THE OMNIPAPER METADATA RDF/XML PROTOTYPE IMPLEMENTATION

TERESA SUSANA MENDES PEREIRA, ANA ALICE BAPTISTA

Department of Information System, School of Engineering, University of Minho, Portugal

Campus de Azurém, 4800-058, Guimarães, Portugal

{analice, tpereira} @dsi.uminho.pt

Abstract

Omnipaper (Smart Access to European Newspapers, IST-2001-32174) is a project from the European Commission IST program (Information Society Technologies) that investigates and proposes ways for access to different types of distributed information sources. This article intends to introduce the technology Resource Description Framework - RDF, developed by W3C for the Web based on metadata, and its practical use in the Omnipaper project, which the authors are involved. We intend to achieve the implementation of a prototype that enables users (professional journalists and occasional users) to have simultaneous and structured access to the articles of a large number of digital European news providers. Omnipaper is not a project about digitalization of news, but about bringing digitized news originating from various sources (and in various formats) together. In this article will be described the procedure implemented in the description of our newspaper articles using the RDF technology, followed by a elaborated description on the manipulation process and treatment of the information structured in RDF, through the RDF Gateway.

Keywords: Metadata, RDF**,** DC Web, Metadata.

INTRODUCTION

During the last decades, the amount of digital information has grown exponentially. The same holds for the number of computers and Internet connections. More and more information is becoming available in electronic form and its accessibility is, in terms of network presence, increasing rapidly. With this growth in availability, the need for information coupling has grown as well. Since it is physically becoming easier to compare information from geographically spread sources, the need for coupling information on a semantic level is on the rise [, 1999 #26].

One of the important challenges of the Web researchers resides in the need of organizing the immeasurable number of Web pages that appear everyday at every hour in the Internet.

The OmniPaper [2] project is investigating ways for drastically enhancing access to many different types of distributed information resources. The key objective of the OmniPaper project is the creation of a multilingual navigation and linking layer on top of distributed information resources in a self-learning environment thus providing a sophisticated approach to manage multinational news archives with strong semantic coupling, delivering to the user more than the sum of the individual service features.

One of the principal aspects of the project is the whole metadata layer of the system. Actually, there are two metadata layers: (1) the first layer (Local Knowledge Layer) that is added to the local archives where the main purpose is to provide a standard semantic description of all the existent articles in order to enable a structured and uniform access to the available distributed archives; and (2) a second layer (Overall Knowledge Layer), an higher abstraction level that will provide a common access user interface by integrating and relating the metadata information coming from the local knowledge layer. Furthermore, this interface is meant to be personalized and multilingual.

This paper explains the research work conducted in the RDF approach to the Omnipaper Local Knowledge Layer and deepens each step shown above.

OVERVIEW TO OMNIPAPER ARQUITECTURE

The Omnipaper's project begun in January 2002 and is structured into seven workpackages, where each one handle a well-defined part of the work, at the same time being highly linked to each other.

As depicted in Figure 1, the architecture used in the project makes a distinction between the Local Knowledge and Overall Knowledge Layer.

The three workpackages included in the Figure 1 (WP2, WP3 e WP5) will each pursue one of the three technological objectives.

On the level of the local layer, ways for retrieving information from distributed sources were analyzed. When combining different archives into one large information pool, access to them must be possible in a uniform way (WP 3). The Overall Knowledge layer will bring to the local layers together in a well-structured manner. It will make cross-archive navigation and linking in a multilingual environment. On top of the overall layer, the user interface will provide a user-friendly and interactive presentation of the knowledge layer (WP 5).

