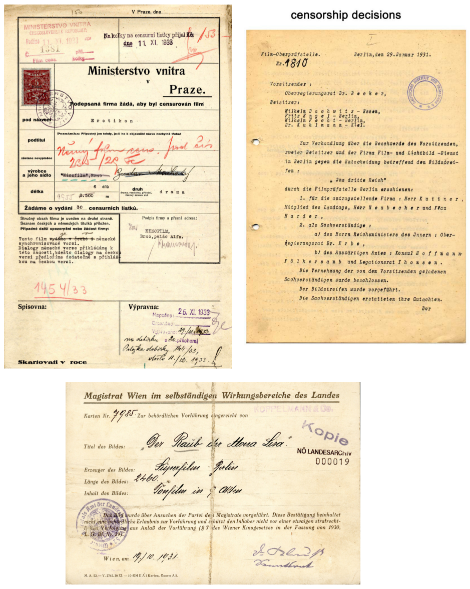
Automatic Annotation of Historical Paper Documents

S. Ferilli, L. Iannone, I. Palmisano, G. Semeraro, T.M.A. Basile, N. Di Mauro
Dipartimento di Informatica – Università di Bari, Via E. Orabona, 4 - 70125 Bari
{ferilli, iannone, semeraro, basile, nicodimauro}@di.uniba.it
ignazio_io@yahoo.it

Abstract
The European Community project COLLATE (Collaboratory for Annotation, Indexing and Retrieval of Digitized Historical Archive Material) is concerned with digitised historical/cultural material. One of the main features of COLLATE system architecture is the integration of software components that exploit state-of-the-art techniques coming from the area of Artificial Intelligence and Knowledge Representation. This work describes the results achieved by applying Machine Learning methods for automatic classification and labelling of documents. Furthermore, we also discuss the advantages obtained by exploiting brand new research achievements in KR for the design of COLLATE data model.

Introduction

The IST-1999-20882 project COLLATE (Collaboratory for Annotation, Indexing and Retrieval of Digitized Historical Archive Material) aims at developing a WWW-based *collaboratory* [4] for archives, researchers and end-users working with digitised historical material (`http://www.collate.de`). The chosen sample domain is a large corpus of multi-format documents concerning rare historic films from the 20's and 30's, provided by 3 major European film archives.
One of the main features of COLLATE system architecture is the integration of software components that exploit state-of-the-art techniques coming from the area of Artificial Intelligence and closely related research, such as Knowledge Representation (KR). This work describes the results achieved by applying Machine Learning methods for automatic classification and labelling of documents. We also discuss the advantages obtained by exploiting brand new research achievements in KR for the design of COLLATE data model.
The need of automatically labelling such a huge amount of documents, along with their significant components, suggested the use of machine learning techniques to learn rules for such tasks from a small number of selected and annotated sample documents, as well as the development of a software component devoted to the management of such annotated documents. The challenge comes from the low layout quality (manual annotations, stamps that overlap to sensible components, ink specks, etc.) and standard (many documents are typewritten sheets, that consist of all equally spaced lines in Gothic type) of such a material, which introduces a considerable amount of noise in its description. In particular, the complexity of the domain and the need for the rules to be understandable by film experts, led to the choice of symbolic first-order logic learning.
The COLLATE repository, set up by the film archives DIF (Deutsches Filminstitut, Frankfurt am Main), FAA (Film Archive Austria, Vienna) and NFA (Národni Filmový Archiv, Prague), includes a large collection of several thousands comprehensive documents concerning film culture, and focuses on documents related to censorship processes. Each archive is characterized by a predominant kind of documents, as reported in Figure 1. The *Application Forms* (upper-left of the Figure), required for applying to get the permission to show a film from a production or distribution company, characterize NFA. The *Censorship Decisions* (upper-right of the figure), very frequent at DIF, concern the decision whether a film could or could not – and in which version – be distributed and shown throughout a Country. The greatest portion of documents belonging to the FAA repository are *Registration Cards* (lower part of the figure), that represent a certification that the film had been approved for exhibition in the present version by the censoring authority.


