Building Highly Structured Semantic Repositories through Reuse and Formalisation of Business Standards
Application in the Aircraft Maintenance Domain

Farid Cerbah
Dassault Aviation
78, quai Marcel Dassault, 92552 Saint Cloud, France
Farid.Cerbah@dassault-aviation.fr

Bernard Vatant
Mondeca
3, cité Nollez, 75018 Paris, France
Bernard.Vatant@mondeca.com

Abstract

This paper describes the approach followed in the design of a semantic model in the aircraft maintenance domain. The constructed semantic model is intended to be exploited to ensure highly interoperable exchanges in business processes. We give an overview of the multi-step and partially automated acquisition approach that has been defined to support the elaboration of ontology-based models from legacy data. In our approach, content-oriented business standards are considered as valuable sources that can be redesigned and improved through semantic modelling. We concretely show how the transitioning approach has been applied to one of the main business standards in aeronautics, the ATA numbering system.

1 Introduction

Over the past ten years, software infrastructures supporting aircraft design and manufacturing have tremendously grown in complexity. This is a consequence of stronger requirements regarding collaborative work capabilities that arise from new aircraft project (virtual) organisations. Software infrastructures should allow numerous “distributed” partners from different companies to share and work on a common and synchronized definition of the aircraft all along the project lifecycle. In such concurrent engineering frameworks, successful completion of business processes critically depends on the ability to effectively exchange and share product data between heterogeneous applications. It is widely recognized in the aeronautic industry that improving interoperability solutions can result in significant added-value in terms of competitiveness and productivity. This makes the aero sector a highly relevant target for semantic-powered SOA technology.

The work described in this paper is part of a case study developed in the TAO European project[1]. While being representative of knowledge-based services encountered in complex aeronautic information systems, this case study is specifically focused on providing semantic-based support to aircraft maintenance processes. More precisely, the goal we are pursuing is to show how maintenance legacy data repositories and business processes can be migrated into a semantic-powered SOA framework. The added value of the resulting service-based applications will be practically assessed in terms of knowledge reuse, interoperability, and increase of automation potential of both services and data exchanges. In the reengineered framework, ontology-based models of technical data and services are intended to be exploited to support highly interoperable exchanges between maintenance

[1] TAO (Transitioning Applications to Ontologies), www.tao-project.eu

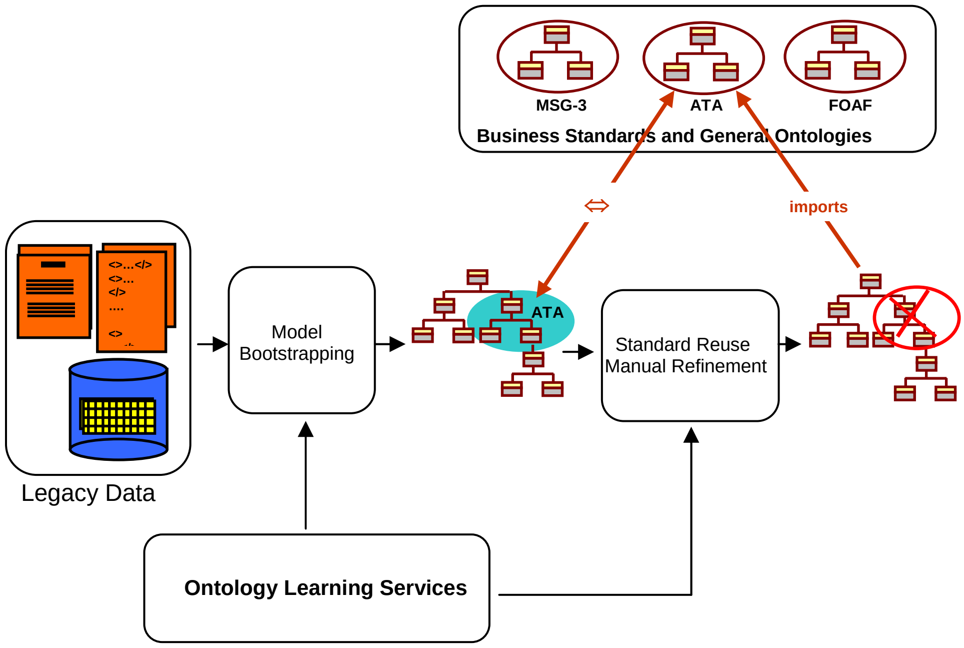

Figure 1. The Ontology acquisition process

management systems and to provide standardised knowledge inputs for user mediated semantic annotation of technical publications.