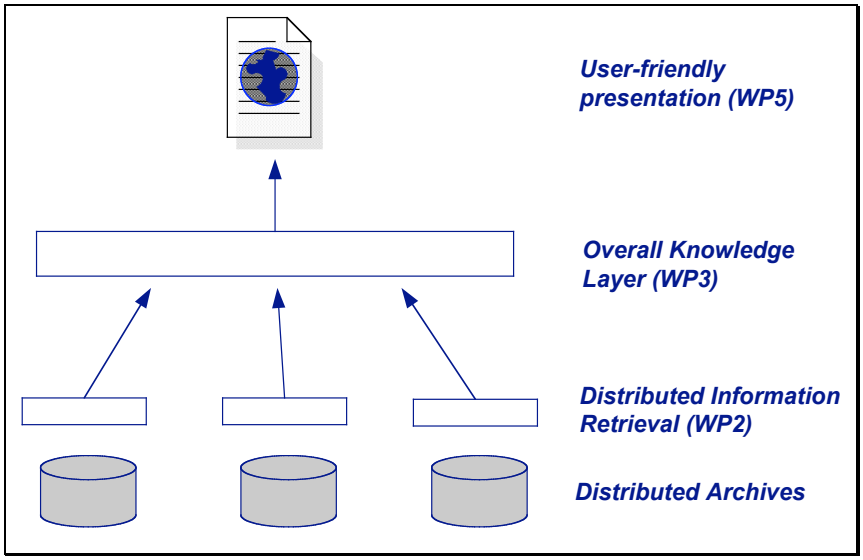

FIGURE 1: SYSTEM ARQUITECTURE

LOCAL KNOWLEGE LAYER

The implementation of the Local Knowledge Layer is developed in the WP2 and results will constitute the input for research on the Overall Knowledge Layer (WP3). On both levels, the RDF and Topic Maps approach will be analysed and compared.

The Local Knowledge Layer is added to the local archives where the main purpose is to provide a standard semantic description of all the existent articles in order to enable a structured and uniform access to the available distributed archives.

In the Local Knowledge Layer we are developing simultaneously two different metadata approaches to both layers: the Resource Description Framework (RDF) [3] approach, and the Topic Maps (TM) [4] approach. These prototypes will be cross-tested on a technical level and will allow conclusions on both levels [5].

OVERALL KNOWLEGE LAYER

The overall knowledge layer combines the features of integrating distributed information with the capability of creating semantic coupling of the corresponding content.

The results from the local layer queries will constitute the input for research on the overall layer. On both levels, new techniques are analyzed and compared.

Meta data techniques are evaluated to act as a knowledge management platform in order to improve content research and multilingual information retrieval. The multilingual aspects are supported by extracting existing keywords and metadata from the heterogeneous archive information and associate it with existing domain specific thesauri for the relevant language.

The overall knowledge layer contains a network of thesauri, thus coupling corresponding standardized terms and enabling the intelligent news archive to find corresponding articles in news archives over different countries

and languages. This allows journalists and researchers to investigate material on specific topics in a multilingual environment, relying on high result quality and content relevance [5].

RDF (RESOURCE DESCRIPTION FRAMEWORK

The Resource Description Framework (RDF) includes a model to express semantics [3].
The RDF model is simply a model of triples, what makes it very powerful, but difficult to implement. By definition, the RDF descriptions using the triples, the graph or the RDF/XML syntax are equivalent. The RDF/XML parsers read, and verify the RDF/XML syntax, and transform the code written in the RDF/XML syntax in a group of triples and, eventually, in a RDF graph.

RDF is divided in two parts, including two different specifications:

1. The RDF Model and Syntax Specification (RDFMSS) [, 1999 #26]is a recommendation of the W3C that contains a model to represent RDF metadata, as well as a syntax for expressing and transporting this metadata in a way that maximizes the interoperability of independently developed web servers and clients.
2. (2) The RDF Schema Specification [6] makes it possible to drawn and to implement in a consistent way, specific metadata vocabularies. These can still be further developed by other projects generating a network of metadata schemas. For example, certain terms of a vocabulary that need to be defined can be specified as refinements of DC [7] elements or of any other previously defined vocabulary. It was already said above that in our metadata description, we used some normalized vocabularies, like DCMES [8] and DC and also some (News) Text Format Standards like NITF [9] and NewsML [10]. Dublin Core is currently being used in many places, and is one of the foundations that RDF is building on.

