PERSONALIZED ACCESS TO MULTI-VERSION DOCUMENTS FOR E-GOVERNMENT APPLICATIONS

Fabio Grandi, Maria Rita Scalas
DEIS, Alma mater Studiorum – Università di Bologna
Viale Risorgimento 2, I-40136, Bologna, Italy
E-mail: fgrandi@deis.unibo.it, mrscalas@deis.unibo.it

Federica Mandreoli, Riccardo Martoglia
DII, Università di Modena e Reggio Emilia
Via Vignolese 905/b, I-4110, Modena, Italy
E-mail: mandreoli.federica@unimo.it, martoglia.riccardo@unimo.it

ABSTRACT

In this paper we describe the design and implementation of two prototype systems for the efficient management of multi-version XML documents in an e-Government scenario. The application aim is to enable citizens to access personalized versions of resources, like norm texts and information made available on the Web by public administrations. In the first system developed, four temporal dimensions (validity, efficacy, transaction and publication times) were used to represent the evolution of norms in time and their resulting versioning and a "stratum" approach was used for its implementation on top of an object-relational DBMS. Recently, the multi-version management system has migrated to a different architecture ("native" approach) based on a multi-version XML query processor developed on purpose. Moreover, a new semantic dimension has been added to the versioning mechanism, in order to represent applicability of norms to different classes of citizens according to their digital identity. Classification of citizens is based on the management of an ontology with the deployment of semantic Web techniques. Preliminary experiments showed an encouraging performance improvement with respect to the "stratum" approach and a good scalability behavior.

KEYWORDS

e-Government, XML, document retrieval, temporal database, semantic Web.

1. INTRODUCTION

In this paper we present our research activities concerning the implementation of Web information systems for e-Government applications (EC E-Gov, 2004; US E-Gov, 2004). More precisely, our work makes use of temporal database and semantic Web techniques to provide *personalized* access to multi-version resources and services provided by the Public Administration (PA). The offering of personalized versions is aimed at improving and optimizing the involvement of citizens in the e-Governance process. In particular, we consider the selective access to norm texts and documents made available on Web repositories in XML format (XML, 2004).

First of all, the fast dynamics involved in normative systems implies the coexistence of *multiple versions* of the norm texts stored in a repository, since laws are continually subject to amendments and modifications. In fact, it is crucial to reconstruct the *consolidated version* of a norm as produced by the application of all the modifications it underwent so far, that is the form in which it currently belongs to the regulations and must be enforced today. However, also past versions are still important, not only for historical reasons: for example, if a Court has to pass judgment today on some fact committed in the past, the version of norms which must be applied to the case is the one that was in force then. In other words, temporal concerns are widespread in the e-Government domain and a legal information system should be able to retrieve or reconstruct on demand any version of a given document to meet common application requirements. Hence, personalization in such a context is based on the user's temporal perspective.

Moreover, another kind of versioning plays an important role in an e-Government scenario, because some documents or some of their parts have or acquire a limited applicability. For example, a given norm (e.g. defining tax treatment) may contain some articles which are applicable to different classes of citizens: one article is applicable to unemployed persons, one article to self-employed persons, one article to public servants only and so on. Hence, a citizen accessing a retrieval service may be interested in finding a tailored version of the norm, that is a version only containing articles which are applicable to his/her personal case. Hence, personalization in such a context is based on limited applicability to the citizen's case and semantic versioning is required to the document repository. Finally, notice that temporal and limited applicability aspects though orthogonal may also interplay in the production and management of versions. For instance, a new norm might state a modification to a preexisting norm, where the modified norm becomes applicable to a limited category of citizens only (e.g. retired persons), whereas the rest of the citizens remain subject to the unmodified norm.

