A temporal data model and system architecture for the management of normative texts (Extended Abstact) *

Fabio Grandi[1], Federica Mandreoli[2], Paolo Tiberio[2], and Marco Bergonzini[2]

[1] DEIS, Alma Mater Studiorum - Università di Bologna, Italy
fgrandi@deis.unibo.it
[2] DII, Università di Modena e Reggio Emilia, Italy
{mandreoli.federica,tiberio.paolo,bergonzini.marco8548}@unimo.it

Abstract. In this paper, we present the preliminary results of an ongoing research activity concerning the temporal management of normative texts in XML format. In particular, four temporal dimensions (publication, validity, efficacy and transaction times) are used to correctly represent the evolution of norms in time and their resulting versioning. Hence, we introduce a multiversion data model based on XML schema and define three basic operators for the management of norm texts. Finally, we describe the architecture of a management system prototype which is being implemented.

1 Introduction

Time is one of the main aspects characterizing several real world facets and phenomena. The ability to model this temporal dimension of the real world and to respond within time constraints to changes in the real world as well as to application-dependent operations is essential to many computer applications. The management of norms represents one of such applications as temporal concerns are ubiquitous in the law domain [16]. Time in normative systems has become a central topic of the cultural and political debate and is of fundamental concern to legal informatics. The law is under increasing pressure to keep pace with social change: normative texts and amendments follow one another in time and get overlapped.

In the context of database research, the management of time has been extensively studied in the last decades [20, **?**]. In particular, many efforts have been devoted to add time support to database models and system functionalities. Temporal database systems are database systems that include special support for the time dimension; in other words, they are systems that provide special facilities for storing, querying, and updating historical and/or future data. In this context, two time dimensions are usually considered: valid time and transaction

* This work has been partially supported by the MIUR-40% Project: "La dinamica della norma nel tempo: aspetti giuridici ed informatici".

time [10]. Valid time is the time of the real world. It denotes the time a fact is true in reality. Transaction time is the time of the system and it denotes the time during which the fact is present in the database as stored data. In order to make a more complete picture, two other temporal dimensions have been considered useful for advanced applications: event time [11] (also called decision time in [13]), which is the occurrence time of events that initiate and/or terminate the validity of some fact in the real world, and availability time [4], which is the time some fact is available in the information system.

Moreover, in the database research community, there is a much current interest in representing and querying semi-structured data. For example, database-resident data can be published as static or dynamic XML documents, which can then be viewed on Web browsers and processed by various Web based applications, including queries written in languages such as XPath and XQuery [21]. As a consequence, temporal aspects began to be considered also in works on semi-structured data. Such approaches will be described in the next section.

In this paper, we present some preliminary results of the research activity we are carrying out in the context of the MIUR-40% Project "La dinamica della norma del tempo: aspetti giuridici ed informatici". In this context, the main objective of our work is the development of an effective and efficient system for the management of norms in time represented as XML documents and made available on the Web. To this end, in Section 3 we introduce a temporal XML data model for the representation and management of versioned normative texts. In Section 4, we give a brief overview of the system architecture for the implementation and efficient exploitation of our data model.

2 Related Work

Several works took into account change, versioning, evolution and also explicitly temporal aspects, in semi-structured and XML-based data management, often applying conceptual tools and techniques developed by temporal database research.

The main focus of some approaches was on the representation and management of changes, where different versions of data are produced by updates. In this approach, temporal attributes are often used to timestamp stored versions (e.g. [3, 1]). They represent the time the updates were applied and, thus, have the (implicit) semantics of transaction time with respect to the system where the changes are effected.

Other approaches considered the classical notion of valid time (e.g. [9, 22]). For example, the "Valid Web" approach [8] is an XML/XSL infrastructure designed to represent and manage temporal Web documents (i.e. documents containing historical information, with timestamps explicitly encoded by the document authors to assign validity to information contents). Temporal documents can then be selectively browsed, in accordance with a user-supplied temporal period of interest.

Some approaches also considered a bitemporal data model, that is supporting both valid and transaction time (e.g. [7, 12]).

Other papers also considered a Web usage specific temporal dimension: navigation time, which concerns the interaction of users during their browsing of Web sites (e.g. [2, 5]).

In the context of legal computer science, previous approaches already dealt with the reconstruction of consolidated norm texts, consisting of their current temporal version [17]. One temporal dimension was considered in such approaches.

3 Modelling Time and Norms

In this section, we present the XML-based temporal data model we propose for the representation and management of versioned normative texts. Multiple temporal dimensions, all involved in the law application lifecycle, are considered.