The RDF/XML code is not simple. The RDF model is itself of an extraordinary simplicity and the descriptions are of a logical mathematical rigour. However, some parts of the model and of RDF/XML syntax are being currently subject to change. In fact, the W3C “RDF CORE” working group [11], is working in a new specification for RDF with two main goals: (1) to make the specification easier to read and to understand and (2) to remove some assumed mistakes from RDF/XML syntax.

On the other hand, the expression of metadata vocabularies in RDF has also not been a pacific issue for the last years. In 1999 DCMI released a working draft for establishing some rules on expressing Dublin Core in RDF. Since then a great discussion has been made around this subject. The “Expressing Qualified Dublin Core in RDF/XML” [12]working draft released in 2001 covers the most important issues regarding this subject. Although this document is not yet a recommendation from DCMI [13], it already has become a candidate recommendation, which gives us some confidence for using it as a reference guide for our RDF/XML descriptions, in Omnipaper’s project context.

OMNIPAPER APPLICATION PROFILE

The concept of application profiles has emerged in discussions on metadata schemas, in relation to work that is being done on metadata registries, specifically in the Dublin Core Metadata Initiative. The application profiles grew out of UKOLN's [14]work on the DESIRE [15] project.

Heery and Patel define application profiles as schemas which consist of data elements drawn from one or more namespace schemas, combined together by implementers and optimised for a particular local application [16]. They state that the principal characteristics of an application profile are that: it may draw on one or more existing namespaces; does not introduce new data elements; it can specify permitted schemes and values; and it can refine standard elements.

In the context of the Omnipaper’s project, the definition of the application profile is intended to describe not only the elements from different vocabularies but also elements from our particular concept of application, used in the description of our articles. The definition of all the elements used in the implementation of the RDF/XML metadata structure is fully described, by Yaginuma et al. [17].

The definition of all the elements used in the description of our articles were based on an elaborated analysis of several XML (eXtensible Markup Language) applications like XML NITF (News Industry Text Format) [9], NewsML (News Agency Implementation Guidelines) [10] developed by IPTC (International Press Telecommunications Council) [18], by metadata vocabularies like DCMES (Dublin Core Metadata Element Set) [8], and DCQ (Dublin Core Qualifiers) [7] and by the vocabulary SMES [19]; a set of metadata elements that best define the features of the newspaper articles and other elements were selected. Theses elements make part of the Omnipaper application profile that is illustrated in the picture below and the RDF/XML implementation is shown in [20]. It includes the following six vocabularies:

- Dublin Core Metadata Element Set (DCMES) – http://purl.org/dc/elements/1.1/;
- Dublin Core Qualifiers (DCQ) - http://purl.org/dc/elements/1.1/;
- News Industry Text Format (NITF) - urn:nitf:iptc.org:20010419:NITF;
- News Markup Language (NewsML) - urn:newsml:iptc.org:20010421:NEWSML;
- Omnipaper RDF Schema (omni) - http://www.dsi.uminho.pt/omn/schemas/omn-schema#; and
- vCard - http://www.w3.org/2001/vcard-rdf/3.0#.