In this framework, we defined data models for multi-version XML documents and built prototype systems for their efficient management in a Web-based e-Government application scenario involving an on-line personalized access to norm repositories. In particular, in this work we will describe and compare two management systems, meeting different application requirements, that we recently developed using different architectures and implementation techniques. The first system is based on multi-dimensional temporal versioning, where temporal aspects are captured by adding timestamping attributes to the XML markup. The prototype was implemented using a "stratum" approach on top of a commercial DBMS and will be briefly described in Section 2 (a more detailed description and evaluation has also been published before as (Grandi et al, 2003a ; Grandi et al, 2003b; Grandi et al, 2005). The second system, which will be described in Section 3, is the current outcome of an ongoing research, which is introduced in (Grandi et al, 2004), and represents the contribution of the present work. The XML data model on which it is based includes semantic annotations in the multi-versioning mechanism, in order to capture limited applicability and to support personalized access. The prototype has been implemented following a "native" approach and is currently under evaluation. Conclusions will finally be found in Section 4.

2. TEMPORAL VERSIONING AND THE "STRATUM" APPROACH

In a first phase of our research we focused on temporal aspects and on the effective and efficient management of time-varying norm texts. To this purpose, we developed a temporal XML data model which uses four time dimensions to correctly represent the evolution of norms in time and their resulting versioning. The considered dimensions are:

- **Validity time.** It is the time the norm is in force. It has the same semantics of valid time as in temporal databases (Jensen et al, 1998), since it represents the time the norm actually belongs to the regulations in the real world.
- **Efficacy time.** It is the time (some part of) the norm can be applied to a concrete case. It usually corresponds to validity, but it can be the case that an abrogated norm continues to be applicable to a limited number of cases. Until such cases cease to exist, the norm continues its efficacy though no longer in force. It also has a semantics of valid time, although it is *independent* from validity time.
- **Transaction time.** It is the time (some part of) the norm is stored in a computer system. It has the same semantics of transaction time as in temporal databases (Jensen et al, 1998).
- **Publication time.** It is the time of publication of the norm on the Official Journal. It has the same semantics as event time in temporal databases (Kim and Chakravarthy, 1993). It is a global and unchangeable property for the whole norm contents and, thus, it is not used as a versioning dimension.

The data model was defined via an XML Schema (XMLSchema, 2004), where the structure of norms is defined by means of a contents-section-article-paragraph hierarchy and multiple content versions can be defined at each level of the hierarchy. Each version is characterized by timestamp attributes defining its temporal pertinence with respect to each of the validity, efficacy and transaction time dimensions.

Legal text repositories are usually managed by traditional information retrieval systems where users are allowed to access their contents by means of keyword-based queries expressing the subjects they are interested in. We extended such a framework by offering to users the possibility of expressing temporal

specifications for the reconstruction of a consistent version of the retrieved normative acts (*consolidated act*). More precisely, the system is able to answer queries in the following XQuery (XQuery, 2004) form:

```
FOR $a IN path
   WHERE constraints on $a
   RETURN const-tree(document($a), temporal specs)
```

Such a statement, following the standard FLWR syntax, allows users to express selection constraints on the variable **$a** iterating over the nodes returned by the path expression **path**. Search keywords can be specified by means of the function **contains** in the **WHERE** clause (e.g. **contains($a,'sea')**). In the **RETURN** clause, the operator **const-tree** is devoted to the reconstruction of the temporally consistent versions of the XML documents containing the selected nodes. The *temporal specs* expression is the conjunction of temporal selection predicates on the four supported temporal dimensions. Our approach is the first to provide full search and reconstruction functionalities with respect to all time dimensions, whereas previous approaches only provided a limited support. For example, the temporal XML markup adopted in the Norma-System described in (Palmirani and Brighi, 2002) includes publication, validity and efficacy time but reconstruction of consolidated versions is made with respect to validity only (other time dimensions can be used as additional search fields in full-text search).

The model is also equipped with two basic operators for the management of norm modifications: one is for changing the textual content of a norm portion and the other is for changing the temporal pertinence of a given version. The former can be used for deletion of (a part of) the norm (*abrogation*), or the introduction of a new part of the norm (*integration*), or the replacement of (a part of) the norm (*substitution*). The latter can be used to deal with the temporal *extension* or the *suspension* of (part of) the norm.

