A Formalism to Describe Open Standards in order to Generate Application Profiles

REBAI Issam[a], DE LA PASSARDIERE Brigitte[b], LABAT Jean-Marc[b]
[a]*Crip5, Université Paris Descartes, Paris (France)*
[b] *LIP6, Université Pierre et Marie Curie, Paris(France)*
issam.rebai@math-info.univ-paris5.fr
brigitte.de-la-passardiere@lip6.fr
jean-marc.labat@lip6.fr

Abstract: The technical problems of compatibility, of exchange of data and of interconnection between services, tools and LMSs has increased with the multiplication of software applications, platforms, and e-learning formats. Standards, specifications and formalisms have been developed to obtain interoperability. However, in practice, it is necessary to maintain some flexibility in order to answer specific needs. This is why e-learning actors need tools that could help them adapt standards to their context of use. In this paper, we describe a formalism to define and generate specific application profiles based on standards. Its goals are to provide a means to describe what differs from the standard (change in vocabulary, new element, others changes), to make the interface generation easier and to harmonize and structure application profiles. We are going to present the formalism to create application profiles.

Keywords: Formalism, application profile, norm, open standard, LOM, Dublin Core

Introduction

The growing use of e-learning has created a market for companies such as editors of computer based educational materials, editors of LMSs (Learning Management Systems)… Simultaneously, the multiplication of software applications, platforms, and formats for training modules has created technical problems of compatibility, of exchange of data and of interconnection between services, tools and LMSs.

To solve these problems, the actors of the market and the e-learning community have created consortiums, such as IMS Global Learning Consortium Inc., in order to define specifications that make their tools and products compatible and interoperable. E-learning actors must respect these standards, specifications and formalisms to guarantee the interoperability between products. In this paper we refer to them as "*Open Standards*".

In practice, imposing to all actors the use of Open Standards has proved illusory. It is necessary to maintain some flexibility in order to answer specific needs while remaining compatible with norms and standards. Many people and organizations are involved in e-learning and try to federate their works. They very often need to adapt the norms and standards to their contexts and communities. To do so they define application profiles.

How do the actors usually manage therefore to adapt a standard to their context of use? More precisely are there means that facilitate such a specialization or customization of an Open Standard? The research questions then deal with how to help them in this task.

So, in this paper, we propose a structural formalism to define and generate a profile from Open Standards.

1. Open Standards in the e-learning domain

Norms, standards and specifications are documents written to the intention of a community and intended to expose the consensual decisions of the initiators. "A *norm* is a document established by consensus and approved by a recognized organism, that provides, for common and repeated uses, the rules, the leading lines or the features, for activities or their results, guaranteeing an optimal order level in a given context" [1]. *Standards* can be defined as "documented agreements containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for their purpose" [2]. "Specifications are less evolved than standards and attempt to capture a rough consensus in the user or implementer community… It can take long time before specifications are finally approved as standards" [3]. In the context of e-learning technology, standards are generally developed to be used in system design and implementation for the purpose of ensuring interoperability, portability and reusability.

These three kinds of documents require know the formats, the protocols, the schemas in order to product materials and share them.

When there is a need to customize an Open Standard in answer to the needs of particular communities that have common requirements concerning applications, an application profile that is more suitable to a community has to be designed. Its must also guarantee the respect of standards and interoperability with others.

Let us look at examples regarding e-learning contents.

1.1 Examples

1.1.1 LOM

LOM (Learning Object Metadata) is an IEEE standard that deals with metadata for educational contents in order to reference, share and search for learning resources. It is a typical example of Open Standard. To adapt the standards to their needs, different organizations propose their own metadata schema: CanCore (Canada), Celebrate (European Schoolnet Community), FORMIST (ENSSIB - National School of the Sciences of Information and Libraries), Manual (C@mpuSciences - French universities of the Academic Network of Self Learning Centers - RUCA), Normetic (Quebec), TLF - the Learning Federation application profiles (Australia), UK LOM Core (Cetis - England) and others. The invoked reasons are often the need to add new metadata in order to provide information unsupported by LOM or to enrich a list of terms associated to a metadata element [4].

1.1.2 IMS-QTI

IMS-QTI is a specification of the IMS Consortium. It aims to ensure the portability and the migration, from a Learning Management System (LMS) to another, of assessments, quiz exams and learners' answers. The generation of exercises with respect to this specification will permit to create databases of questions of varied sources and to use them in any LCMS (Learning Content Management System) compatible with this specification. For

this specification, there also exist extensions to answer specific needs due to the context. We can mention, for example, MathQTI that permits to present the mathematical expressions [5], or the proposition of Auzende that permits to create mathematical exercises in which the variables are bound by constraints [6].