In this case study, several identified requirements are related to the need for improved knowledge sharing as a way to strengthen interoperability. In the aeronautic sector, it is widely accepted that part of the solution to data interoperability lies in the use of business standards. Many effective standards defined by industry organisations are basically content-oriented and tailored to ensure semantic interoperability in automated processes. However, the lack of suitable representation languages has not allowed to make them fully exploitable for this purpose. Semantic languages will soon be perceived as powerful devices that could enable to effectively exploit and improve existing business standards for better interoperability in software infrastructures. In this respect, our goal is to demonstrate how in the complex aircraft maintenance domain a stratified reference model can be constructed through systematic reuse of widely accepted standards turned into ontologies and existing general-purpose ontologies. This upper reference model is intended to be reused and eventually specialised by all applications pertaining to this specific domain.

To implement such a stratification of domain and corporate knowledge, the representation framework should allow the definition of comprehensive models through progressive refinement of self-contained modules. Ontologies with their classification and modularisation capabilities appear to be good candidates to fulfill these requirements.

In this short paper, we give an overview of the multi-step and partially automated acquisition approach that have been designed to guide the designers when transitioning legacy applications to ontologies. Then, we concentrate on a widely used aero standard and show how it can be improved though semantic modelling.

2 Overview of the acquisition process

First step in the acquisition process is model bootstrapping (see figure 1). Learning services are applied to automatically generate the first ontologies from the legacy application-specific data and models. The approach we are advocating is more corrective than constructive. We consider that such a proactive approach allows to avoid the "blank page syndrome" in knowledge intensive migration projects, especially when the users are not familiar with semantic modelling and technologies.

However, it should be mentioned that the efficiency of this approach critically depends on ontology service performance. By following this way, we make the strong assumption that ontology learning services will result in relevant and exploitable models and correcting the automatically generated models is less costly than manually building models from scratch.

After this bootstrapping step, the resulting models need to be refined through several additional steps.
Reuse of pre-existing ontologies is another important step that could also be supported by specific services. The goal of this step is to replace parts of the current model by semantically equivalent parts defined in pre-existing ontologies. Reuse has two effects on the model under construction:

- Modularisation: The model is segmented into self-contained modules that make it more structured, easier to understand and maintain.
- Normalisation: Widely accepted concept definitions are incorporated in the model. Such concepts with a high potential to be shared by several applications are needed to fulfill interoperability requirements.

As shown in figure 1, reuse can concern semantic resources from different domains and at different genericity level. For example, when elaborating ontologies for an application in aircraft maintenance, a reusable ontology for the ATA model could be imported instead of using a specific definition of the same aircraft breakdown (see section 3). However, general ontologies like FOAF[2] can also be fruitfully exploited to get homogeneous descriptions of actors involved in logistic activities. Some concepts could be borrowed from a related domain. For example, aircraft structural maintenance descriptions may have to refer to concepts already defined in ontologies dedicated to airframe design.

To define the ontology model in this case study, we consider that a particular domain should be represented by a stratified model defined as a structured set of ontologies to be shared by all applications pertaining to this specific domain. In addition to this domain model, a privileged set of general-purpose ontologies is selected. Both domain model and general-purpose ontologies will be exploited as the reference background to all applications.