Our temporal data model with the modification and query operators was implemented in a prototype system for the management and maintenance of a collection of time-varying norms. The system is able to store norms encoded as XML documents and efficiently access them by answering queries which can involve both temporal constraints and search keywords. The system architecture is based on two different components: the former consists of the XML document management facilities offered by Oracle 9i (Oracle, 2004) to handle structural and textual constraints, the latter is a software stratum that we built on top of the former to handle the temporal aspects. Extensive experimental results on the system behavior show good performance and the ability to manage large collections of XML multi-version documents. A discussion of such architectural solution, named the “stratum” approach, in comparison with our new implementation solution, named the “native” approach, is carried out in Section 3. A detailed description of the “stratum” approach and an account of its evaluation can be found in (Grandi et al, 2003a; Grandi et al, 2005).

3. SEMANTIC VERSIONING AND THE “NATIVE” APPROACH

In a second phase of the research, the multi-version model based on temporal dimensions was extended to include a semantic versioning dimension in order to provide personalized access to norm texts..

In general, machine-understanding of the information available on the Semantic Web requires a semantic markup of the contents and the availability of automated reasoning tools. In order to let information and its interpretation be shared by several agents including automatic tools, the introduction of common reference *ontologies* becomes necessary (Guarino, 1998; OWL, 2004). In our case, we defined a *civic ontology*, which corresponds to a classification of citizens based on the distinctions introduced by successive norms (*founding acts*) that imply some limitation, total or partial, in their applicability. Hence, in our extended model, the new versioning dimension encodes information about the applicability of different parts of a norm text to the relevant classes of the civic ontology.

Consider, for instance, Fig. 1. The left part of the figure depicts a simple civic ontology built from a small corpus of norms ruling the status of citizens with respect to their work position. The right part of the figure shows a fragment of a multi-version XML norm text supporting personalized access with respect to this ontology. Notice that, at this stage of the project, we manage “tree-like” ontologies, defined as class taxonomies induced by the *IS-A* relationship. This allows us to exploit the pre-order and post-order properties of trees in order to enumerate the nodes and check ancestor-descendant relationships between the classes in order to efficiently process queries; such codes are displayed in the upper left corner of the ontology classes in the Figure, in the form: (pre-order,post-order).

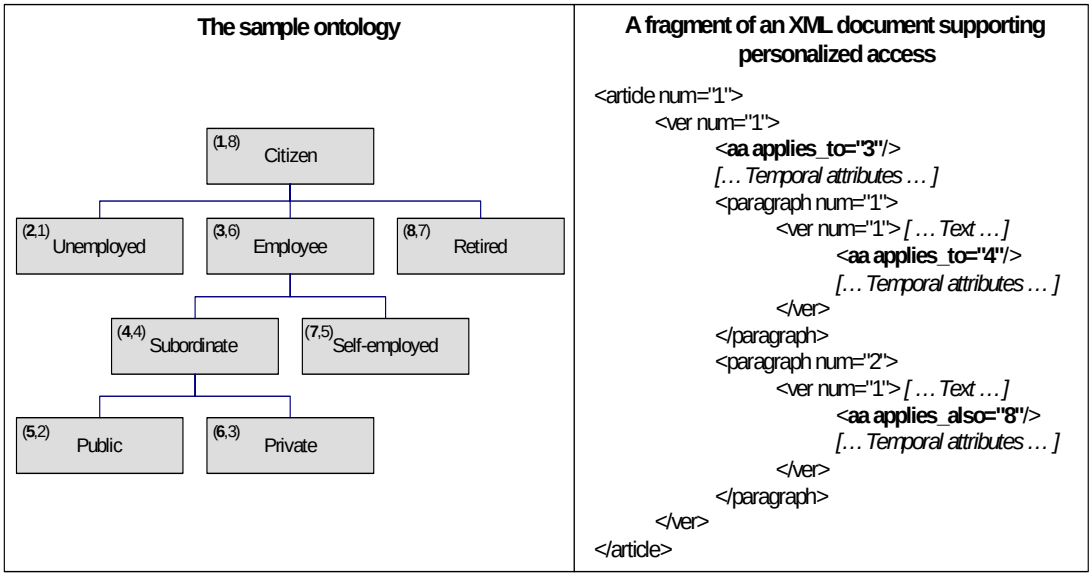

Fig. 1. On the left, an example of civic ontology, where each class has a name and is associated to a (pre,post) pair; on the right, a fragment of an XML norm containing applicability annotations