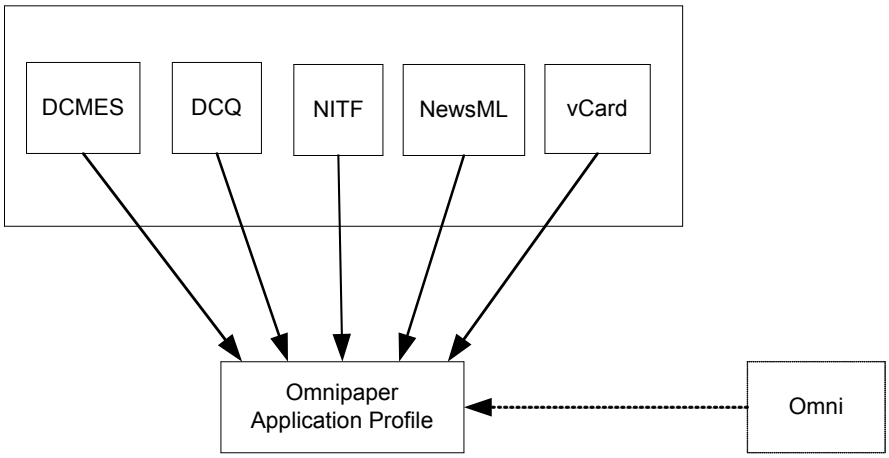

FIGURE 2 - METADATA VOCABULARIES FOR OMNIPAPER

OMNIPAPER SCHEMA

The Resource Description Framework Schema Specification (RDFSS) [3] is a proposed recommendation of the World Wide Web Consortium (W3C) [21]. The schema specification language is a declarative representation language influenced by ideas from knowledge representation (e.g. semantic nets, frames, predicate logic) as well as database schema specification languages and graph data models [3].

The main goal of the RDFSS consists of the definition of a schema specification language that allows the definition of mechanisms needed to define elements from specific vocabularies, to define classes of resources which those elements may be used with, according to restriction mechanisms, based on the constraint rdfs:domain, and in the definition of values for the properties through the rdfs:range.

The RDF Schema specification provides mechanisms of extensibility of RDF vocabularies, through the use of classes and specific properties of the RDF Schema. We can define our own vocabularies that are not more than specializations of vocabularies normalized as it is the case of DCMES (Dublin Core Metadata Element Set).

The concept of RDF schema is directly related with the concept of application profile. While in RDF Schema, is defined a vocabulary that it can be used in the context of one or more RDF applications. In each application profile, it is possible to identify the RDF schema, the elements of each vocabulary and its context in certain RDF application.

Heery and Patel [16] distinguish namespace schema from application profile schema.

Namespace is defined within the W3C XML schema activity [22] and allows for unique identification of elements, which may be defined through an RDF SCHEMA. Within the W3C XML and RDF schema specifications, namespaces are the domain names associated with elements which, along with the individual element name, produce a URL that uniquely identifies the element [16].

According to Heery and Patel [16] with ‘namespace’ we can

- Identify the management authority for an element set;
- Support definition of unique identifiers for elements;
- Uniquely define particular data element sets or vocabularies.

While in application profiles we can [16]:

- Use elements of one or more namespaces;
- Not introduce new elements;
- Specify schemas and allowed values;
- Refine normalized definitions.

In the Omnipaper project we use some metadata elements that were not previously defined neither in standard vocabularies nor in widely-used vocabularies. For this purpose a specific Omnipaper vocabulary containing these metadata elements was created using RDF schema.
In Figure 2, illustrated below, we show the properties defined in the Omnipaper RDF Schema [23].