3 Semantic Models for ATA Numbering System

We discuss in this section various ways of formalising a typical business standard as highly reusable ontologies. The well-known ATA numbering system, defined in the ATA iSpec 2200 standard[3], is one of the primary models required since it is used as an aircraft part structuring and identification device in a large range of applications.
ATA System is used in various legacy databases, documentation and business management systems for different functional purposes. The migration of such systems to a common interoperable semantic framework meets the challenge of this diversity, since actually behind a single numbering system are hiding different implicit interpretations, each relevant to specific functional applications. The hierarchical structure of the codes is based on the breakdown of a typical aircraft (see example in figure 2). But this decomposition is actually used for a parallel breakdown of aircraft documentation into matching Chapters, Sections and Subjects. This structure can be used in any kind of document or data, relevant to any process, from aircraft production to maintenance tasks. So the structure which seems to be at first glance a meronymy (structure using whole-part relationships) of an aircraft, is actually used to define the internal structure of documents. And indirectly, when those documents or data are dealing with maintenance tasks or any kind of process attached to an aircraft component, the ATA structure is a kind of classification or indexing scheme for the process themselves.

Effectiveness and pervasiveness of the ATA system is actually grounded in this fundamental ambiguity. The same code being used to identify an aircraft component, a section of documentation, or

[2] http://xmlns.com/foaf/1.0/
[3] Air Transport Association: htpp://www.airlines.org

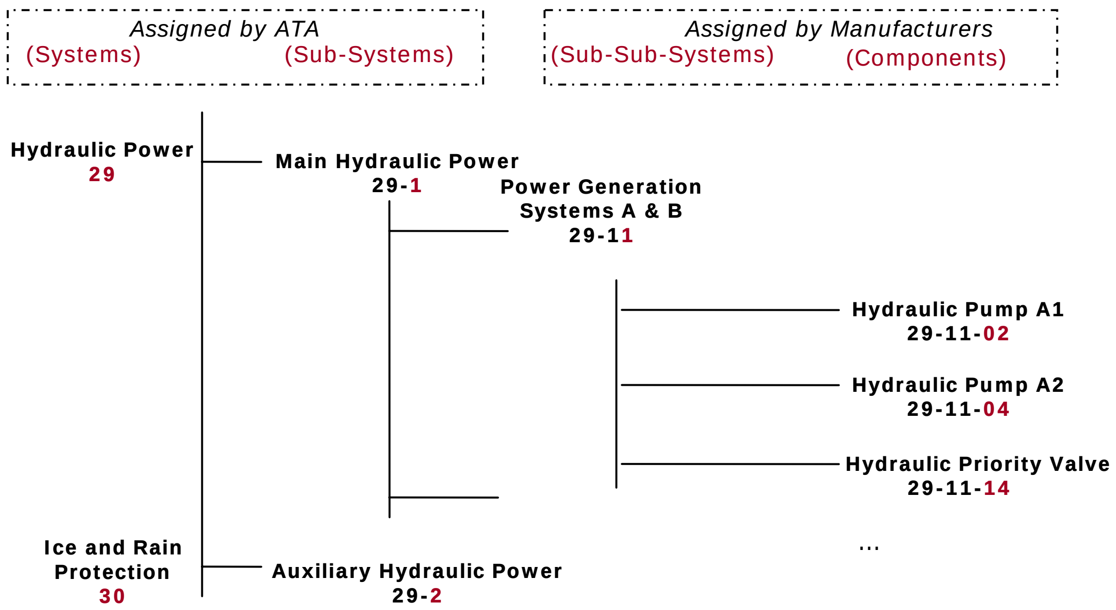

Figure 2. Extract of ATA Model

a maintenance task is a very interesting feature in legacy business management, but in semantic migration, this ambiguity is not sustainable. Aircraft components, document sections and process are different types of objects, bearing different attributes and hierarchy types, and of course different identifiers. How all those identifiers leverage the common numbering system, and how the different hierarchies leverage the common numbering structure is a critical question in transition.

