QBLS: A Semantic Web Based Learning System

Sylvain DEHORS
ACACIA, INRIA Sophia Antipolis,
Sophia Antipolis, France
Sylvain.Dehors@sophia.inria.fr

Catherine FARON-ZUCKER
MAINLINE, I3S & Computer Science dpt. of the EPU School
University of Nice Sophia Antipolis
Sophia Antipolis, France
faron@essi.fr

Abstract: In this paper we present the QBLS system, a www-based intelligent learning system that completely relies on semantic web standards and technologies. It reuses a large coherent set of learning resources taken from the web and has been experienced as an online support system for lab sessions of Java programming at the Computer Science department of the EPU engineering school of Sophia Antipolis. In this work, we highlight an implementation of semantic web design principles where organization and retrieval of learning resources rely upon ontologies describing the learned domain, the pedagogical strategy, and user profiles. Resource collection is based on the reuse of pedagogical materials going through a semi-automatic annotation process. We describe here our ontological choices as well as the method followed to annotate. Finally we recount the deployment of the system, coupled with a wiki, offering web-based course support.

1 Introduction

The semantic web (Berners Lee et al. 2001) aims at exchanging information that can be understood both by humans and machines. Standard formalisms like RDF (RDF 2006) and OWL (OWL 2006) are proposed as a basis for exchanging not only data, but knowledge. The field of "computerized" or "intelligent" learning systems is often described as very promising for the application of the Semantic Web (Devedzic 2004). In systems targeting the management of content (LCMS) or direct tutoring (LMS) the representation of knowledge in somehow formal ways, has been practiced for a long time. A now well spread vision in the e-learning community is to automate reusing existing resources coming from different sources (Duval et al. 2001). Resources must be previously described, possibly using semantic web formalisms (Nejdl et al. 2002). Despite its appeal, this approach falls into classical interoperability problems faced in previous research on Learning Objects. Experience proved that interoperability of Learning Objects at a fine level raises lots of unsolved issues (Willey 2003). However the Semantic Web has still much to offer to learning practitioners: it provides expressiveness and inference mechanisms to reproduce many existing results achieved in closed tutoring systems, and it allows combining them into a single or distributed web based application. One of the direct gains obtained from this technology lies in the reuse of existing generic tools and models. Demonstrating this principle we have developed the QBLS system, a www-based intelligent learning system that completely relies on semantic web standards and technologies. QBLS stands for Question-Based Learning System, its objective is to give access to on-line resources while performing assignments (or answering questions). With the presentation of this pedagogical tool, we propose a framework and a methodology to effectively reuse a large coherent set of learning resources taken from the web. Our contribution is not theoretical but rather deeply rooted and motivated by higher education teachers' needs and the overall approach has been experienced as a www-based course support system for lab sessions of Java programming at the Computer Science department of the EPU engineering school of Sophia Antipolis.
We present first an overview of the system, highlighting its Semantic Web design principles. In this system, organization and retrieval of learning resources rely upon ontologies describing the domain to learn, the pedagogical strategy, and user profiles. The resource collection process relies on the reuse of pedagogical materials through semi-automatic annotation of the original content. Finally we recount its deployment coupled with a wiki server as a www-based course support system and present significant figures about the impact of the system.

2 Overview of QBLS

2.1 An Ontology-based Design

The semantic web approach, applied to the design of a learning system, targets the reuse of conceptual models (more specifically ontologies) to guide learning resources retrieval, to organise them and to provide user adaptation involving ontological reasoning (Stojanovic et al. 2001). We look here at how this can be achieved in realistic situations. Classical "knowledge models" used in adaptive and intelligent learning systems (Brusilovksy 2003) are defined as sets of "concepts that form a knowledge space" in other terms they are ontologies as "conceptual representation of a domain" (Gruber 1995). The graph representations, often used in the literature on learning systems, find a handy expression in RDF and the needs, in term of expressivity, for describing such models are largely covered by languages of the OWL family. To present and justify our ontological choices we follow a standard distinction in e-learning, where two complementary aspects are modelled: the domain and the pedagogy (Mizogushi & Bourdeau 2000).

Domain model: When tackling real world usage, it appears that the cost of authoring detailed domain ontologies in OWL is tremendous. Moreover this is outside the competency of a regular teaching staff. In most prominent examples of applications (Brusilovsky 2003), "knowledge models" are networks of concepts connected by binary relations, possessing a loosely defined hierarchical semantic. Thus the intermediate representation, Simple Knowledge Organisation System (SKOS 2005) proposed by W3C, can perfectly express such a model. It defines the relationships "broader" and "related" between concepts, and "subject" and "primarySubject" to link resources with concepts. The model also proposes some inferences, like transitivity along "broader" links. More complex representations seem too rich and hard to understand for a teacher (considered a non-expert in knowledge modelling). Fig. 1 shows how resources about Java programming are described by domain concepts expressed using SKOS.