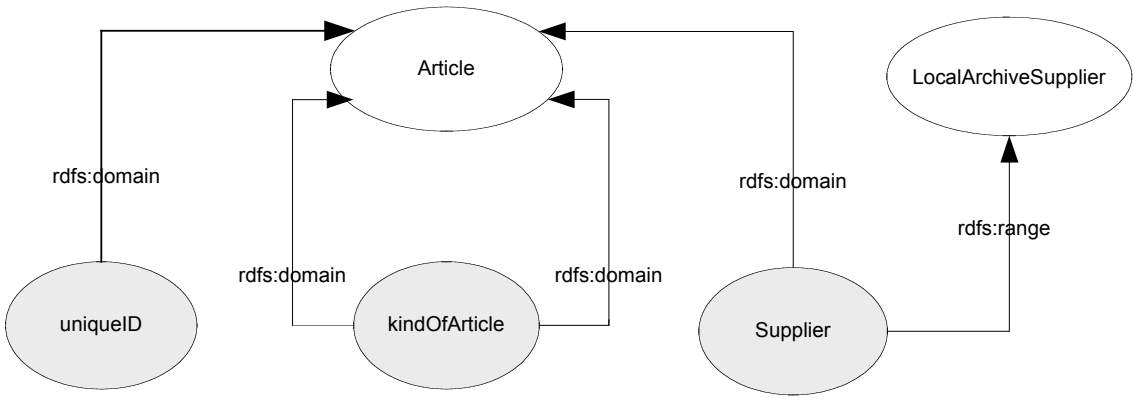

FIGURE 3 - SCHEMA PROPERTIES

uniqueID: Applies to all kinds of articles
kindOfArticle: Applies to all kinds of articles, many has a value any kind of article (any or its subclasses).
supplier: Applies to all kind of local archive owners. May have a value any kind of Local Archive Supplier (any of its subclasses).

Each property has a specific meaning, defines its permitted values, the types of resources it can describe, and its relationships with other properties [, 1999 #26].

In order to support this properties' definition, we defined in the RDF schema a set of classes as explained below.

The constraint rdfs:range indicates the classes that the values of a property must be members of. Each property can only have at maximum one rdfs:range constraint. "Although it is possible to express two or more range constraints on a property, a similar outcome can achieved by defining a common super class for any classes that represent appropriate values for some property" [6]. For this purpose were defined the classes Article and LocalArchivesSupplier where the classes Interview, Review, OpinionLetter and News were defined as subclasses of the class Article, and the classes PTE, MyNews and Mediargus were defined as subclasses of the class LocalArchivesSupplier, as illustrated in the Figure 3 and 4.