1.1.3 SCORM, IMS Content Packaging and IMS Common Cartridge

SCORM, IMS Content Packaging and IMS Common Cartridge are concerned by the technical portability of educational modules. Their adoption facilitates the reuse of an educational module in different LMSs. The modules created in accordance with these specifications can be spread out and executed automatically on all compatible LMSs with these specifications. There again, needs of personalization exist, and other profiles are created as shown in [7].

1.2 Analysis and Comments

The existence of criticisms, suggestions of adaptation and improvements for Open Standards proves that using standards without adapting them to the local context is difficult.

The requests formulated by the users can be defined by three categories. *Requests to restrain* the number of attributes to be filled or requests to reduce the value set. This is generally the case when a Open Standard includes a large number of data that the users feel difficult to fill or whose utility they do not see in their context. *Requests to increase* the attributes or their value set. Contrary to the previous category, this occurs when a Open Standard is not rich enough to cover the needs of the context, when it does not offer possibilities for satisfactory extension or when the vocabulary is not sufficient to express needs with enough details. *Requests to clarify* the documentation and the users' guidelines. They are due to the ambiguousness of the definitions assigned to the data, to the absence of explanation on their utility or to the lack of examples on how to provide or use information.

To satisfy all these requests there are three possibilities:

1. to ask for modifications of the Open Standard in order to produce a new version. In the case of LOM, the work is still in progress. The ISO organization is currently working on the MLR (Metadata Learning Resources) norm. The goals of this work are to adapt, to correct, to amend or to improve the technical aspects of the LOM and to propose a norm [8]. It was also the case of QTI and its new version 2.1 (June 2006). The problem is that changing the specifications is very time and effort consuming. It is also costly for the companies which develop software that respect these standards.
2. to use a private document and to develop a gateway tool that realizes a conversion toward the Open Standard. This is less expensive but it requires that the degree of similarity between the private document and the Open Standard be good enough. The main drawback is that, if the standard changes, it becomes necessary to change the gateway.
3. to create an application profile compatible with the Open Standard. This way is more and more preferred because it guarantees a certain compatibility with the norm and therefore facilitates the passage from a profile to another. This is what we have worked on. We propose tools that make it easier for the profile designer to define an application profile compatible with a norm, standard or specification.

2. From an Open Standard to an Application Profile

Before presenting the formalism that we have elaborated to describe an Open Standard, we first give a global view here of what the process from an Open Standard to an application profile is.

According to the glossary of the Dublin Core [9], an application profile is a "set of elements of metadata, policies and leading lines defined for a given application". For Heery, an application profile allows to compose a new schema by combination of metadata from different standards [10]. One can (i) add some complementary metadata necessary to the context, (ii) affect an optional or obligatory property to the fields of the metadata or (iii) modify the vocabulary associated to the definition of an element in order to adapt it to the particular context of the users. Moreover, to be complete, an application profile must contain a documentation that defines the policies and some exemplary practices. Finally, for Duval, "the purpose of an application profile is to adapt or combine existing schemas into a package that is tailored to the functional requirements of a particular application, while retaining interoperability with the original base schemas" [11].

From these definitions, we can say that an application profile adapts an Open Standard to a particular context. A question arises: How to create correct application profiles?

To create new application profiles from Open Standards, Friesen and all suggest, from their experience of CanCore, a set of rules and good practices to create a new application profile [12]. But they do not propose a formalism to support the implementation of the application profile.

As seen, the concept of application profile is largely used in the field of the description of resources and metadata. As our purpose is larger, we prefer to use only the term "profile", which is more general and which means that it is used in different contexts and domains, which are described in a Open Standard. Moreover, it allows introducing access control and presentation regarding the users' groups.

In this paper, to facilitate this implementation we present a formalism in order to specify the characteristics of the profile and to support information for its contextualisation.