The model is based on a hierarchical organization of normative texts (i.e. legal norms can be based on a contents-section-article-paragraph hierarchy), which corresponds to a tree-like inner structure and is particularly suitable to XML encoding. Such an encoding is enriched with timestamping metadata modelling the temporal aspects of normative texts. In accordance with the hierarchical structure, the temporal extension we devised supports ancestor-descendant inheritance, that is temporal attributes (timestamps) of a node N are inherited by its child nodes, unless redefined. The time dimensions we consider are the following:

Publication time. It is the time of publication of norms on the Official Journal. It has the same semantics as event time (and availability time, as the two time dimensions, in such a context, coincide).

Validity time. It is the time the norm is in force (in the Italian regulations, usually a norm is in force from the publication date plus 15 days on, until its validity is changed by a subsequent act). It has the semantics of valid time, as it represents the time the norm actually belongs to the regulations in the real world.

Efficacy time. It is the time the norm can be applied to a concrete case. It usually corresponds to the validity of norms, 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.

Transaction time. It is the time the norm is stored in our computer system. Obviously, it has the same semantics of transaction time as in temporal databases.

All the above dimensions are independent and thus can be treated in an "orthogonal" way.

The temporal model is defined by means of an XML Schema [23]. It comes out as a quite unfaithful translation and extension of one of the DTDs published by the "Norma in Rete" (Norm on Network) [14] working group. The "Norma

Fig. 1. The XML-schema for the representation of norms in time

in Rete" initiative has been jointly promoted by the Italian agency for the introduction of information technology in public administrations (AIPA) and the Ministry of Justice, which is framed to supply the missing link between information technology and the public administration in the field of legimatics. The project has a major role in trying to overcome the fragmentation affecting online availability of norms by means of the introduction of standards, mainly based on XML-related technologies, for publication and exchange of norms on the network. Although that working group developed a DTD for the XML encoding of normative texts, also providing for the representation of multiple *versions*, we decided not to follow their approach by adopting the already available DTD. The motivation of our decision arose from the complex way of representing versions in the "Norma in Rete" approach, which involves different levels of indirection and, thus, in our opinion, does not seem amenable to an efficient management. On the contrary, taking into account the efficient management requirement, we developed an alternative XML encoding scheme for normative text based on an XML-schema, whose full version is depicted in Figure 1 ("R" and "O" near attribute names stand for required and optional, respectively). In our modeling approach, the meta-level of normative texts is rooted by the `norm` element which is characterized by a number and type attribute values and includes a `title` and `contents` elements. The XML schema allows us to mark-up normative acts with the publication date which is a property of the overall document and thus it has also been modelled as an attribute associated to the outermost element. Then, at each level of the contents-section-article-paragraph hierarchy it is possible to represent different versions (by means of the `ver` element) characterized by their timestamps associated to the three other temporal dimensions. The active norm reference `an_ref` is the identifier of the modifying norm whose enforcement caused the versioning.

Figure 2 shows a document conforming to the temporal XML schema. The "L247/1999" Law concerns the cereal importation and contains two sections, three articles and four paragraphs. It has been published on 1999/12/15 and is valid from 2000/1/1 (it has been recorded in the system on 2000/1/10). Only (two) paragraphs underwent punctual modifications and thus have more than one versions. For this reason, all parts but paragraphs inherit the timestamps from the `contents` tag. For the paragraphs, instead, it is necessary to explicit the temporal attributes since they are redefined by the corresponding versions. Paragraph 2 of Sec. 1, Art. 1 has been modified by the "LD135/2000" Legislative Decree, in force since 2000/6/1 (modification recorded on 2000/6/10). Paragraph 1 of Sec. 2, Art. 1 has been modified by the "L107/2001" Law, in force since 2001/7/5 (modification recorded on 2001/7/15).

Notice that, in the former case the old version continues to be applicable (e.g. to the cases for which it was applicable before the modification), whereas in the latter case the modifying Law has stated that the old version is definitely no longer applicable (hence, efficacy time has been stopped to 2001/7/5 like validity). In both cases, the presence of two versions (number 1.1 and 1.2) with equal contents is formally necessary to correctly map the temporal pertinence of the text before the modification on a bitemporal space (transaction × validity/efficacy).