For instance, the class "Employee" has pre-order "3" which is also its identifier, whereas its post order is "6". The article in the XML fragment on the right-hand-side of Fig. 1 is composed of two paragraphs and contains applicability annotations (tag **`aa`**). Notice that applicability is inherited by descendant nodes unless locally redefined. Hence, by means of redefinitions we can also introduce, for each of part of a document, complex applicability properties including extensions or restrictions with respect to ancestors. For instance, the whole article in the Figure is applicable to civic class "3" (tag **`applies_to`**) and by default to all its descendants. However, its first paragraph is applicable to class "4", which is a restriction, whereas the second one is also applicable to class "8" (tag **`applies_also`**), which is an extension. The reconstruction of pertinent versions of the norm based on its applicability annotations is very important in an e-Government scenario. The representation of extensions and restrictions gives rise to high expressiveness and flexibility in such a context.

As to the queries that can be submitted by a user in the new system, they can contain four types of constraints: temporal, structural, textual and applicability. Such constraints are completely orthogonal and allow the user to perform very accurate searches in the XML norm repository. Let us focus first on the applicability constraint. Consider again the ontology and norm fragment in Fig. 1 and let John Smith be a self-employed citizen (i.e. belonging to class "7") accessing the norm: hence, the sample article in the Figure will be selected as pertinent, but only the second paragraph will be actually presented as applicable. Furthermore, the applicability constraint can be combined with the other three ones in order to fully support a multi-dimensional retrieval. For instance, John Smith could be interested in all the norms ...

- which contain paragraphs (*structural constraint*) dealing with health care (*textual constraint*), ...
- which were valid and in effect between 2002 and 2004 (*temporal constraint*), and...
- which are applicable to his personal case (*applicability constraint*).

Such a query can be issued to our system using the standard XQuery FLWR syntax as follows:

```
FOR $a IN path
   WHERE textConstr ($a//paragraph//text(), 'health AND care')
     AND tempConstr ('vTime OVERLAPS PERIOD(2002-01-01,2004-12-31)')
     AND tempConstr ('eTime OVERLAPS PERIOD(2002-01-01,2004-12-31)')
     AND applConstr ('class_7')
   RETURN $a
```

where **`textConstr`**, **`tempConstr`**, and **`applConstr`** are suitable functions allowing the specification of the textual, temporal and applicability constraints, respectively (the structural constraint is implicit in the XPath expressions used in the XQuery statement). Notice that the temporal constraints can

involve all the four available time dimensions (publication, validity, efficacy and transaction), allowing high flexibility in satisfying the information needs of citizens in the e-Government scenario. In particular, by means of validity and efficacy time constraints, a user is able to extract consolidated current versions from the multi-version repository, or to access past versions of particular norm texts, all consistently reconstructed by the system on the basis of the user's needs and personalized on the basis of his/her identity. The citizen's digital identity is defined as the total amount of information concerning him/her, which is necessary for the sake of classification with respect to the ontology (Grandi et al, 2004). All such information must be retrievable in an automatic way from the PA databases. To this purpose, facilities for querying PA databases must be provided and implemented through standardized access services. Matching between the citizen's identity and the ontology classes is then made via a suitable reasoning service embedded in the system.

2.1 Implementation and performances

All the multi-version and personalized-access XML norm querying features have been implemented in our second prototype system.