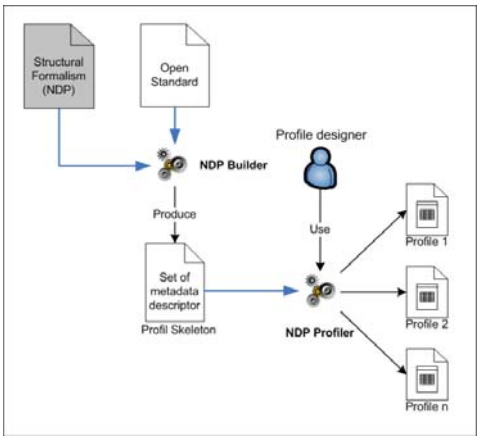

Fig. 1. From Open Standard to Application Profiles

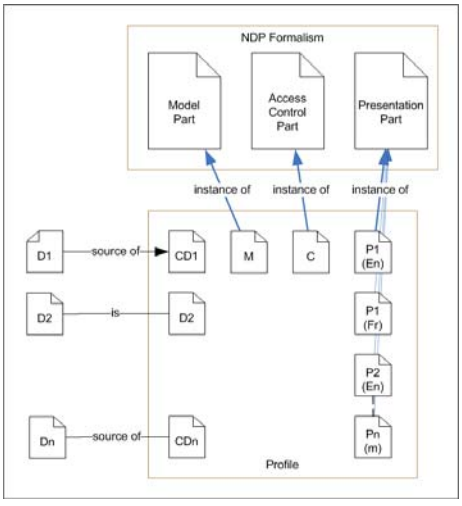

Fig. 2. NPD Formalism and profile components

The NDP Formalism describes the structure to produce in order to store the metadata descriptors. The *NDP builder* uses this formalism to generate a skeleton profile of the Open Standard. This skeleton contains the set of metadata descriptors. Thereafter a profile designer uses the *NDP Profiler* to specify properties for the contextualisation in the descriptors. Thus, it is possible to create for one Open Standard different profiles respecting the same formalism.

3. Formalism Description

The NDP Formalism, in addition to the Document (D) instance of the Open Standard, is composed of three parts: a "Model" (M) that completes the Open Standard with specific information about the context, an "Access Control" (C) that specifies the rules of interaction between the elements defined in the Open Standard and the users, and finally a "Presentation" (P) that permits to determine how the elements of the Open Standard must be presented to the users.

Therefore a profile, associated to a Open Standard, can be defined by a quadruplet (D, M, C, P), where D is present at least once, M is optional, C is optional and P can be repeated (Figure 2). If a profile is based on several Open Standards then all of them must be present in the profile. Some can be included as such, others be customized (as in Figure 2). The information of each part as well as its role is presented below. This information is exploited to specialize the elements of the Open Standard. It is usable by any tool that manipulates the Open Standard (creation, indexing, research, consulting, import, export, etc.).

3.1 Document

The NDP Formalism only applies to Open Standards that have elements organized according to a hierarchical structure as the one defined by the W3C XML Schema [13]. This is not really a restriction as it is generally the case.

Usually, the Open Standard defines two parts. On the one hand, the structure described with a controlled vocabulary that names the elements and gives the constraints which apply to their content and on the other hand, the links between the elements.

Among the information, some, that we call "describers", identify or characterize every element. There are four compulsory describers: the *name*, the *data type*, the *minimal* and the *maximal* occurrences of each element. Two optional describers, an *identifier* (other than the name) and a *short description*, can be added.

Creating a profile from a Open Standard consists, partially, in modifying these describers and, by so doing, in creating a customized document (CD). Two types of modifications must be distinguished: the restraining modifications and the modifications that change the hierarchical structure.

In the first case, a full compatibility with the Open Standard is ensured. For example, it is the case of some changes as the type of data (size of 8 characters instead of 20) or as the cardinality of an element (minimal or maximal number of occurrence) when these new cardinalities remain included in the interval defined by the Open Standard.

In the second case, the modifications no longer guarantee the full compatibility of the customized document with the Open Standard. In the NDP Formalism, we suggest to ensure full compatibility with the Open Standard.

As both the Open Standard and the customized document are close regarding their structure and contents, we will henceforth refer to it as Document (D).

3.2 Model

The "Model" part allows adding, to every element, five describers in addition to the previous describers. Their role is either to complete the information, or to impose supplementary constraints on the content of the elements.

The *stability* describer has merely an informative role. It allows classifying the elements according to the frequency of their modification. The "fixed" value indicates that the element will only be informed only one time. The "weak volatility" value indicates that the content can change but that it seldom does. Finally, the "strong volatility" value indicates that the content often changes. This describer is useful to attract the attention of the users on the fact that there can (or cannot) be some modifications here.