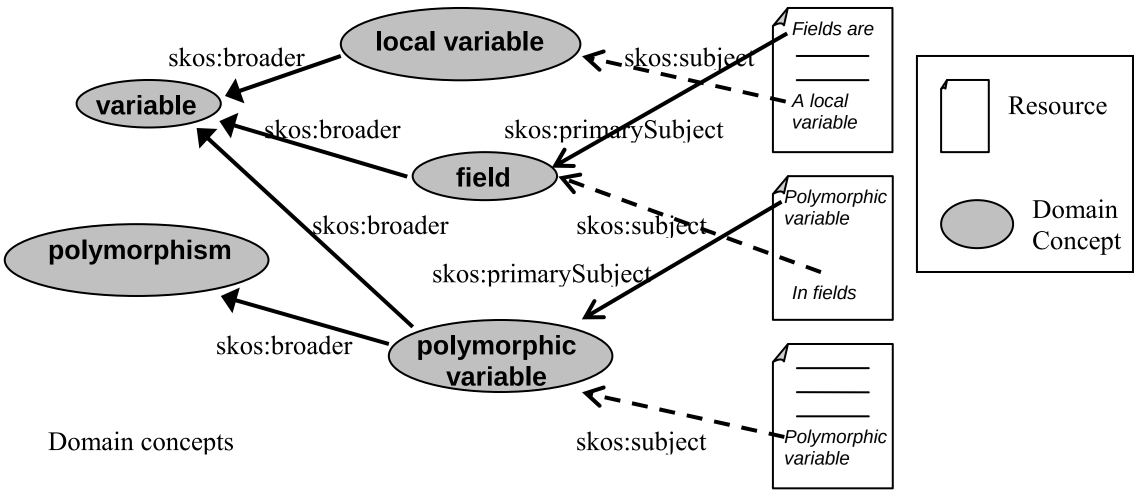

Fig. 1 – Resources annotated by domain concepts using SKOS

Pedagogical model: The major goal of learning systems is to support a given pedagogical strategy. In this scope pedagogical ontologies can be associated with reasoning mechanisms and rules to enforce a given strategy. Often this strategy consists in selecting or computing a specific path among the resources. Thus, formal semantics are required here to enable such computation. For example

Fig. 2 shows how resources (representing XHTML files) instanciate OWL classes in our experiment. Pedagogical rules are also part of this model, but they are still to be standardised for the semantic web.

By the light of the experiment we consider that defining a formal pedagogical ontology is worth the effort as opposed to defining one for the domain. In fact the chances of reusing this ontology are higher. Pedagogical ontologies are often based on a broad learning theory (like bloom's taxonomy) which increases their generality. Moreover the number of concepts dealt with is rather small. For example the pedagogical ontology proposed by (Ullrich 2004) possesses 26 concepts, like "Definition", "Example", etc. ,whereas in a small vocabulary for introducing the Java domain, we had to define 171 concepts. Other experiments on Java rely on domain ontologies that define up to 630 concepts. To be used for resource selection, path planning, etc. "pedagogical" rules must be introduced (e.g. "a definition must be presented before an example"). These rules are not domain related and their number is kept to a minimum to limit teacher's overload. Those pedagogical aspects are often static in classical system, because interactions supporting the pedagogy have to be hard

coded in some way. Using ontologies offers a little more freedom, and customisation possibilities. Moreover it revealed a big gain for deployment time, as the formalisms and tools already existed.
The design of QBLS is based on these principles. Learning resources (slides) are semantically annotated using RDF statements that link resources to domain and pedagogical concepts.