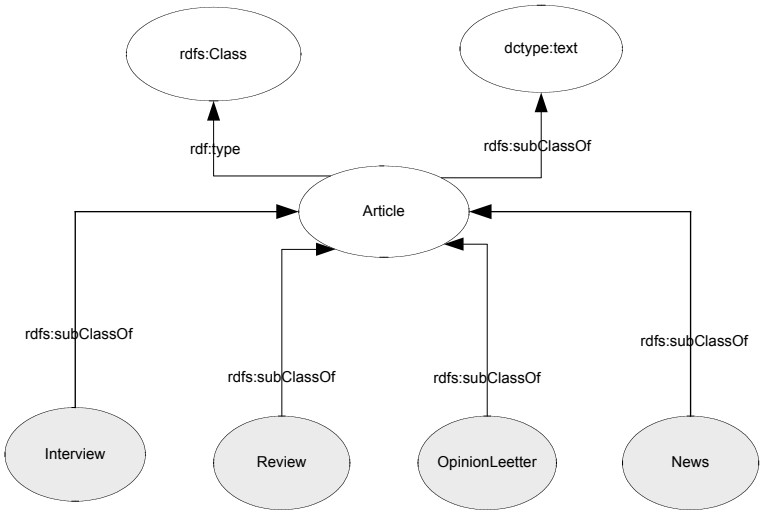

FIGURE 4 - OMNIPAPER SCHEMA: RELATIONSHIP BETWEEN ARTICLE CLASSES

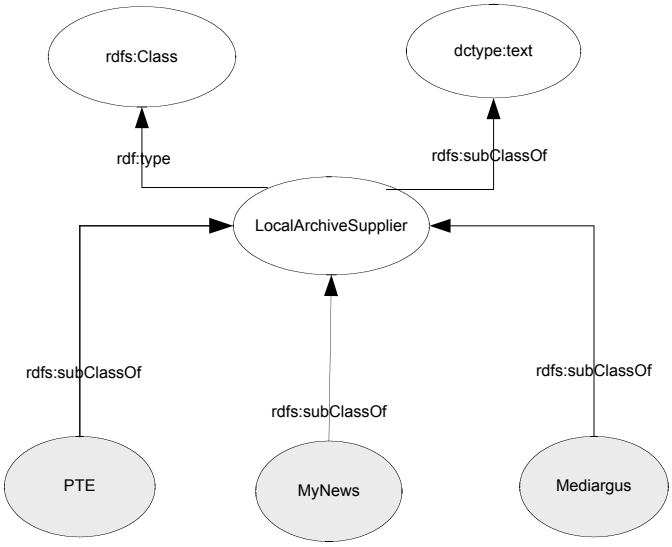

FIGURE 5 - OMNIPAPER SCHEMA – RELATIONSHIP BETWEEN LOCALARCHIEVESUPPLIER CLASSES

METADATABASE IMPLEMENTATION

The prototype for handling RDF layer in the Local Knowledge Layer was accomplished using RDF Gateway [21]. The RDF prototype was developed over RDF Gateway, since RDF Gateway can generate HTML pages to users, and can also create a Native Database and a Package that are by themselves the applications.

The application was developed in a scripting language called RDFQL. RDFQL is based on ECMA Script (Java Script) with query extensions to perform federated searches across multiple data sources. In the RDFQL server pages (RSP) we wrote and deployed on the RDF Gateway the complete definition of our application. The code inside RSP files (RDF Server Pages) interacts with the users and database engine, executing queries within the metadata database and sending answers to the users. Articles that match the query will be displayed by showing a selection of their metadata (title, date and author). By clicking the article, a SOAP request should be sent to retrieve the full text of the article from the news archive server.

The final result is an application that can be accessed using a Web browser by specifying an URL In a first phase of the prototype, we developed the description of the digital articles followed by the implementation

of the manipulation process in RDF Gateway, and we also developed a program to transform and upload information to the metadata base. The program transforms article and keyword files into RDF Files and uploads it. All this process off the system architecture is illustrated in the Figure 6.

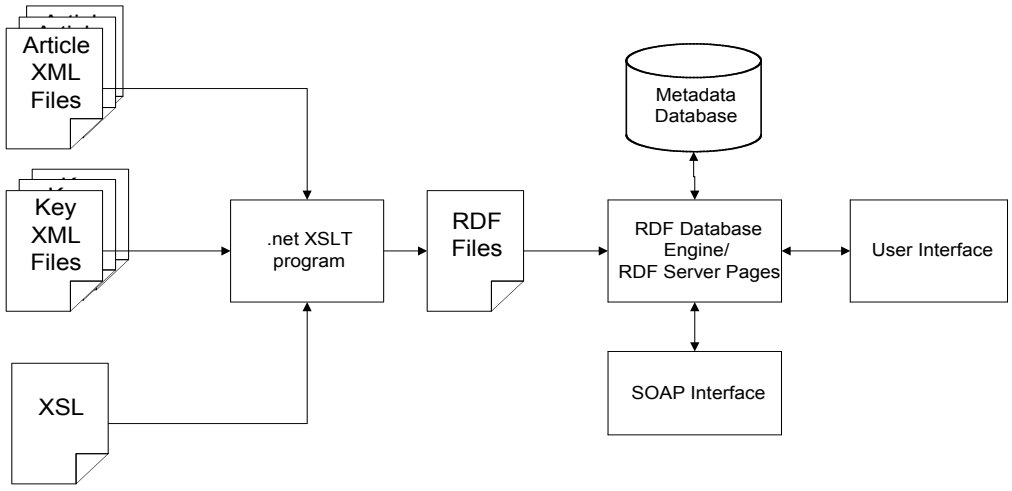

FIGURA 6: SYSTEM DESIGN

CONCLUSIONS AND FUTURE WORK

In the Omnipaper prototype, the main purpose of the Local Knowledge Layer is to provide a standard semantic description of all the existent articles in order to enable a structured and uniform access to the available distributed archives.

In this paper we discussed the development of RDF prototype, in the context of the European digital newspaper articles in the scope of Omnipaper's project.

The RDF prototype tries to investigate efficient ways to describe and store metadata used in the description of the digital newspaper articles and at the same time analyse de efficiency of the RDF technology in the information retrieval provided from different types of distributed information resources.

The metadata approach developed using Resource Description Framework (RDF) and Topic Maps (TM) technologies are being developed in simultaneously and will be cross-tested on a technical level allowing conclusions on both levels.