censorship decisions

V Praze, dne
MINISTERSTVO VNITRA ČESKOSLOVENSKÉ REPUBLIKY:
Podáno 11. XI. 1933
Č. 1381 příl. kolky
Film cens.
Na kolky na censurní lístky přijal Kč dne 11. XI. 1933
Ministerstvo vnitra v Praze.
Podepsaná firma žádá, aby byl censurován film
pod názvem: Erotikon
Poznámka: Připustný jen tehdy, je-li ho k objasnění názvu nezbytně třeba!
podtitul
zůstane nevyplněno
výrobce a jeho sídlo: "Kinofilm", Brno
délka: 6 dílů 2.500 m
druh (drama, veselohra, přírodní, časový snímek atd.): drama
Žádáme o vydání 30 censurních lístků.
Stručný obsah filmu je uveden na druhé straně.
Seznam českých a německých titulků přiložen.
Případné další upozornění nebo žádost firmy:
Tento film ~~vydáme v české i~~ německé synchronisované versi.
Dialogy německé verse přikládáme k této žádosti, kdežto dialogy na českou versi předložíme dodatečně s přihláškou na českou versi.
Podpis firmy a přesná adresa:
KINOFILM,
Brno, palác Alfa.
1454/33
Spisovna:
Výpravna:
Napsáno: 25. XI. 1933
Srovnáno:
Vypraveno: 29./11. 1933
s 20 přílohami
Skartovali v roce

Film-Oberprüfstelle. Berlin, den 29. Januar 1931.
Nr. 1810
Vorsitzender:
Oberregierungsrat Dr. B e c k e r,
Beisitzer:
Wilhelm D a c h w i t z - Essen,
Fritz E n g e l - Berlin,
Wilhelm F e c h t - Berlin,
Dr. K u h l m a n n - Kiel.
Zur Verhandlung über die Beschwerde des Vorsitzenden, zweier Beisitzer und der Firma Film- und Lichtbild -Dienst in Berlin gegen die Entscheidung betreffend den Bildstreifen:
„Ins dritte Reich"
durch die Filmprüfstelle Berlin erschienen:
1. für die antragstellende Firma: Herr K u t t n e r, Mitglied des Landtags, Herr N e u b e c k e r und Frau H a r d e r,
2. als Sachverständige:
a) des Herrn Reichsministers des Innern: Oberregierungsrat Dr. E r b e,
b) des Auswärtigen Amtes: Konsul H o f f m a n n-F ö l k e r s a m b und Legationsrat T h o m s e n.
Die Vernehmung der von dem Vorsitzenden geladenen Sachverständigen wurde beschlossen.
Der Bildstreifen wurde vorgeführt.
Die Sachverständigen erstatteten ihre Gutachten.
Der
DEUTSCHES INSTITUT FÜR FILMKUNDE E.V.

Magistrat Wien im selbständigen Wirkungsbereiche des Landes
Karten Nr. 7985 Zur behördlichen Vorführung eingereicht von KOPPELMANN & Co.
Kopie
NÖ LANDESARCHIV 000019
Titel des Bildes: „Der Raub der Mona Lisa"
Erzeuger des Bildes: Superfilm Berlin
Länge des Bildes: 2460 m
Inhalt des Bildes: Tonfilm in 9 Akten
Das Bild wurde über Ansuchen der Partei dem Magistrate vorgeführt. Diese Bestätigung beinhaltet nicht eine behördliche Erlaubnis zur Vorführung und schützt den Inhaber nicht vor einer etwaigen strafrechtlichen Verfolgung aus Anlaß der Vorführung (§ 7 des Wiener Kinogesetzes in der Fassung von 1930, L. G. Bl. Nr. 29).
Wien, am 19/10. 1931.
M. A. 52. — V. 2745. 30 XI. — 10-RM II A 1 Karton. Önorm A 5.

Figure 1: Sample COLLATE documents

The Learning System

INTHELEX (INcremental THEory Learner from EXamples) [5] is the learning system embedded in the COLLATE architecture as a learning component. It carries out the induction of *hierarchical* first-order logic theories from positive and negative examples: it learns simultaneously *multiple concepts*, possibly related to each other; it guarantees validity of the theories on all the processed examples; it uses feedback on performance to