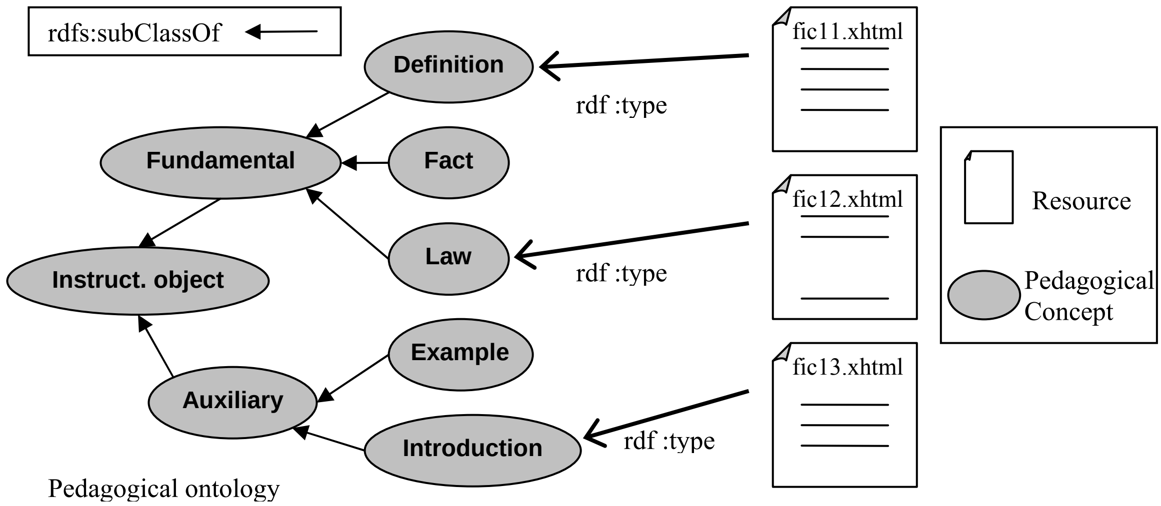

Fig. 2 –Instantiation of pedagogical concepts by resources

Resources can be retrieved based on the domain ontology: they are linked to domain concepts through subject relationships as explained above and a semantic search engine, Corese (Corby et al. 2000), retrieves all the resources for a given concept (and its "subconcepts" by transitivity of the "broader" link). Resources are also annotated with respect to the pedagogical ontology: they are typed according to pedagogical concepts ("Definition", "Law", "Fact", "Example", etc.). When resources are retrieved from the resource base, they are "ranked" according to these types, e.g. definitions have priority upon laws, laws upon facts, facts upon examples, etc.
Among the systems that could potentially follow this semantic web "pattern", adaptive systems would need an additional model and data which is the user model and user history. Dynamic adaptation is often performed based on *overlay models* (De Bra et al. 2002). The RDF model perfectly supports this, through semantic annotation of the knowledge space with specific user modelling relationships. In QBLS, adaptation is based on such a user overlay model, keeping track of the user activity stored in RDF statements.

2.2 Semi-Automatic Annotation of Learning Resources

Defining the knowledge model is only one part in the deployment of a learning system. Identifying the corresponding knowledge in the resources is, in itself, a major research issue as manual annotation reveals to be a highly time consuming task. When addressing large realistic amounts of resources as we did (see section 3.), solutions must be found to ease the process. We have observed that generic annotation tools, like (Annotea 2006) are too rich and complex to really help. Instead, we propose an annotation process based on the exploitation of the layout features of the resource contents (words in bold, italics, titles, etc.). Here are several arguments motivating this choice:

- Part of the work is already done, in an informal way, in the resources layout. With resources of good quality, styles have semantics that are enforced throughout the content. This allows for direct automatic extraction of a noticeable part of the knowledge.
- Editors like OpenOffice Writer (or Microsoft Word) provide interesting features like styles hierarchies to enforce standardisation all along one or several documents. They are easy to handle, as teachers are used to manipulate them, and they offer the possibility to fully modify the content, something that no small tool can perform.
- In a learning scenario, semantic mismatch between the presentation and the content are unacceptable. They would disturb learners and remove any pedagogical benefit brought by the system. Thus when applying automatic annotation techniques, a validation phase must be performed to ensure that no errors have been introduced. Visual validation of automatic annotations is much more comfortable and easy to realise in this case.

— Finally, studies (Brusilovsky 2003) proved that form-based annotation (annotating by filling forms) was not acceptable on large corpora. Whereas mark-up based seems to be more scalable. The technique we adopted is in fact a mark-up based one, but it mainly relies on a classical widely available editor.

We applied this method to annotate with OpenOffice Writer[1] a Java course available on the web. The course was originally in Microsoft Power Point (.ppt) format and had to undergo a little pre-processing. The material was automatically enriched by matching the terms of both a domain ontology and a pedagogical ontology, and setting corresponding text and paragraph styles. Titles were automatically extracted based on their paragraph style. Terms expressing concepts were identified based on the domain ontology. The annotation process consists in assigning a paragraph style to each resource - indicating a pedagogical type - and a text style to each word in the resource content identifying a domain concept this resource is related to. The domain ontology was reused from the web and enriched for missed terms throughout the annotation process. The pedagogical ontology was reused without modification (Ullrich, 2004).