The model is provided with three basic operators:

– $O_1(vt, et, tt)$ for the reconstruction of a temporally consistent normative act; it requires as arguments a temporal value (time point) for each time dimension and reconstructs the document by selecting –at each level of the hierarchy– the text making up the desired version (if it exists). Valid time (vt) is the only required argument, while defaults for efficacy and transaction time are $et = vt$ and $tt = \text{now}$, respectively.
– $O_2(en, vt_s, vt_e, et_s, et_e, txt, an)$ for the lossless update of documents (corresponding to an explicit textual substitution); it requires the name of the element to be substituted (en), the validity and efficacy timestamps to be assigned to the new version ($[vt_s, vt_e]$ and $[et_s, et_e]$), the new text (txt) and a reference to the active (i.e. modifying) norm (an). It applies a lossless sub-

```
<norm num="247" type="Law" publication="1999-12-15"
id="L247/1999">
   <title> Cereals importation </title>
   <contents vt_start="2000-01-01" et_start="2000-01-01" tt_start="2000-01-10" >
      <ver num="1">
         <section num="1">
            <ver num="1"> <heading> Import from Communitarian countries </heading>
                <article num="1">
                   <ver num="1"> <heading> Import from Spain </heading>
                      <paragraph num="1">
                         <ver num="1" > Sec. 1 Art. 1 Par. 1 (unmodified) </ver>
                      </paragraph>
                      <paragraph num="2">
                         <ver num="1.1" vt_start="2000-01-01" et_start="2000-01-01"
                              tt_start="2000-01-10" tt_end="2000-06-10"
                              an_ref="LD135/2000"> Sec. 1 Art. 1 Par. 2 before modification </ver>
                         <ver num="1.2" vt_start="2000-01-01" vt_end="2000-06-01"
                              et_start="2000-01-01" tt_start="2000-06-10"
                              an_ref="LD135/2000"> Sec. 1 Art. 1 Par. 2 before modification </ver>
                         <ver num="2" vt_start="2000-06-01" et_start="2000-06-01" tt_start="2000-06-10"
                              an_ref="LD135/2000"> Sec. 1 Art. 1 Par. 2 after modification </ver>
                      </paragraph>
                   </ver>
                </article>
                <article num="2">
                   <ver num="1"> <heading> Import from other EC countries </heading>
                      <paragraph num="1">
                         <ver num="1"> Sec. 1 Art. 2 Par. 1 (unmodified) </ver>
                      </paragraph>
                  </ver>
                </article>
            </ver>
         </section>
         <section num="2">
             <ver num="1"> <heading> Import from countries outside the European Community </heading>
                 <article num="1">
                    <ver num="1">
                        <paragraph num="1">
                           <ver num="1.1" vt_start="2000-01-01" et_start="2000-01-01"
                                tt_start="2000-01-10" tt_end="2001-07-15"
                                an_ref="L107/2001"> Sec. 2 Art. 1 Par. 1 before modification </ver>
                           <ver num="1.2" vt_start="2000-06-01" vt_end="2001-07-05"
                                et_start="2000-06-01" et_end="2001-07-05" tt_start="2001-07-15"
                                an_ref="L107/2001"> Sec. 2 Art. 1 Par. 1 before modification </ver>
                           <ver num="2" vt_start="2001-07-05" et_start="2000-07-05" tt_start="2001-07-15"
                                an_ref="L107/2001"> Sec. 2 Art. 1 Par. 1 after modification </ver>
                        </paragraph>
                    </ver>
                 </article>
             </ver>
         </section>
     </ver>
  </contents>
</norm>
```

Fig. 2. An example of multiversion XML document

stitution by adding a new version of the element and by also updating the timestamps of the modified version, and of the ancestor elements in order to comply with the inheritance semantics if necessary. Notice that if the new text is empty the O_2 application corresponds to an abrogation.

– $O_3(en, vt, et, vt_s, vt_e, et_s, et_e, an)$ for the modification of the timestamps of an existing document portion (e.g. corresponding to an extension of validity); it requires the name of the concerned element (en) and the temporal coordinates of the version to be modified (vt and et, while now is used for transaction time to select the current version), the new values of the validity and efficacy timestamps ($[vt_s, vt_e]$ and $[et_s, et_e]$) and a reference to the active norm (an).

Notice that, for the operators O_2 and O_3, the new transaction-time timestamps are automatically generated by the system, with tt_s equal to the instant the operator is applied (now) and tt_e undefined.

Fig. 3. The overall system architecture

4 System Architecture

In this section, we describe the architecture of the system prototype we are developing for the implementation and efficient exploitation of our data model. The final version of the system for the management of norms in time should be able to store norms encoded in XML documents and efficiently access them by means of selections which could also involve temporal constraints and keywords. Moreover, the system must be available on line and easily accessible through a Web interface.