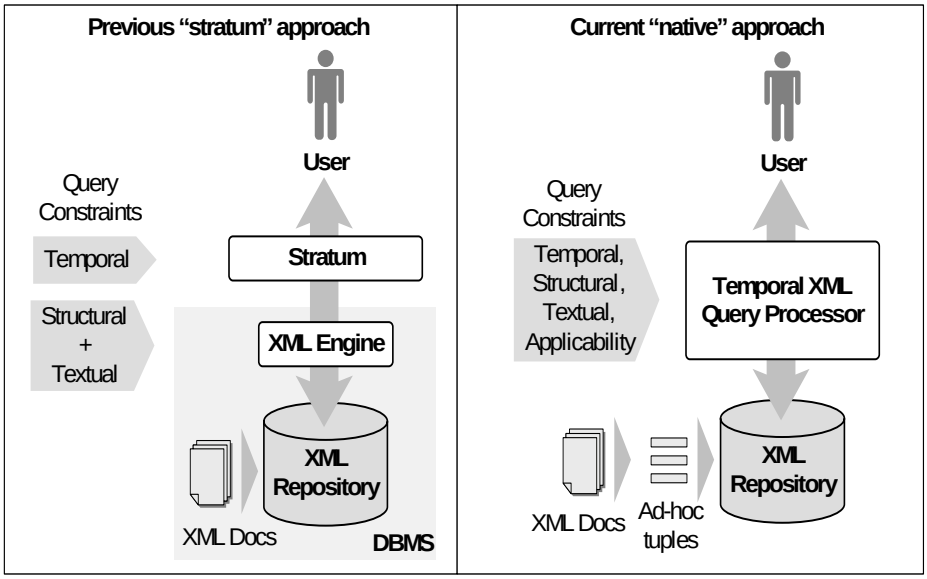

Fig. 2. First ("stratum") versus second ("native") system architecture

The system architecture is shown in the right-hand side of Fig. 2 and is based on an XML-native approach, since it composed of a Temporal XML Query Processor designed on purpose, which is able to manage the XML data repository and to provide all the temporal, structural, textual and applicability query facilities in a single component. The prototype is implemented in Java JDK 1.5 and exploits ad-hoc data structures (relying on embedded "light" DBMS libraries) and algorithms which allows users to store and reconstruct on-the-fly XML norm texts satisfying the four types of constraints. Differently from the "stratum" approach we used in our previous prototype (see the left part of Fig. 2), where temporal constraints were processed separately, all the structural, textual and temporal constraints are simultaneously handled by the Temporal XML Query Processor. Such a component stores the XML norms not as entire documents but by converting them into a collection of ad-hoc temporal tuples, representing each of its multi-version parts (i.e. paragraphs, articles, and so on); these data structures are then exploited to efficiently perform structural join algorithms (Al-Khalifa et al, 2002) we specifically devised and tuned for the temporal/semantic multi-version context. Textual constraints, like in the "stratum" approach, are handled by means of an inverted index. The benefits of our "native" approach over the "stratum" one are manifold:

- by querying ad-hoc and temporally-enhanced structures (which have a finer granularity than the entire documents managed by standard XML engines), the "native" approach is able to access and retrieve only the strictly necessary data;

- only the parts which are required and which satisfy the temporal constraints are used for the reconstruction of the retrieved documents;
- there is no need to retrieve whole XML documents and build space-consuming structures such as DOM trees, as required in the "stratum" approach.

As a consequence, we expected that the query processing efficiency can be greatly enhanced and the memory requirements dramatically reduced. In order to evaluate the effectiveness of the "native" approach, we compared its performance with our previous "stratum" implementation on a common query benchmark and also conducted a number of exploratory experiments to analyze its behavior in performing personalized access through applicability constraints. The experiments have been effected on a Pentium 4 2.5Ghz Windows XP Professional workstation, equipped with 512MB RAM and a RAID0 cluster of 2 80GB EIDE disks with NT file system (NTFS). We performed the tests on three XML document sets of increasing size: collection C1 (5,000 XML norm text documents), C2 (10,000 documents) and C3 (20,000 documents). In this paper, due to space requirements, we will present in detail the results obtained on the collection C1, then we will briefly describe the scalability performance shown on the other two collections. The total size of the collections is 120MB, 240MB, and 480MB, respectively. In all collections the documents were synthetically generated by means of an ad-hoc XML generator we developed, which is able to produce different documents compliant to our multi-version and personalized access model. For each collection the average, minimum and maximum document sizes are 24KB, 2KB and 125KB, respectively.