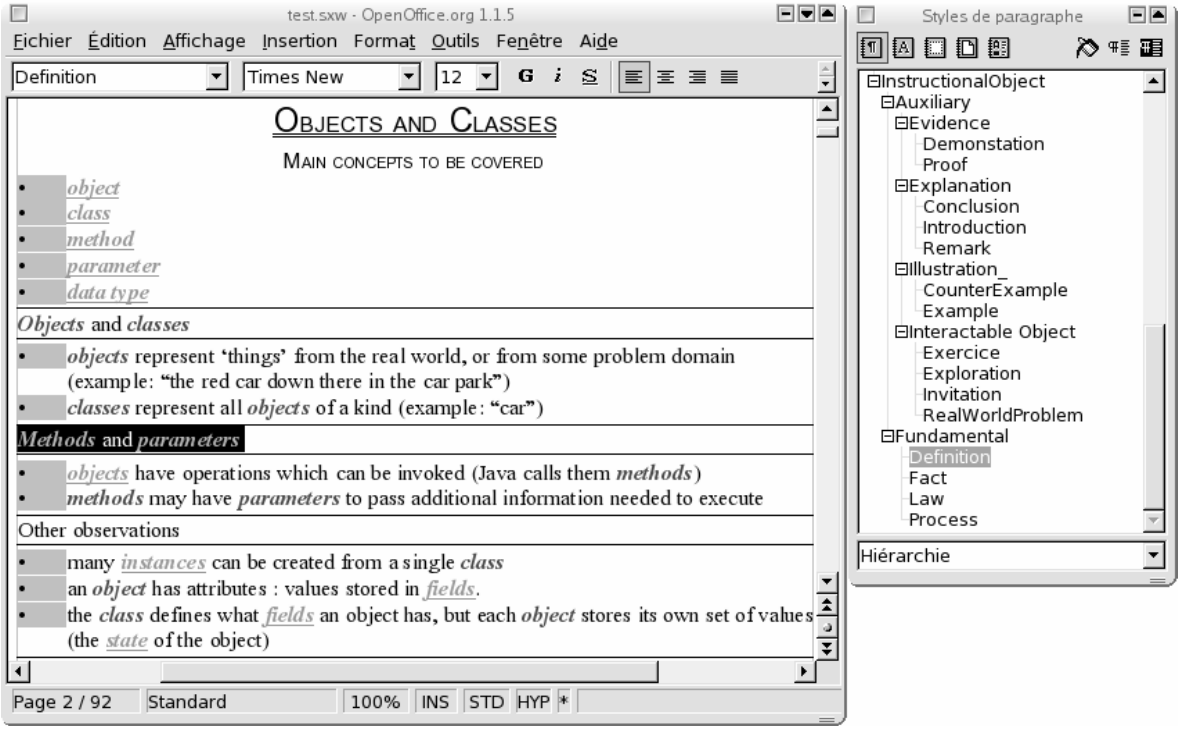

Fig. 3 – Annotation of learning resources with OpenOffice Writer and its style hierarchy

Fig. 3 presents a screenshot of the OpenOffice Writer window where three resources are displayed, separated by horizontal lines. The second resource from the top is currently being annotated by selecting the style "Definition", and applying it to the title "Methods and parameters". The text has already been automatically annotated by setting special styles on the words "methods" and "parameter" which correspond to concepts of the domain ontology. The word "object" has been manually annotated by selecting a special text style indicating a reference to a different concept that is not the primary subject of this resource. As a result, the "annotations" in their graphical form state: this resource is a definition for both the domain concepts "method" and "parameter" and it is related to the domain concept "object".

Once annotated, the OpenOffice file is processed through different XSL transformations that produce on one hand, a set of XHTML resources contributing to the pool of resources, and on the other hand RDF statements that contain the extracted annotations. In section 3 we present how this information is exploited in practice.

2.3 Architecture

Integrating available semantic web tools within the application drastically reduces the effort to develop such an e-learning system. For instance QBLS has been designed around the semantic search engine Corese (Corby et al. 2004) relieving the authors of any development effort dealing with the knowledge manipulation aspects. This is coupled with W3C XML-based transformations (XSL, XML) that naturally handle the generation of static or adaptive HTML interfaces. The flexibility of this approach enabled us to replicate and adapt the original code (JSP/XSLT technology) in a fast and efficient manner to deploy now four different applications of this system.