Moreover, either of the above semantics can apply, depending on the context, to either a generic aircraft, or to a specific aircraft model, e.g., the Falcon F7X product, or even to some specific instance of such a model. If all those interpretations were to be used in separate migrations to a semantic format, they would certainly come up with different modelling options, all valid in their context of use. The main modelling challenge is therefore to take into account those different semantics, while ensuring their interoperability through a common reference model.
Another interesting and challenging feature of the ATA System is its extensibility. Each manufacturer can use specific digits to represent specific components of its aircraft models (see last two levels in example of figure 2), and of course use it as the base system, to structure documentation and data, classify tasks etc.
This extensibility must be preserved in the ontology, which must also provide a way to distinguish the generic elements from the specific extensions. This distinction is made in the original numbering system by specific rules defining which part of the number is generic, and which part is assigned by the manufacturer. This implicit fine-grained semantics encoded inside identifying strings cannot be used as is in a semantic framework, and must be explicitly expressed otherwise

The following sections are presenting three modelling options of the ATA system, all of them using the features of OWL-DL[4] language, or even of its simpler species OWL-Lite[5]. This choice allows those models to support reasoning tools available for this level of logic.

Option 1 : ATA concepts as classes

This is the most natural option in semantic migration, and sometimes considered as the only possible one for concepts. In this option, each ATA number is used to identify a class. Most naturally, the hierarchy defined by the numbering system is derived into a class-subclass hierarchy. Such a

[4] http://www.w3.org/TR/owl-ref/#OWLDL
[5] http://www.w3.org/TR/owl-ref/#OWLLite

representation assume that instance of such classes are clearly defined. But as explained above, those instances might be either specific systems, e.g., the Engine Fire Extinguishing System of the Falcon F7X, or of a specific aircraft of this model. But it could be as well a maintenance task relevant to any of those, or a documentation or data item relevant to any process or task relevant to them. Therefore this option defines parallel class hierarchies for aircraft components, document sections, tasks and subtasks. It uses the ATA numbers as a common attribute value for instances of each class at the same hierarchy level, and expresses local constraints controlling the integrity of hierarchy and values of ATA numbers used.

Option 2 : ATA concepts as knowledge base individuals

In this option, the various uses of ATA concepts are defined as individual instances of generic classes, such as System, Subsystem, Unit, for aircraft components, or Chapter, Section, Subject, for documentation material breakdown. The parallel hierarchies are defined using meronymic relationships, and transversal relationships link instances at the same level. The ATA number is used for instances the same way as in Option 1, it can be used for integrity checking, achieved by rules external to the OWL model itself. Such rules could be written for example as SPARQL constructs, the hierarchy being declared only for e.g., aircraft components, and computed for the matching material hierarchy.

Like in option 1, an arbitrary number (N) of parallel hierarchies must be maintained, along with a matching number of transversal properties (N^2).

Option 3 : ATA concepts as indexing concepts

The last option is considering ATA System as a generic *concept scheme*, a hierarchy of abstract concepts. The hierarchy is declared once for all in the scheme structure, and the various instances using this scheme are linked to the concepts through indexing relationships.

Like in option 2, an arbitrary number of classes can be defined with a parallel hierarchy, but instead of needing N^2 one-to-one relationships, each of those hierarchies is indexed against the generic scheme. Actually, the hierarchy of systems, chapters, tasks, etc. does not need to be declared in an exhaustive way, it can be inferred from the generic hierarchy of concepts, using ad hoc inference rules.

The RDF language SKOS[6], currently on the track to be a W3C recommendation, is providing the framework to express both the concept scheme structure, and the indexing relationships.

4 Conclusion

Elaborating such a stratified model is a quite large modelling initiative that might look in contradiction with the assumption we previously made on the need for limited acquisition efforts. We should stress that, to meet interoperability requirements, the availability of these ontologies with high reuse potential is an inescapable prerequisite. While we can reasonably assume that some of the needed ontologies will be made available by external organisations, such as standardisation organisations, the first transitioning projects might be burdened by the need to initiate the development of missing upper ontologies.

Our contribution on the ATA model clearly demonstrates how content-oriented oriented standards can be improved through ontology based modelling and, more importantly, turned fully operational to ensure interoperable exchanges in business processes.

[6] http://www.w3.org/TR/swbp-skos-core-guide/