activate the theory revision phase; in addition to the possibility of refining a previously generated version of the theory, learning can also start from an empty theory; it is based on the *Object Identity assumption* (terms denoted by different names within a formula must refer to different objects). It exploits a previous version of the theory (if any), a graph describing the dependence relationships among concepts, and an historical memory of all the past examples that led to the current theory.
Incremental learning is necessary when either incomplete information is available at the time of initial theory generation, or the nature of the concepts evolves dynamically. Both cases are very frequent in real-world situations, hence the need for incremental models to complete and support the classical batch ones, that perform learning in one step and hence require the whole set of observations to be available from the beginning.
The learning cycle performed by INTHELEX can be described as follows. A set of examples of the concepts to be learned, possibly selected by an expert, is provided by the environment. This set can be subdivided into three subsets, namely training, tuning, and test examples, according to the way in which examples are exploited during the learning process. Specifically, training examples, previously classified by the expert, are abstracted and stored in the base of processed examples, then exploited to obtain a theory that is able to explain them. Such an initial theory can also be provided by the expert, or even be empty. Subsequently, the validity of the theory against new available examples, also abstracted and stored in the example base, is checked by taking the set of inductive hypotheses and a tuning/test example as input and producing a decision that is compared to the correct one. In the case of incorrectness on a tuning example, the cause of the wrong decision can be located and the proper kind of correction chosen, firing the theory revision process. In this way, tuning examples are exploited incrementally to modify incorrect (too weak or too strong) hypotheses according to a data-driven strategy. Test examples are exploited just to check the predictive capabilities of the theory, intended as the behavior of the theory on new observations, without causing a refinement of the theory in the case of incorrectness on them.
Another peculiarity of INTHELEX is the integration of multistrategy operators that may help solve the theory revision problem by pre-processing the incoming information [1]. The purpose of induction is to infer regularities and laws that may be valid for the whole population. INTHELEX incorporates two inductive refinement operators, one for generalizing hypotheses that reject positive examples, and the other for specializing hypotheses that explain negative examples. Deduction is exploited to fill observations with information that is not explicitly stated, but is implicit in their description, and hence refers to the possibility of better representing the examples and, consequently, the inferred theories. Indeed, since the system is able to handle a hierarchy of concepts, some combinations of predicates might identify higher level concepts that are worth adding to the descriptions in order to raise their semantic level. For this reason, INTHELEX exploits deduction to recognize such concepts and explicitly add them to the example description. Abduction aims at completing possibly partial information in the examples, adding more details. Its role in INTHELEX is helping to manage situations where not only the set of all observations is partially known, but each observation could also be incomplete. Abducibles are the predicates on which assumptions (abductions) can be made; integrity constraints provide indirect information on them and, since several explanations may hold for this problem setting, are also exploited to encode preference criteria for selecting the best ones. The proof procedure implemented in INTHELEX corresponds, intuitively, to the standard Logic Programming derivation suitably extended in order to consider abducibles and integrity constraints. Lastly, abstraction removes superfluous details from the description of both the examples and the theory. The exploitation of abstraction in INTHELEX concerns the shift from the language in which the theory is described to a higher level one. An abstraction theory contains information on the operators according to which the shift is to be performed. INTHELEX, automatically applies it to the learning problem at hand before processing the examples. The implemented abstraction operators allow the system to replace a number of components with a compound object, to decrease the granularity of a set of values, to ignore whole objects or just part of their features, and to neglect the number of occurrences of a certain kind of object.
INTHELEX is currently available in binary format for i586 DOS-based platforms `http://lacam.di.uniba.it:8000/systems/inthelex/`).

Experimental Results

The dataset considered for the experimental session consisted of 102 documents (each described by 223 literals on average) from the 3 classes of interest (37 Application Form from NFA, 36 Censorship Decision from DIF and 29 Registration Card from FAA), plus 17 reject documents corresponding to articles from the contemporary film press, newspapers or magazines. INTHELEX was considered a suitable learning component because many of its features met the requirements imposed by the complexity of the documents to be handled, as later confirmed by experimental results reported in the following. The first-order logic descriptions of documents, needed to run INTHELEX, were automatically generated by the system WISDOM++ [3]. Each document was considered as a positive example for the class it belongs to, and as a negative example for the other classes; reject documents were considered as negative examples for all classes. Since each kind of document is composed by different blocks whose labels regard the role they play in it, it is necessary to learn, firstly, the class the document belongs to, before starting a process aimed at learning definitions for the semantic labels in it.

Hence, a first experiment, aimed at learning definitions for each class, starting from the empty theory, was carried out and the predictive accuracy of the resulting theories was tested according to a 10-fold cross validation methodology, ensuring that each fold contained the same proportion of positive and negative examples. Table 1 reports the experimental results, averaged on the 10 folds, of the classification process in this environment as regards number of clauses that define the concept (CL), number of performed revisions (REV), Accuracy on the test set (expressed in percentage) and Runtime (in sec.). After this preliminary phase of classification, a further experiment aimed at learning rules to identify the blocks of which the documents are made up was carried out. As regard the class of documents Registration Card (FAA), the domain experts provided the following labels characterizing the objects belonging to it (in brackets the number of instances of the items in the document dataset): registration_au [28], date_place [26], department [17], applicant [11], reg_number [28], film_genre [20], film_length [19], film_producer [18], film_title [20]. Like in the classification step, each example is positive for the label(s) it belongs to, and negative for all the others. The average results of the 10-fold cross-validation are reported in Table 2.