In the state of art of technology, there are two main solutions for the management of large collections of XML documents: the adoption of a native XML data server or of a relational DBMS provided with XML document management facilities. The former solution (e.g. use of Tamino [19]) relies on ad-hoc access and storage structures to manage XML repositories. Even if in the future they could become a valid alternative, such systems do not currently guarantee the same reliability, robustness and performance levels of the mature relational technology. The latter solution includes widely used DBMSs like Oracle 9i [15], DB2 [6] and SQL Server [18]. The main goal of such an approach is to add XML to the supported data types and allow XML data to be queried and interoperated with relational data. Among the above cited systems, to the best of our knowledge, the only system which provides an adequate support to XML data is Oracle

9i. In particular, it provides two ways for storing XML documents: they can be simply stored as text in a table column or split into several tables containing different levels with respect to their tree structure. Then, XML documents can be queried by means of a query language which is an XPath dialect.

In this context, our choice fell on the DBMS solution, also because it is quite likely that prospective users of our system (e.g. public administrations) already own a DBMS used for other purposes and, thus, do not have to buy and install a different system to manage also the XML legal stuff. Moreover, Oracle 9i was chosen for the development of our prototype, because it was a good candidate to efficiently support the kind of temporal queries we need. The overall architecture is depicted in Fig. 3.

The system stores XML documents into a table where additional columns are devoted to store metadata (in particular, timestamping metadata). More precisely, the table `tnorms` has the following schema:

```
tnorms( ID, XML-DOC, TYPE, PUBLICATION,
        VT-START, VT-END, TT-START, TT-END, ET-START, ET-END )
```

Every tuple in this relation represents a temporal XML document. The timestamp attributes have the same value of the timestamping tags associated with the `contents` tag, which represent the time values of the whole norm text. The addition of such metadata is aimed at improving the efficiency of query execution by introducing a preliminary document filtering phase. The encoding of norm texts into temporal XML documents is based on the data model described in the previous Section and, thus, stored documents conform to the XML-Schema in Fig. 1. Moreover, in order to further speed up the processing of temporal queries, all the timestamps, even when they could be derived owing to the semantics of inheritance, are explicited at every level of the document hierarchy. In this way, we can fully exploit the potentialities of the XML query language provided by Oracle 9i, when conditions on tag and attribute values are specified in queries at different levels of the document tree structure. The extraction of document metadata and the explicitation of timestamps are performed by the preprocessing module shown in Fig. 3.

In order to support norm retrieval by keywords, an inverted index has been built on the contents of selected XML elements (`title`, `heading` and `paragraph`).

When a query including a temporal predicate is submitted to the system, the temporal part of the query is converted into calls to the O_1 operator which are executed by the query processor module. The query processor translates the request into a SQL query involving metadata columns, which filters out the tuples of the `tnorms` table which do not qualify for the temporal conditions. For each qualifying tuple, a norm reconstruction phase follows, during which the only parts which satisfy the temporal constraints are extracted. Preliminary experimental results concerning pure temporal queries and mixed queries, including both temporal predicates and search by keyword, show how the exploitation of the metadata columns in the `tnorms` table yields even a three order of magnitude speed up with respect to issuing the query on the `XML-DOC` column only. For instance, for a temporal query on 1000 documents (about 100MB of data) the execution time drops from 178 seconds to 0.18 second. The difference is even more apparent for 3000 documents where the execution time drops from more than 485 seconds to 0.29 second.

Finally, as to the management of change that normative texts undergo during their life-cycle, the maintenance of the XML repository is managed by the update processor (implementing the O_2 and O_3 operators).

5 Conclusions

In this paper, we introduced a temporal XML data model which is able to manage the dynamics of norms in time and represent their multiple versions with respect to publication, validity, efficacy and transaction times. The model is based on an XML schema which allows the introduction of timestamping metadata at each level of the document structure which is subject to change, up to the granularity of a single paragraph. Moreover, it is equipped with three operators, one for the reconstruction of a consistent temporal version and the other two for the management of textual and temporal changes. Finally, the architecture of a prototype relying on the Oracle 9i DBMS has been outlined. The prototype is under development and evaluation at the University of Modena and Reggio Emilia. The preliminary experimental results on query performance are encouraging.