The *value set* describer concerns content elements only (i.e. the leaves of the hierarchical structure of the document). It allows adding a constraint to their content and, if necessary, associating a value set. In this describer, the "free" value indicates that there are no restrictions on the content of the element. The "list" value indicates that the user has to select a term in a list to fill the element. Finally the "combo-box" value indicates that an expandable list of terms is associated to the element. The user has the possibility to insert a new value. In this case, it is a proposition of extension of the list that can be addressed to the application profile manager in order to validate and include it. The existence of a rigorous system of validation and control is necessary to ensure a good management of the list.

The *vocabulary* describer becomes compulsory when a list is associated to a content element. It contains the identifier of the list. It can reference an ontology entry, a taxonomy, a simple value set or a hierarchical list.

The *ordering* describer concerns the elements having a minimal number of occurrences superior to one and permits to impose an order in the values.

The *default value* describer only concerns content elements. It allows associating, by default, a value or one term in a list.

3.3 Access Control

The "Access Control" part allows configuring the user-software interaction. It defines the rights and the duties of every users' group. It determines the element that each user can reach and what are the means at his/her disposal to fill it. To do that, it is necessary to declare three entities: users' groups, controllers and filters. Then, it is necessary to establish associations between these elements.

A users' group defines the users that have the same rights and duties. It is defined by a name and a description indicating their role with regard to the document. Different criteria (status, specialty, age, etc.) can be chosen by the Profile Designer to classify the users belonging to a group.

A filter is a tool that establishes the rules of access and interactions between the user and the application. For every element, it allows specifying two describers:

- The *display* describer makes it possible to know if the filter must present or not the element to the user. When an element is not displayed, by inheritance, all its descendants are not displayed either. The objective of this describer is to reduce the cognitive load of a users' group.

- The *requirement* describer can either play an informative role, or add supplementary constraints. The "compulsory" value imposes to the user to assign a value to the element. The "optional" value indicates that the user is free to fill or not that element. The "recommended" value, on the other hand, suggests the information without imposing it. Finally, the "fixed" value forbids modifying it.

A controller contains a set of filters and defines the one that must be used by default for the users who do not belong to any group.

Once the users' groups, the filters and the controllers are defined, it is possible to associate to every users' group, the filter to be used for every controller.

3.4 Presentation

The "Presentation" part provides the necessary information for building a users' interface. Its role is to build a customized view of the document for every users' group. Three entities are configurable: the elements, the vocabulary and the style.
For the elements, the "Presentation" part adds eleven describers. Five describers for the display are supposed to testify whether or not the construction of the Graphical User Interface (GUI) is necessary. These describers are: *icon, label, label ToolTip, data ToolTip, status Bar*. Six describers for the help are destined to provide a description and a support about the elements in order to help the users understand their role. These describers are: *definition, extended Definition, explanation, examples* that contains examples of values, *help* and *supplementary Info Link*.
For the vocabulary, it is possible to associate a label to every term declared in the "Model" part that, let us remind it, presents a concept of an ontology or a list of values. This association allows, when displaying it, to substitute the term by its label. Thus, instead of using an abstract term, that may be incomprehensible to the user, it is possible to personalize the content display of an element by using a terminology more usual in the users' environment. This substitution allows displaying the concept in different languages and using the same concept, even though it is named differently. Finally, it is possible to define styles to configure the properties used to display the elements (size, police, color, border…) and then to associate theses styles with the users' groups or elements' relevance.

4. Implementation

Since the majority of the Open Standards is created by means of the XML Schema [13], we chose to use XML for implementing the NDP Formalism. This choice allows us to create the different NDP Formalism parts (Model, Access Control and Presentation) through DTDs or Schemas. They are accessible on the website [14]. On the same web site, a generator analyzes a XSD schema to generate the skeletons of the three instances (M, C, P). Then, by filling them, the profile designer obtains the profile.

We experimented the NDP Formalism on three examples by using the generator that we developed. First, we created a frame in order to describe Dublin Core metadata schema. Then, we created the French profile LOM-FR [15] based on LOM standard [16]. Finally, we created an application profile from LSCM schema (Learning Software Component Metadata) which it uses to describe software components [17]. More voluminous is the Open Standard, more time consuming is the profiling process. That is for this reason; we developed the profile builder application that accelerates this operation while automatically generating the skeletons used to contain the profile information.