	CL	REV	Accuracy	Time
DIF	1.00	7.50	99.17	17.13
FAA	3.50	9.70	94.17	334.05
NFA	2.25	6.35	95.74	89.88

Table 1: Statistics for Classification

FAA	CL	REV	Accuracy	Runtime
registration_au	5.6	12.5	91.43	3739.36
date_place	6.9	13.5	86.69	7239.62
department	1.9	6.6	98.95	118.62
applicant	2	4.5	97.89	93.99
reg_number	5.1	14.4	91.95	4578.20
film_genre	4	8.4	93.02	2344.89
film_length	5.5	9.9	90.87	3855.39
film_producer	4.9	10.4	94.05	4717.17
film_title	5.4	11.1	89.85	4863.08

Table 2: Statistics for Understanding of FAA

DIF	Clau	Revisions	Accuracy	Runtime
cens_signature	2.2	11.6	98.32	1459.88
cert_signature	2.2	7.6	98.31	176.59
object_title	5	15.2	94.66	3960.82
cens_authority	2.9	12.1	97.64	2519.45
chairman	4.6	13.8	93.10	9332.84
assessors	4.6	15	94.48	12170.93
session_data	2.5	8.6	97.68	1037.96
representative	5.6	20.7	92.98	13761.95

Table 3: Statistics for Understanding of DIF

The labels specified for class Censorship Decision of DIF were: cens_signature [35], cert_signature [35], object_title [36], cens_authority [36], chairman [36], assessors [36], session_data [36], representative [49]. Table 3 shows the results of a 10-fold cross-validation run on this dataset. Finally, the documents Application Form of the class NFA were characterized by these labels, almost all different from the others: dispatch_office [33], applic_notes [18], no_censor_card [21], film_producer [20], no_prec_doc [20], applicant [22], film_genre [17], registration_au [25], cens_process [30], cens_card [26], delivery_date [16]. Again, a 10-fold cross-validation was applied, whose averaged results are reported in Table 4.

NFA	Clauses	Revisions	Accuracy	Runtime	NFA	Clauses	Revisions	Accuracy	Runtime
dispatch_office	6.8	13.9	94.28	13149.31	applicant	6.7	11.5	93.66	3739.36
applic_notes	2.5	5.7	98.81	231.05	film_genre	2.8	6.9	98.53	684.35
no_censor_card	5.3	11.2	95.47	8136.79	registration_au	4.1	12.5	94.64	5159.74
film_producer	4.9	9.6	93.98	5303.78	cens_process	4.8	10.8	98.51	4027.90
no_prec_doc	4.6	11	93.97	5561.14	cens_card	5.6	11.8	94.62	3363.86
delivery_date	4	9.1	95.52	3827.34					

Table 4: Statistics for Understanding of NFA

As expected, the classification problem turned out to be easier than the interpretation one (that is concerned with the semantics of the layout blocks inside documents). This is suggested by the tendential increase in number of clauses, performed revisions and runtime from Table 1 to Tables 2, 3 and 4. Such an increase is particularly evident for the runtime, even if it should be considered that the high predictive accuracy should ensure that very few documents will cause theory revision.

The XML Engine

XML Content Manager (XMLCM) is the software component devoted to the management of information flow within COLLATE. XMLCM can be described as a set of software entities that manipulate information at different levels of abstraction. As its name suggests, it is strongly based on the eXtensible Mark-up Language (XML) technology by W3 Consortium[1]. Though powerful and orientated to interoperability, XML alone cannot guarantee the whole support needed by COLLATE in order to properly represent and cope with its domain object model. In fact, the need of enriching documents belonging to the film heritage through the addition of annotations by film scientists and consequently the need of performing search on documents as well as on their annotations, require COLLATE system to be able to manage what has been called "scientific discourse" [2]. This brought us to adopt more powerful KR languages - Resource Description Framework (RDF)[2] and its evolution DAML+OIL [7] – and to develop a specific XMLCM architectural layer that can deal with RDF. Such a component offers all the typical operations on RDF basic objects, i.e. Descriptions, Models, Statements. In addition, it has some features that currently available RDF tools do not offer, such as the support to RDF resource sharing among multiple users (human beings or software agents) and the support to collaboration. Another feature of that layer is the compliance with DARPA Agent Manipulation Language (DAML) specification that, together with the Ontology Inference Layer (OIL), gives the possibility of accomplishing basic reasoning processes, such as classification of instances, with the power of results achieved in Description Logics (DL) research. In fact, DAML+OIL is a specific DL language, thus XMLCM can be easily integrated with state-of-the-art DL reasoning technologies, such as FaCT reasoner [6].