Table 1. Query execution time of the "stratum" and "native" approaches (time in milliseconds, collection C1)

Query	Constraints				Selectivity	Performance (msec)	
	Tm	St	Tx	Ap		Stratum	Native
Q1	–	✓	✓	–	0.6%	2891	1046
Q2	–	✓	✓	–	4.02%	43240	2970
Q3	✓	✓	–	–	2.9%	47638	6523
Q4	✓	✓	✓	–	0.68%	2151	1015
Q5	✓	✓	✓	–	1.46%	3130	2550
Q1–A	–	✓	✓	✓	0.23%	n/a	1095
Q2–A	–	✓	✓	✓	1.65%	n/a	3004
Q3–A	✓	✓	–	✓	1.3%	n/a	6760
Q4–A	✓	✓	✓	✓	0.31%	n/a	1020
Q5–A	✓	✓	✓	✓	0.77%	n/a	2602

Experiments were conducted by submitting queries of five different types (Q1–Q5). Table 1 presents the features of the test queries and the query execution time for both the "stratum" and the "native" architectures. All the queries require structural support (St constraint); types Q1 and Q2 also involve textual searches by keywords (Tx constraint), with different selectivities; type Q3 contains temporal conditions (Tm constraint) on three time dimensions: transaction, valid and publication time; types Q4 and Q5 mix the previous ones since they contain both keyword searches and temporal conditions. For each of those query types, we also present a personalized access variant involving an additional applicability constraint (denoted as Qx–A in Table 1. Let us first focus on the upper part of the table, and in particular on the comparison of the performances offered by the two approaches. The "native" approach shows to be faster in every context, providing a shorter response time (including query analysis, retrieval of the qualifying norm parts and reconstruction of the result) of approximately one or two seconds for most of the queries. Notice that, while the response time of the "stratum" approach is not too different for query types Q1, Q4, Q5, for the other query types the performance gap is quite important (for instance, query Q2 is answered approximately 15 times slower in the "stratum" approach). The reason is that the selectivity of the query predicates strongly influences the performance of the "stratum" approach, seriously impairing its performance when large amounts of documents containing some (typically small) relevant portions have to be retrieved, as it happens for queries Q2 and Q3. On the other hand, the "native" approach is able to deliver a faster and more reliable performance in all cases, since it practically avoids the retrieval of useless document parts. Further, consider that, for the same reasons, the main memory requirements of the Temporal XML Query Processor embedded in the "native" approach are, on average, 5% or less of the "stratum" approach. This property is also very promising towards future extensions to cope with concurrent multi-user query processing. The lower part of the table presents the performance of our second system with respect to the queries involving additional

applicability constraints, enabling personalized access. Thanks to the properties of the adopted pre-order and post-order encoding of the civic classes, the system is able to solve personalization problems by means of simple comparisons involving such encodings and, thus, with a very high efficiency. The time needed to answer the personalized access versions of the Q1–Q5 queries is approximately 0.5–1% more than for the original versions. Moreover, since the applicability annotations of each part of an XML document are stored as simple integers, also the size of the applicability annotated tuples, as stored in the system, is practically unchanged (only a 3–4% storage space overhead is required with respect to documents without semantic versioning), even with quite complex annotations involving several applicability extensions and restrictions.

Finally, we only report a comment about the performance of our current prototype in querying the other two collections C2 and C3 and, therefore, concerning the scalability of the system. We ran the same queries of the previous tests on the larger collections and saw that the computing time always grew sub-linearly with the number of documents. For instance, query Q1 executed on the 10,000 documents of collection C2 (which is as double as C1) took 1,366 msec (i.e. the system was only 30% slower); similarly, on the 20,000 documents of collection C3, the average response time was 1,741 msec (i.e. the system was less than 30% slower than with C2). Also with the other queries the measured trend was the same, thus showing the good scalability of the system in every type of query context.