5. Conclusion

Nowadays, the adaptation of norms or standards (Open Standards) to context needs is a more and more common practice. In this article, we proposed a NPD Formalism to create profiles from Open Standards. This Formalism is destined to profile designers who wish to use a norm or a standard but want to customize it to their specific needs, as is the case for creators of application profiles on metadata (LOM, IMS-QTI…).

To illustrate our proposition, we hereby provide, in [14], an example of the French application profile of LOM (LOM-FR) [15]. This application profile has been chosen to be fully compatible with DublinCore [18]. This norm consists of 15 metadata elements. Some

of them are already included in LOM, but not all of them. The French application profile of LOM therefore includes elements drawn from DublinCore. The example presents the case of the "DC.*type*" metadata element (used to describe the type of the content of a resource: picture, text, etc.) and the case of the “Standard Learning Re-source” metadata element (5.2) inherited from LOM.

Using the NDP Formalism facilitates the creation of profiles because it separates context information in three distinct parts (Model, Access Control and Presentation) and provides an adequate structure for every part to inform the profile specificities.

Having such a formalism permits to describe profiles with the same structure (D, M, C, P). It permits to design generic data processing systems able to take in charge these profiles and to adapt the exploitation of document descriptions to the users’ needs while respecting norms. This Formalism facilitates the interoperability between different profiles of a same Open Standard since it is possible to establish some correspondences.

To conclude, if the definition of a profile is a long process, the NDP Formalism is a quite interesting solution as it includes in structured documents the required data to implement the profile moreover usable by any tool that manipulates the Open Standard (creation, indexing, research, consulting, import, export, etc.). With the "Presentation" part, an interface generator is able to design a GUI (Graphical User Interface) adapted to the needs of specific users' groups.

References

[1] Sutter E. (1992). An international classification for standards, Paris-France, ISBN: 0012-4508

[2] Information technology – Business agreement semantic description techniques: Part 1, ISO/IEC 15944-1 (2002)

[3] Friesen N., McGreal R. (2002). International E-Learning specifications, The International Review of Research in Open and Distance Learning, Vol.3, No2, ISSN :1492-3831

[4] de La Passardière B., Jarraud P. (2004). ManUeL, un profil d'application du LOM pour C@mpuSciences, Revue STICEF, Vol. 11, ISBN: 1764-7223, 11-57

[5] MathQTI, Specification for Mathematical Questions & Test Interoperability, http://www.maths.ed.ac.uk/mathqti/

[6] Auzende O., Giroire H., Le Calvez F. (2007). Extension of IMS-QTI to express constraints on template variables in mathematics exercises, In International Conference on Artificial Intelligence in EDucation – AIED, 524-526

[7] Morimoto Y., Ueno M., Shibata S., Yokoyama S. Miyadera Y. (2006). SCORM Extension for Describing Learning State Transitions Based on Learner's Actions, IEIC Technical Report (Institute of Electronics, Information and Communication Engineers), ISSN: 0913-5685, Vol.105, n°.581, 25-30

[8] Bourda Y., Delestre N. (2005). Améliorer l'interopérabilité des profils d'application du LOM, Revue STICEF, Vol. 12, ISSN: 1764-7223, http://sticef.org

[9] Dublin Core glossary, http://dublincore.org/documents/usageguide/glossary.shtml

[10] Heery R., Patel M. (2000). Application profiles: mixing and matching metadata schemas, Ariadne Issue 25, http://www.ariadne.ac.uk/issue25/app-profiles/

[11] Duval E., Hodgins W., Sutton S., Weibel S. (2002). Metadata Principles and Practicalities D., D-Lib Magazanie, Vol 8, n°4, , ISSN: 1082-9873

[12] Friesen N., Mason J., Ward N. (2002). Building Educational Metadata Application Profiles, Proc. of Int Conf on Dublin Core and metadata for e-communities, Firenze University Press, 63-69

[13] W3C XML schema, http://www.w3.org/XML/Schema

[14] NDP, http://www.math-info.univ-paris5.fr/~rebi/programmes/ndp.php

[15] LOMFR (2006). profil français d'application du LOM, AFNOR, n° NF Z76-040

[16] LOM, Final Draft Standard for Learning Object Metadata, Approved draft, Document IEEE 1484.12.1-2002, 2002

[17] Rebaï I. (2006) Conception et développement d'une plate-forme permettant la capitalisation de composants logiciels pour les environnements d'apprentissage humain, Ph-D, University René Descartes.

[18] Dublin Core Metadata Initiative, http://dublincore.org/schemas/xmls/