Architectural overview

Figure 2 provides an overall sketch of the whole XMLCM architecture. It can be seen as a set of layers relying on each other. Each layer has its own level of abstraction in the sense that it tackles a peculiar issue within the whole system.

Persistence Layer is devoted to guarantee an effective physical storage of XML resources. The design of this layer is based on the well-known *Strategy* pattern [8], allowing for the implementation of different physical persistences, that can be easily switched through, in order to exploit the better solution to the specific problem of an application, thus enabling the component to be reused in a wide variety of systems. At this time, three implementations have been developed:

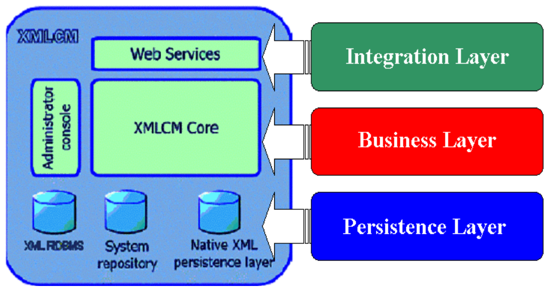


Figure 2. XMLCM Overall architecture.

- Binary compression on file system. This process relies on a commercially available API called PDOM[3].
- RDBMS storage of XML. This solution is based on Oracle 9i[4] RDMBS and exploits its XML capabilities.
- Native XML Databases. This solution takes advantage of an XML dedicated Database server such as Software AG's Tamino[5]. Tamino based persistence has been developed within the project COVAX [9] and is the most performing layer of persistence in terms of effective storage of XML documents.

The Business Layer grants all the typical operations on an XML resource. They range from the simple creation, update, deletion and retrieval of a document (or a bunch of them), to the manipulation of basic XML documents subunits called elements. Again for each element (or bunch of elements) creation, updating, deletion and retrieval are possible. This layer embeds a language for querying single o multiple documents fully compliant with XQL specification [10]. Besides this there is a strong support provided for document versioning. In fact within XMLCM more than a version can be stored for each document. The system tracks version histories providing support for comparisons and merger of two or more different versions of the same XML resource. Also, in order to navigate through different version and carry out ordinary XQL-query among versioned documents, system can be instructed by means of a proprietary XML based language. This language forces the system to restrict the scope of ordinary XQL query just on some particular set of versions adjusting versions peculiar parameters (e.g.: authors, date, etc.).

The Integration Layer is responsible for integration of XMLCM with external systems. It has been developed following the state of the art in terms of interoperability. The paradigm we used is: Web Services. This allows every external application to exploit all XMLCM services using a very simple communication protocol such SOAP [11]. Web services seem to be the most promising response to distribution and decentralization needs of Internet based application. Developing such a layer we guarantee the possibility of *moving* single subsystems of a wider architecture (say COLLATE whole system) across the internet. Moreover it guarantees loosely coupling between system components, and this results in high maintainability of the whole architecture.

[1] http://www.w3.org/xml
[2] http://www.w3.org/RDF/
[3] http://www.infonyte.com/en/prod_pdom.html
[4] http://otn.oracle.com/products/oracle9i/content.html
[5] http://www.softwareag.com/tamino/