The cross-testing of the prototypes and comparisons to both metadata structures approach (implemented in Topic Maps and RDF), will allow the project partners to make well considered decisions on the development of the final prototype.

In the next phase we will implement the whole RDF and Topic Maps's prototypes, and then we will accomplish ranking tests, comparing both approaches, according to the comparison criteria previously defined.

NOTES AND REFERENCES

1. *Resource Description Framework (RDF) Model and Syntax Specification*. Feb. 22, 1999. W3C. Available from: <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>
2. *Omnipaper - Smart Access to European Newspapers, EU project IST 2001-32174*. 2002. Available from: <http://www.omnipaper.org>
3. *Resource Description Framework (RDF)*. 2003. W3C. Available from:<http://www.w3.org/RDF/>
4. Group, M.o.t.T.O.A.. *XML Topic Maps (XTM) 1.0*. 2001. XTM TopicMaps.org. Available from:<http://www.topicmaps.org/xtm/1.0/>
5. Paepen, B.E., Jan; Schranz, Markus; Tscheligi, Manfred. *Omnipaper: Bringing Electronic News Publishing to a next Level Using XML and Artificial Inteligence*. Proceedings of the 6th International ICCC/IFIP Conference on Electronic Publishing held in Karlovy Vary, Czeck Republic, November 2002.
6. *RDF Vocabulary Description Language 1.0: RDF Schema*. Jan. 23, 2003. W3C. Available from: <http://www.w3.org/TR/rdf-schema/>
7. *Dublin Core Qualifiers*. 2002. Available from: <http://dublincore.org/documents/dcmes-qualifiers/>
8. *Dublin Core Metadata Element Set, Version 1.1: Reference Description*. Feb 4, 2003. Available from: <http://www.dublincore.org/documents/dces/>
9. *NITF - News Industry Text Format*. IPTC. 2002. Available from: <http://www.nitf.org/>
10. Welcome to the NewsML™ Website. IPTC. 2002. Available from: <http://www.newsml.org/>
11. *RDFCore Working Group*. W3C. 2003. Available from: <http://www.w3.org/2001/sw/RDFCore/>
12. Schwänzl, R.; K., Stefan, *Expressing Qualified Dublin Core in RDF/XML*. Apr. 14, 2002. Available from: <http://www.dublincore.org/documents/2002/04/14/dcq-rdf-xml/>
13. *Dublin Core Metadata Initiative*. 2003. Available from: <http://www.dublincore.org
14. *UKOLN* . 2002. Available from: <http://www.ukoln.ac.uk>
15. *DESIRE: Development of a European Service for Information on Research and Education. EU project.*2003. Available from: <http://www.lub.lu.se/desire/>
16. Heery, R. ; P., Manjula . *Application Profiles: Mixing and Matching Metadata Schemas*. Ariadne. Sep. 25, 2002. Available from: <http://www.ariadne.ac.uk/issued25/app-profiles/>
17. Yaginuma, T.P., Teresa; Baptista, Ana Alice, *Metadata elements for digital news resource description*. 2002.
18. *Information Technology for News*. 2000, 2001. Available from: <http://www.iptc.org>
19. SCHEMA, U., *Forum for Metadata Schema Implementors*. 2002. Available from: <http://www.schemas-forum.org>
20. Pereira, T., B., Ana Alice, *Omnipaper Apllication Profile*. 2002. Available from: <http://www2.dsi.uminho.pt/tpereira/OmnipaperApplicationProfile1.rdf>
21. Chappell, G.R., Derrish; Michal , Aaron, *Intellidimension Gateway*. 2002. Available from: <http://www.intellidimension.com/>
22. *Extensible Markup Language (XML) Activity Statement*. W3C. 2002. Available from: <http://www.w3.org/XML/Activity>
23. Pereira, T.B., Ana Alice, *Omniapaper Schema Namespace*. 2002. Available from: <http://www2.dsi.uminho.pt/tpereira/Omni_Schema.rdf>