[1] http://www.openoffice.org

One of the long run objectives of the semantic web is semantic interoperability between systems. We believe the QBLS system is one good example of it. As it uses standard formalisms, and respects the semantics of the standard models (SKOS, OWL, RDF), it can be coupled with other semantic web applications. Technical interoperability is supported by standard web technology, while semantic communications use shared ontologies and annotations relying on semantic web formalisms.

The pedagogical value of the association with other tools is to provide different learning activities. For example it can be coupled with an on-line interactive exercise tool to help students accessing the content of the course while doing adaptive assignment. Other systems can also integrate the URL of QBLS directly in their internal mechanisms. For example for the Advanced Semantic Platform for Learning developed in the context of the Knowledge Web NoE., the domain ontology is shared between the different tools, including Magpie (Dzbor et al. 2005). Learners browse HTML content and access the QBLS service for spotted domain concepts. For each term highlighted in the Magpie semantic browser, a contextual menu proposes a range of available services. Selecting the QBLS service entry opens a second window where resources associated to the highlighted term can be browsed.

3 Deployment of QBLS at the EPU Engineering School

3.1 Educational Setting

QBLS has been used as a www-based course-support system in two different experiments at the engineering school of Sophia Antipolis to support lab sessions. We first experienced it during "lab sessions" supported by QBLS, those were 2 hours assignment sessions on signal analysis (Dehors et al. 2005). It is now supporting lab sessions of introductory teaching of the Java programming language for a semester long. This teaching consists of bi to tri-weekly lab sessions (of 2 hours) during 14 weeks. The course involves 84 students in first year of engineering curriculum. Labs use the BlueJ programming environment and are collaboratively written by teachers using a wiki. During the writing process, teachers collaboratively add links towards QBLS using a macro. The syntax of the macro in the wiki page is presented below. Such a syntax simplifies the task of adding links, an activity that can be described as annotating the assignment content with domain concepts. The first argument identifies the URI of the concept and the second indicates the string that appears in the wiki page and materialises the link:

[[Access(prog:Method, "see Method")]]

The course content was entirely reused from slides publicly available on the BlueJ web site. This course contains a total of 443 slides constituting the pool of resources. Our experience is thus an example of a real attempt at reusing external resources from the web. Fig. 4 shows the QBLS interface (the small frame) displaying five resources (two definitions and three examples) found about the concept of "method" in Java with their associated pedagogical type and title.