RDF Management

We previously pointed out that XML alone was not very effective within COLLATE. In fact we needed to represent the relationships among COLLATE domain objects with a level of abstraction that allowed for some basic reasoning on relationships themselves. For instance final users had to be able to navigate among heterogeneous resources following strongly typed links among them. The most common resources were the annotations that film scientist made upon documents. All those annotations were looked at as they constituted scientific discourses on a given resource in the archive. In such a view the need for a well established technology for addition of metadata to COLLATE document arouse. Resource Description Framework (RDF) proved to be the most suitable achievements in producing metadata in Web based systems. Furthermore its possible serialization in XML was one of the most interesting features since it could be easily manipulated by tools that XMLCM technology offered. That is why we developed a further component that could be plugged into XMLCM (but that could work as standalone as well). This component was designed since its origins to be compliant with RDF standards. It featured all the commonly used operations on RDF basic structures (i.e. RDF Models and Statements). Our main aim was to synthesize in one framework all the available features that we could find in the most widespread tools devoted to RDF. Furthermore XMLCM solution for RDF added to the most common APIs the support for multi-user environments and applications and multi query language capabilities. The need for having more than a query language for RDF (and its derivatives such as DAML+OIL) was the complete lack of standards in querying RDF resources. We currently allow for querying RDF resources using many query languages such as RDQL[6], RQL[13] and SquishQL[7]. Also, an early support for ontological services (such as validation of knowledge bases in DAML+OIL) has been provided. Currently, a tighter and more effective integration with cutting edge reasoning technologies (FaCT reasoner) is under development.

References

[1] S. Ferilli. A Framework for Incremental Synthesis of Logic Theories: An Application to Document Processing. *Ph.D. thesis*, Dipartimento di Informatica, Università di Bari, Bari, Italy, November 2000.

[2] I.Frommholz, H. Brocks, U. Thiel, E. Neuhold, L. Iannone, G. Semeraro, M. Berardi & M. Ceci. Document-centered Collaboration for Scholars in the Humanities - The COLLATE System. In T. Koch and I.T. Solvberg (Eds.), *Proceed. of ECDL 2003, 7th European Conference on Digital Libraries*, Lecture Notes in Computer Science, Springer:Berlin, 2003.

[3] F. Esposito, D. Malerba, & F.A. Lisi. Machine learning for intelligent processing of printed documents. *Journal of Intelligent Information Systems*, 14(2/3):175-198, 2000.

[4] R.T.Kouzes, J.D.Myers, & W.A.Wulf. Collaboratories: Doing science on the internet. *IEEE Computer*, 29(8):40-46, 1996.

[5] F. Esposito, G. Semeraro, N. Fanizzi & S. Ferilli. Multistrategy Theory Revision: Induction and Abduction in INTHELEX. *Machine Learning*, 38(1/2):133-156, Kluwer Academic Publ., Boston, January/February 2000.

[6] I.Horrocks. Using an expressive description logic: FaCT or fiction? In Cohn, A. G., Schubert, L. and Shapiro S. C. (Eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixth Int. Conference (KR'98)*, 636-647. Morgan Kaufmann Publishers, San Francisco, California, June 1998.

[7] I.Horrocks. DAML+OIL: a Reason-able Web Ontology Language. In Jensen, C. S., Jeffery, K. G., Pokorny, J., Saltenis, S., Bertino, E., Böhm, K., and Jarke, M. (Eds.), *Advances in Database Technology - EDBT 2002*, Lecture Notes in Computer Science 2287, 2-13, Springer:Berlin, 2002.

[8] E. Gamma, R. Helm, R. Johnson, J. Vlissides, *Design Patterns*. Addison-Wesley Pub Co; 1st edition, 1995.

[9] O. Licchelli, F. Esposito, G. Semeraro & L. Bordoni. Personalization to Improve Searching in a Digital Library. In P. Isaías, F. Sedes, J.C. Augusto and U. Ultes-Nitsche (Eds.), *New Technologies for Information Systems*, Proceed. of the 3rd Int. Workshop on New Developments in Digital Libraries, in conj. with 5th International Conference on Enterprise Information Systems ICEIS 2003, 46-55, Angers, France, April 22, 2003.

[10] E. Xavier. Using Extensible Query Language (XQL) for Database Applications. In Proceedings of Eighth Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS ' 01) April 17-20, 2001 Washington DC pp.2-9 http://www.computer.org/proceedings/ecbs/1086/1086toc.htm

[11] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, S. Weerawarana. Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI. IEEE Internet Computing March/April 2002 (Vol. 6, No. 2) pp. 86-93.

[12] RQL: A Declarative Query Language for RDF, G. Karvounarakis, S. Alexaki, V. Christophides, D. Plexousakis, Michel Scholl, The Eleventh International World Wide Web Conference (WWW' 02), Honolulu, Hawaii, USA, May 7-11, 2002.

[6] http://www.hpl.hp.com/semweb/rdql.htm#RDQL%20Grammar

[7] http://swordfish.rdfweb.org/rdfquery/squish-bnf.html