References

1. Toshiyuki Amagasa, Masatoshi Yoshikawa, and Shunsuke Uemura. A data model for temporal xml documents. In *Proc. of 11th Intl' Conf. on Database and Expert Systems Applications (DEXA 2000)*, London, England, September 2000.
2. Jesper Andersen, Anders Giversen, Allan H. Jensen, Rune S. Larsen, Torben Bach Pedersen, and Janne Skyt. Analyzing clickstreams using subsessions. In *Proc. of 3rd ACM Intl' Workshop on Data Warehousing and OLAP (DOLAP 2000)*, Washington, DC, November 2000.
3. Sudarshan S. Chawathe, Serge Abiteboul, and Jennifer Widom. Managing historical semistructured data. *Theory and Practice of Object Systems*, 5(3):143–162, 1999.
4. Carlo Combi and Angelo Montanari. Data models with multiple temporal dimensions: Completing the picture. In *Proc. of 13th International Conference on Advanced Information Systems Engineering (CAiSE 2001)*, Interlaken, Switzerland, June 2001.
5. Ernesto Damiani, Barbara Oliboni, Elisa Quintarelli, and Letizia Tanca. Modeling users' navigation history. In *Proc. of Intl' Workshop on Intelligent Techniques for Web Personalisation (in conj. with IJCAI-01)*, Seattle, WA, August 2001.
6. The db2 product family. `www-3.ibm.com/software/data/db2/` .
7. Curtis E. Dyreson, Michael H. Böhlen, and Christian S. Jensen. Capturing and querying multiple aspects of semistructured data. In *Proc. of 25th Intl' Conf. on Very Large Data Bases (VLDB '99)*, Edinburgh, Scotland, September 1999.
8. F. Grandi and F. Mandreoli. The valid web: it's time to go... Technical Report TR-46, TimeCenter, `www.cs.auc.dk/research/DP/tdb/TimeCenter/`, 1999.

9. F. Grandi and F. Mandreoli. The valid web: an xml/xsl infrastructure for temporal management of web document. In *Proc. of the Intl' Conf. on Advances in Information Systems (ADVIS'2000)*, Izmir, Turkey, October 2000. LNCS 1909.
10. Christian S. Jensen, Curtis E. Dyreson, (Eds.), M. Böhlen, J. Clifford, Ramez A. Elmasri, Shashi K. Gadia, F. Grandi, P. Hayes, Sushil K. Jajodia, W. Käfer, N. Kline, N. Lorentzos, Y. Mitsoupoulos, A. Montanari, D. Nonen, E. Peressi, B. Pernici, John F. Roddick, Nandlal L. Sarda, Maria Rita Scalas, A. Segev, Richard T. Snodgrass, Michael D. Soo, Abdullah U. Tansel, P. Tiberio, and G. Wiederhold. The Consensus Glossary of Temporal Database Concepts - February 1998 Version. In Opher Etzion, Sushil Jajodia, and Suryanarayana Sripada, editors, *Temporal Databases — Research and Practice*, pages 367–405. Springer-Verlag, 1998. LNCS No. 1399.
11. Seung-Kyum Kim and Sharma Chakravarthy. Modeling time: Adequacy of three distinct time concepts for temporal data. In *Proc. of 12th International Conference on the Entity-Relationship Approach (ER'93)*, Arlington, TX, December 1993.
12. Leonid A. Kalinichenko Manuk G. Manukyan. Temporal xml. In *Proc. of 5th East European Conf. on Advances in Databases and Information Systems (ADBIS '01) – Vol. 1, Research Communications*, Vilnius, Lithuania, September 2001.
13. M.A. Nascimento and M.H. Eich. Decision time for temporal databases. In *Proceedings of the Third International Workshop on Temporal Representation and Reasoning (TIME'95)*, Melbourne Beach, FL, April 1995.
14. Norma in rete. http://www.normainrete.it.
15. The oracle 9i database. `otn.oracle.com/products/oracle9i/content.html`.
16. F. Ost and M. Van Hoecke (eds.). *Time and Law. Is the Nature of Law to Last.* Bruylant, Bruxelles, 1998.
17. M. Palmirani and R. Brighi. Norma-system: A legal document system for managing consolidated acts. In *Proc. of 13th Int'l Conf. on Database and Expert Systems Applications (DEXA)*, pages 310–320, Aix-en-Provence, France, September 2002.
18. The sql server database. `www.microsoft.com/sql/`.
19. The tamino xml server. `http://www.softwareag.com/tamino/`.
20. A. Tansel, J. Clifford, V. Gadia, S. Jajodia, A. Segev, R.T. Snodgrass, and (eds.). *Temporal Databases: Theory, Design and Implementation.* Benjamin/Cummings, Redwood City, CA, 1993.
21. W3C. Xquery 1.0: An xml query language. `http://www.w3c.org/TR/xquery/`, 2002.
22. F. Wang and C. Zaniolo. Preserving and querying histories of xml-published relational databases. In *Proc. of 2nd Intl' Workshop on Evolution and Change in Data Management (ECDM 2002)*, pages 26–38, Tampere, Finland, October 2002.
23. W3c xml schema. `www.w3.org/XML/Schema`.