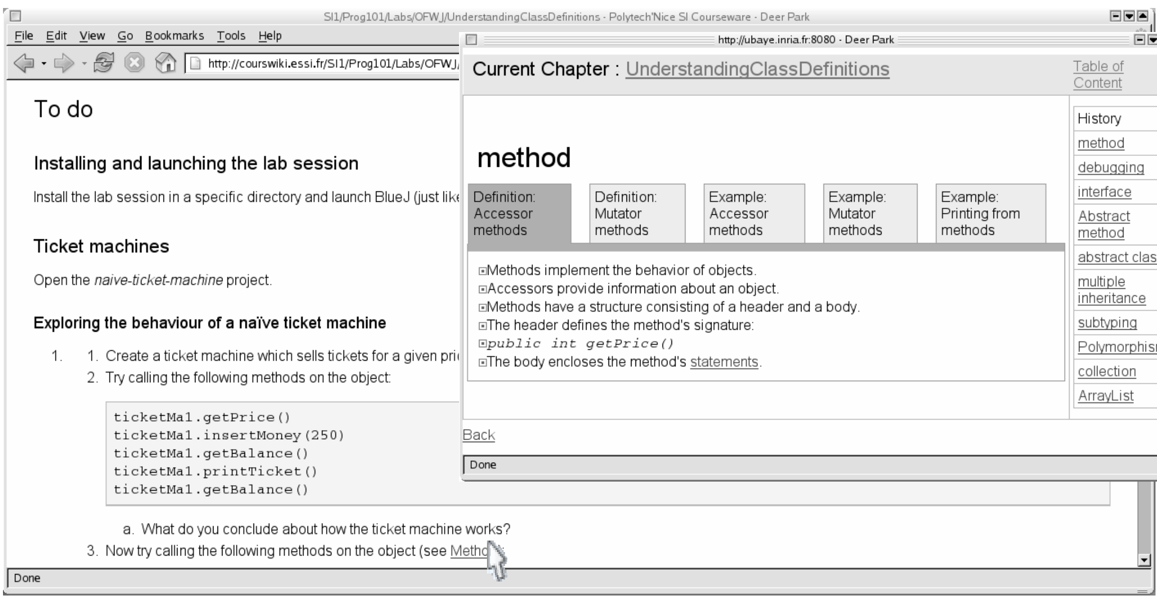

Fig. 4 – ScreenShoot of the QBLS interface accessed through a wiki page

Ontologies used in the system also originated from the web. For the domain we only extracted a list of candidate concepts from an existing experiment (Henze 2005). A specific vocabulary for the domain of Java was then incrementally built throughout the annotation process. Only 11 types out of 25 available in the pedagogical ontology have been actually used by the teacher to annotate the resources. The most intensively used indicate the pedagogical orientation of the content: Definition, Law and Examples. This is a quite natural result for an introductory course and thus comforts the assessment of this process. Regarding the domain ontology, 95 concepts out of 171 have been used. Among the 95 concepts around a half of them was not present initially in the original reused vocabulary. This last result shows the gap existing between the vision of a given teacher annotating the resources and a domain vocabulary supposedly focusing on the same topic and reused from the web.

3.2 Experimental Findings

3.2.1 usage feedback

The activity on the system has been recorded throughout the 4 months of the experiment. Students were identified by their wiki login. This enabled us to precisely track their activity for analysis. In addition, they answered a questionnaire for a user centred evaluation.
Usage: Only part of the material has actually been seen by students. But coverage reaches 80%, a first encouraging result as one of the main problem before was poor usage of the available material. It also shows the "utilitarian" behaviour we expected from students: they only browse what is relevant for their current focused problem. Having them going through all the material would imply some kind of restriction in navigation freedom, which is opposed to the applied pedagogic paradigm. We also recorded that in spite of not being compulsory, a majority of students have used QBLS and a third of them have accessed more than 10 pedagogical resources. We consider this group as effective users of the system. This figure shows the interest of the students for that kind of access to knowledge even if classical slides shows and paper handouts are available. In addition to connexions during lab sessions, we recorded 40% of the activity outside lab hours. Let us note that some students regretted to be unable to access the system from home when they didn't have internet connexion, revealing that this figure could have been even higher, and will certainly increase in a near future. This emphasises that even when having face to face courses in a regular location, distant on-line learning, especially from home, is now playing an important role in academic settings.
Evaluation: Globally, students positively evaluated QBLS. In particular most of them declared that they could perform lab sessions without any paper course support and, for a majority, alone at home. From the "performance" point of view, we notice no correlation between the use of the system and the ability to perform well at tests or exams. The cognitive profiles of learners that are helped by the system do not match the categorisation brought by exam results. We understand this as a quite natural result. The impact of the system itself is better evaluated through user centred questionnaires and observation.
Focusing on the added value of the semantic web, students very positively rated the fact that they can easily access the relevant resources of the online course to perform exercises. On the contrary they rather negatively rated the usefulness of the tab headings indicating the pedagogical type of the resources (e.g. Definition, Law, Example, etc. see Fig. 4.). At first sight, this could mean that the added value of the pedagogical annotation is low. But looking at how they used that information we recorded that 50% of the time they didn't just visit the resources in order from left to right but made a deliberate choice. What we can conclude, is that annotations are of prime importance to organise resources according to a pedagogical progression. The display of that semantic information might not be relevant for everyone but its interpretation by the automated system is crucial. This also confirms the idea that the complexity of the model must be hidden for the end-user. Beginners, as we have observed them, cannot exploit the richness of this information at the early stages of their progression. They only benefit from it through the "intelligent" system that can handle this knowledge.
Some students did not realize that they were using different tools (wiki+QBLS). As a matter of fact the interfaces were quite similar in appearance. The seamless integration of the tools working over a distant connection proved efficient. The semantic technologies deployed required some engineering efforts, but considering the light weight, and customisable application developed (around a thousand lines of code for the "core" service), this is definitely an big step forward for implementation issues in intelligent learning systems. Finally, we also experienced that the complexity of the languages must remain hidden from both teachers and learners.

