: “what can be done with this kind of knowledge?” Following some possible scenarios are mentioned:

- Interface configuration
- Webservice adaptations
- Structural translations of documents
- Terminology services

Within this paper, we will elaborate on the first one.

### 3.3 Deriving knowledge

Communication standards have been designed to simplify an information exchange. As mentioned above, they represent languages a system has to learn. Like all languages, the standards are not defined precisely enough resp. the vendors have implemented a dialect of the language according to what they have understood when examining the standard. Whether a certain vendor has really taken care of all sections being relevant for a correct implementation or not, does not matter. Implementations are done by humans. Consequently, the implementations vary because of the different intellectual performance of the developers. (Although this sounds negative, it is the human nature and therefore inevitable.) But for computer programs these diverging dialects lead to non-managable differences.

Therefore, each implementation has some more or less important differences to what the standard has defined. In the context of our focus, we may speak of different dialects. The mapping of concepts between the different dialects can be represented exactly the same way as the mapping of the communication standards itself.

The approach is based on the representation of knowledge. In contrast to the relative superficial realisation of interfaces without any frameworks oder other mandatory constraints, ontologies are formally oriented and organised, i.e. they are precise on the appropriate granularity level. A specific application must be formally described using an application ontology in advance, so that all concepts are described and interlinked. An application is therefore exactly described by an application ontology. In a second step of the preparation, the mapping onto a mediator domain must be done, so that the relation of concepts among the different ontologies is exemplified.

As a result, the configuration of an interface can be derived. In this case, the ontological representation of the application and its mapping to the communication standard must be converted into an appropriate interface configuration.

Of course, depending on the use case it does not make sense to derive the necessary information from the describing ontologies each time it is needed, because the way the information is described is really complex and the evaluation is time consuming. Therefore, an interpreter has to translate the information into an alternative representation once.

When using the web ontology language (OWL) [14, 15, 16] as a means of formalisation, an XSL translation (XSLT) can be used therefore, because OWL itself is defined based on XML.

Nowadays, communication servers are used to translate messages using one dialect into a message using another dialect or a totally different language. This translation represents a mapping process as well. It is formalized by using the scripting language of the communication server. In other words, it is also aka knowledge representation by different means.

So, it should be possible, to translate the knowledge provided in an ontological representation of the mapping into another representation using the scripting language of the communication server.

### 3.4 Generating knowledge

The whole process can also be reverted. The configuration of a communication server or an application’s interface can be used as input to generate the knowledge hidden within it, if the formalisation of the interface configuration is known.

## 4 Discussion

An issue worth a deeper discussion is the amount of efforts required to formalize the necessary depth of details in order to make use of this approach. It must be clear, that the first attempt will cause a reasonable amount of efforts to start with this process. But generating the knowledge out of the configuration of running systems will also cause a lot of formalization.

Later on, the information must be versioned and modularized to simplify its maintenance. Both requirements are in principal supported by OWL, the web ontology language [14, 15, 16]. The versioning process can directly re-use the version information within the files. The modularization happens by the import of ontologies, so that the final configuration will import all relevant modules.

## 5 Conclusions

The ultimate options to develop applications are not in place yet. An underlying ontology like CSO can be used to model and maintain the information needed to support interface configurations. Given the current status of system integration it will take several years to reach this point. In summary: Currently, we are not much beyond a syntactical data exchange.

## 6 Acknowledgements

The authors are indebted to their colleagues from SDOs such as HL7 for open minded and kind cooperation and valuable support.

## References

- [1] HL7 v2.5 – version 2.5, [www.hl7.org](http://www.hl7.org)
- [2] HL7 V3 – version 3, [www.hl7.org](http://www.hl7.org)
- [3] HL7 CDA – The Clinical Document Architecture, [www.hl7.org](http://www.hl7.org)
- [4] Oemig F, Blobel B: "Harmonizing the semantics of technical terms by the Generic Component Model", 10th International Special Topic Conference of the European Federation for Medical Informatics in Reykjavík Iceland, 2-4 June 2010, <http://www.sky.is/efmi-stc-2010-.html>
- [5] Oemig F, Blobel B, A Communication Standards Ontology Using Basic Formal Ontologies, In: L. Bos, B. Blobel, S. Benton, D. Carroll (Eds.): Medical and Care Compunetics 6, p. 105-113. Series Studies in Health Technology and Informatics, Vol. 156. IOS Press, Amsterdam, Berlin, Oxford, Tokyo, Washington, 2010.
- [6] "The Open Biomedical Ontologies", <http://www.obofoundry.org/>
- [7] International Organization for Standardization (1996) ISO/IEC 10746-2, Information Technology – Reference Model for Open Distributed Processing: Foundations. Geneva. ISO/IEC 10746 / ITU-T x.901.
- [8] Blobel B: "Application of the Component Paradigm for Analysis and Design of Advanced Health System Architectures". International Journal of Medical Informatics 2000; 60, 3: pp 281-301.
- [9] Oemig F, Blobel B: "An Ontology Architecture for HL7 V3: Pitfalls and Outcomes", in Proceedings of the World Congress 2009, Munich, Germany, <http://www.wc2009.org>
- [10] Blobel B: "Architectural approach to eHealth for enabling paradigm changes in health", Methods Inf Med 49,2 (2010), 123-134.
- [11] Oemig F, Blobel B: "Semantic Interoperability Adheres to Proper Models and Code Systems: An Examination of Different Approaches or Score Systems" in CeHR Conference Proceedings 2007: "eHealth: Combining Health Telematics Telemedicine, Biomedical Engineering and Bioinformatics to the Edge", S.97ff. In: Blobel B, Pharow P, Zvarova J, Lopez D (Editors) ISBN 978-3-89838-089-8 (Aka), ISBN978-1-58603-834-2 (IOS Press), p.97-104
- [12] Oemig F: Entwicklung einer ontologiebasierten Architektur zur Sicherung semantischer Interoperabilität zwischen Kommunikationsstandards im Gesundheitswesen. Inaugural-Dissertation zur Erlangung des Doktorgrades Doktor der Biomedizinischen Wissenschaften, Universität Regensburg, Medizinische Fakultät, Regensburg, 2011.
- [13] "Advancing Clinico Genomic Trials on Cancer", Project ID: FP6-IST-026996; European Commission, 02/06-01/10, <http://www.eu-acgt.org/>, <http://www.ifomis.org/wiki/ACGT>
- [14] OWL: "Web Ontology Language", W3C Recommendation 10. February 2004, <http://www.w3.org/TR/owl-features/>, last accessed 5.5.08
- [15] OWL2: Web Ontology Language, Primer: <http://www.w3.org/TR/owl2-primer/>, last accessed 10.06.10
- [16] OWL2 Language Profiles, W3C: <http://www.w3.org/TR/owl2-profiles/>, last accessed 10.6.10