4. CONCLUSION

In this paper we presented our research work concerning the design and implementation of efficient Web-based information systems for e-Government applications. Recent activities include the development of a platform (“stratum” approach) for temporal management of multi-version norm texts on top of a commercial DBMS and the migration of such a system towards a more efficient platform (“native” approach) for which a specialized Temporal XML Query Processor has been designed. The new system also offers advanced functionalities, as it provides a personalized access to resources on the basis of the digital identity of citizens. While the first system employs temporal database techniques for the management and maintenance of multi-version XML data, the second system also employs Semantic Web techniques, including the adoption of an ontology, for the management of applicability constraints and personalized access. Preliminary experimental work on query performance, with repositories of synthetic XML documents, showed encouraging results. In particular, the “native” approach proved to be very efficient in a large set of experimental situations and showed excellent scale-up figures with varying load configurations.

Future work will consider the improvement of the approach to cope with more advanced application requirements and the completion of the technological infrastructure required with the implementation of auxiliary services (e.g. for automatic classification of logged-on citizens with respect to the civic ontology). Further work will also include the assessment of our developed systems in a concrete working environment, with real users and in the presence of a large repository of real legal documents.

ACKNOWLEDGEMENT

This work has been supported by the MIUR-PRIN Project: “The European citizen in e-Governance: philosophical-juridical, legal, information and economic profiles”.

REFERENCES

Grandi, F. et al, 2003a. A Temporal Data Model and Management System for Normative Texts in XML Format. *Proceedings of the 15th ACM Intlernational Workshop on Web Information and Data Management (WIDM)*, New Orleans, LA, pp. 29–36.

Grandi, F. et al, 2003b. A Temporal Data Model and System Architecture for the Management of Normative Texts. *Proceedingsof the 11th Natlional Conf. on Advanced Database Systems (SEBD)*, Cetraro, Italy, pp. 169–178.

Grandi, F. et al, 2004. Management of the Citizen's Digital Identity and Access to Multi-version Norm Texts on the Semantic Web. *Proceedingsof the International Symposium on Challenges in the Internet and Interdisciplinary (IPSI 2004)*, Pescara, Italy.

Grandi, F. et al, 2005. Temporal Modelling and Management of Normative Documents in XML Format. *Data & Knowledge Engineering*, Vol. 47 (in press).

Al-Khalifa, S. et al, 2002. Structural Joins: A Primitive for Efficient XML Query Pattern Matching. *Proceedingsof 18th International Conference on Data Engineering (ICDE*), San Jose, CA, pp. 141–154.

EC E-Gov, 2004. European Commission e-Government Home Page. http://europa.eu.int/information_society/eeurope/2005/all_about/egovernment/index_en.htm

Guarino, N., editor, 1998. *Formal Ontology in Information Systems*. IOS Press, Amsterdam, The Netherlands.

Jensen, C. S. et al, 1998. The Consensus Glossary of Temporal Database Concepts - February 1998 Version. In Etzion, O., Jajodia, S., and Sripada, S., editors. *Temporal Databases –- Research and Practice*. Springer-Verlag. LNCS No. 1399, pp. 367–405.

Kim, S.-K. and Chakravarthy, S. 1993. Modeling Time: Adequacy of Three Distinct Time Concepts for Temporal Data. *Proceedings of 12th International Conerence. on Entity-Relationship Approach (ER)*, Arlington, TX, pages 475–491.

Oracle, 2004. The Oracle 9i Database Home Page. Oracle Corporation, http://www.oracle.com/technology/products/oracle9i/

Palmirani, M. and Brighi, R., 2002. Norma-system: A Legal Document System for Managing Consolidated Acts. *Proceedings of 13th International Conference on Database and Expert Systems Applications (DEXA)*, Aix-en-Provence, France, pages 310–320.

Protégé, 2004. The OWL Plugin for Protégé Home Page. Stanford University, http://protege.stanford.edu/plugins/owl/

US E-Gov, 2004. U.S. President's e-Government Initiatives Home Page. http://www.whitehouse.gov/omb/egov/

WebOnt, 2004. The Web Ontology Group Home Page. W3C Consortium, http://www.w3.org/2001/sw/WebOnt/

XML, 2004. The eXtensible Markup Language Home Page. W3C Consortium, http://www.w3c.org/XML/

XMLSchema, 2004. The XML Schema Home Page. W3C Consortium, http://www.w3c.org/XML/Schema/

XQuery, 2004. The XML Query Home Page.W3C Consortium, http://www.w3c.org/XML/Query