3.2.2 Annotation Process

Regarding the annotation process, it revealed itself quite efficient taking around 20 hours for the teacher to fully annotate the content. Let us note that she was quickly autonomous from the knowledge engineer in her annotation task. She could upload and download the annotated documents online to update the annotations alone at leisure. As we explained above, the domain model had to be completed in parallel to the annotation

task. We must concede that an interface for editing the domain vocabulary would be appreciated here to help manipulating this model. Thus the authoring of the domain model is somehow compulsory, but we have experienced that it can be handled if there is a strong connection between the modelling activity and its results in the user interface tested by the teacher/annotator. A general outcome is that teachers must understand rapidly, and "see" the outcome of their annotation activity in order to accept the workload and not entering the details of the knowledge models.
The use of a standard tool (Open Office) allowed the teacher to feel confident with the tool very quickly. We did not face any trouble with cross-platform issues, or file formats. Reused material (Power Point) was opened straight away in the Impress tool, which unfortunately does not use styles as efficiently as Writer. We developed a generic XSLT style-sheet that performs this transformation without major loss of information. The extraction of the information from XML and its expression in RDF is quite straight forward and does not require major engineering to be developed (250 lines of XSLT).
The major drawback of this approach is the necessity to develop a translation from the xml format of the editor (the same would apply for MS word), to XHTML, the only language accepted by the browser on the learner side. The embed transformation in the tools is not giving good results. Thus while the graphical power of those tools strongly argues in favour of their integration in the annotation process, current technological limitations makes it still difficult. However custom educational authoring tools cannot reasonably reach the quality of industrial standard editors, while this quality must be present in learning resources. We are confident that, in a near future, good exports to XHTML will be available, and for the time being we developed a generic transformation that can be reused.

4 Related Work

Using semantic Web technologies to offer conceptual navigation in a set of learning resources is an idea favoured in numerous research projects. We retain the following ones as the most relevant and interesting.
The Personal reader (Henze 2005) presents a service architecture relying on RDF and ontologies to exchange information about learning resources, the domain, and learners. The domain application also concerns Java courses, the ontology is expressed in RDFS and the system uses Triple for reasoning. Learning resources are existing web pages (no authoring involved). Main characteristics are that the domain ontology does not respect the semantics of the RDFS model, reducing the interest of this model for other experiments, and it does not tackle the problem of generating semantic annotations.
In (Winter & Greer 2005) some "Best practices" for student modelling are proposed using the semantic web. This contribution understands the problem of generic domain ontologies that do not fit a given course content. It recommends to use adapted SKOS vocabularies loosely coupled with the ontology to fit each course specifics. We match this advice on using SKOS, but the link with an existing generic ontology presupposes the existence of such ontologies. The authors also admit that efforts needed to follow those guidelines made them limit the number of course they annotated, showing the difficulty to rely on and use large domain ontologies.
Using classical editors to annotate and generate resources with semantic metadata is a problem the Alocom (Verbert et al. 2005) project tries to answer by providing plugins for Microsoft products. Resources are annotated with respect to an ontology that can map to existing standards like SCORM (SCORM 2006). By complying with the packaging standards, this approach is limited compared to our modelling and is more targeted toward reutilisation in commercial platforms complying with the standard.
Outside the field of semantic web, most learning systems use tailored courses. They require teachers to specifically create each document used by the system. Teachers are provided with authoring tools (Brusilovsky 2003) to create new documents, and most of them end-up generating proprietary formats (De Bra et al. 2003). However it seems that they could as well use interoperable languages of the semantic web. This would allow the integration of existing inference engines in those systems, and enhance the portability of their adaptation algorithms.

5 Conclusion

In this paper we have presented the QBLS system, an e-learning platform giving online access to a pool of annotated resources through a dynamic web interface. This work demonstrates the interest of semantic web technology in a practical setting. It answers the cost issue of developing web-adaptive systems by reusing existing tools. Benefits from this approach are huge while the constraints of using standards are very low. The expressivity of RDFS or OWL is largely sufficient for learning applications as researchers envision them today. For the semantic web, work still needs to be done on rules standards as rule-based mechanisms are essential in many e-learning applications. Our experience of reusing existing ontologies is quite conclusive concerning the pedagogical ontology, presenting quite abstract concepts. However it also outlines the difficulty (yet unsolved) of finding an acceptable match between different visions of a domain.

The proposed process of reusing existing material also proved to be efficient and brings a real answer to the tremendous effort usually required to author a fairly large amount of resources for a given system. Here not only the resources are reused, but they are also made available for everyone on the web, along with their annotations and associated models, all expressed in standard semantic web formalisms. However in our opinion this interoperability is more targeted towards bringing together different tools around the same educational setting rather than for exchanging and reusing learning objects across the web, which still poses many unsolved problems. In the future we plan to add more adaptation features to the system and start tackling this difficult problem of mixing resources from different sources. The problem of reducing further the burden of the annotation process will also need to be addressed. We propose to take advantage of even more existing information in the resources to achieve this "semantisation" of the content.

6 References

Annotea project (2006) http://www.w3.org/2001/Annotea/.

Brusilovsky P. (2003) Developing adaptive educational hypermedia systems: From design models to authoring tools. In: T. Murray, S. Blessing and S. Ainsworth (eds.): *Authoring Tools for Advanced Technology Learning Environment*. Dordrecht, pp. 377-409.

Berners-Lee, T., Hendler, J., Lassila, O. (2001) The Semantic Web. *Scientific American*, May.

De Bra, P., Aerts, A., Berden, B., De Lange, B., Rousseau, B., Santic, T., Smits, D., Stash, N. (2003) AHA! The Adaptive Hypermedia Architecture. *Proceedings of the ACM Hypertext Conference*, Nottingham, UK, August.

Corby, O., Dieng-Kuntz, R., Faron-Zucker, C., Gandon., F. (2005) Searching the Semantic Web: Approximate Query Processing based on Ontologies. Accepted for publication by *IEEE Intelligent Systems Journal*, to appear.

Corby, O., Dieng-Kuntz, R., Faron-Zucker, C. (2004) Querying the Semantic Web with the CORESE search engine. In R. Lopez de Mantaras and L. Saitta eds, in *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI'2004)*, subconference PAIS'2004, IOS Press, pp 705-709, Valencia, 22-27, August.

Dehors, S., Faron-Zucker, C., Giboin, A., Stromboni, J.P. (2005) Semi-automated Semantic Annotation of Learning Resources by Identifying Layout Features. In *International Workshop on Applications of Semantic Web Technologies for E-Learning*, AIED'2005, Amsterdam, July.

Devedzic, V. (2004) Education and The Semantic Web. *International Journal of Artificial Intelligence in Education (IJAIED)*, Vol.14, pp. 39-65.

Duval, E., Forte, E., Cardinaels, K., Verhoeven, B., Van Durm, R., Hendrikx, K., Wentland-Forte, M., Ebel, N., Macowicz, M., Warkentyne, K, Haenni, F. (2001) The ARIADNE Knowledge Pool System, *Communications of the ACM 44,* no. 5, pp. 73-78.

Dzbor, M., Motta, E., Stutt (2005) A. Achieving Higher-level Learning Through Adaptable Semantic Web Applications. *Intl. Journal of Knowledge and Learning*, Vol.1, No.1/2, Inderscience Enterprises Ltd., UK.

Gruber T.R. (1995) Toward principles for the design of ontologies used for knowledge sharing, *International Journal of Human Computer Studies*, pp.907-928.

Henze, N. (2005) Personal Readers: Personalized Learning Object Readers for the Semantic Web, In *Proceedings of 12th International Conference on Artificial Intelligence in Education (AIED'05)*, Amsterdam, The Netherlands, 18th - 22nd July.

Mizoguchi R., Bourdeau J. (2000) Using Ontological Engineering to Overcome AI-ED Problems. *Intl. J. of Artificial Intelligence in Education*, Vol.11, No.2, pp. 107-121.

Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmer, M., Risch, T. (2002) EDUTELLA: A P2P Networking Infrastructure Based on RDF. 11th International World Wide Web Conference (WWW2002), Hawaii, USA, May.

OWL (2006) Web Ontology Language, http://www.w3.org/2004/OWL/.

RDF (2006) Resource Description Framework, http://www.w3.org/RDF/.

SCORM specification (2006) The Advanced Learning Initiative, http://www.adlnet.org/.

SKOS Core Guide (2005) http://www.w3.org/TR/2005/WD-swbp-skos-core-guide-20050510/.

Stojanovic, L., Staab, S and Studer, R. (2001) ELearning Based on the Semantic Web, in *Proceedings of WebNet2001 - World Conference on the WWW and Internet*, Orlando, Florida, USA.

Ullrich., C. (2004) Description of an instructional ontology and its application in web services for education. In *Proceedings of Workshop on Applications of Semantic Web Technologies for E-learning (SW-EL'04)*, pp. 17-23, Hiroshima, Japan, November.

Verbert, K., Jovanovic, J., Gasevic, D., Duval E. and Meire, M. (2005) Towards a global component architecture for learning objects: a slide presentation framework, *World Conference on Educational Multimedia, Hypermedia & Telecommunications*, ED-MEDIA 2005, June 27 - July 2, Montreal, Canada.

Wiley, D.A.(2003) Learning Objects: Difficulties and Opportunities, http://wiley.ed.usu.edu/docs/lo_do.pdf.

Winter, M., Brooks, C., Greer., J. (2005) Towards Best Practices for Semantic Web Student Modelling, in *Proceedings of the 12th International Conference on Artificial Intelligence in Education (AIED'2005)*, Amsterdam, The Netherlands, July 18-22.