Representing Time-dependent Information in Multidimensional XML[*]

Theodoros Mitakos, Manolis Gergatsoulis, Yannis Stavrakas and
Efstathios V. Ioannidis

Institute of Informatics & Telecommunications, National Centre for Scientific Research (N.C.S.R.) "Demokritos", Greece

Multidimensional XML (MXML) is an extension of XML that incorporates dimensions in order to represent in an elegant and concise way context-dependent data, that is, data which can exhibit different variations in value or structure (e.g. that can exist in different languages or various degrees of detail). In this paper we show that MXML dimensions can be used to express time and associate it with XML elements and attributes. We discuss the properties of time domain and the features of MXML which are necessary in order to represent time, and give key points that motivate further investigation on the use of MXML for representing time-dependent data.

Keywords: temporal semistructured data, multidimensional languages, XML

1. Introduction

XML [4] is emerging as a new standard for representation and data exchange over the Web [1]. However, it provides minimal support in cases where the same conceptual entity exhibits different facets under different contexts. The problem of representing and manipulating varying entities is especially important in the frame of WWW, where information providers cannot assume too much about the background context of the information consumers. Multidimensional XML (MXML) [8, 7] is a formalism that is suitable for representing context-dependent data in an elegant and concise way.

In this paper we argue that MXML can be used to represent time-dependent information. We show, through an illustrative example, how MXML elements and attributes can be qualified with the time period of their validity. We discuss how time domains can be expressed through MXML dimensions, and we take into account the natural ordering of time to introduce a compact representation of dimension values. We present key points that motivate further investigation of using MXML for representing time-dependent data.

Representing temporal XML documents has been also studied in [2], where leaf data nodes can have different variants, each qualified with the time period of its validity. The approach in [2] however, does not address explicitly the problem of representing variants of nodes other than leaves. Moreover, our approach is more general since we allow multiple time dimensions, as well as other user-defined dimensions.

2. Multidimensional XML

Multidimensional XML [7, 8] is an extension of XML in which elements and/or attributes can have various facets depending on the values of a set of dimensions. An element/attribute whose content depends on one or more dimensions is called *multidimensional element/attribute*. A *multidimensional XML document* is an XML document containing multidimensional elements and/or multidimensional attributes.

The syntax of XML is extended in order to incorporate dimensions. In particular, a multidimensional element has the form:

```
<@element_name attribute_specification>
   [context_specifier_1]
      <element_name attribute_specification_1>
         element_content_1
      </element_name>
   [/]
       . . .
   [context_specifier_N]
      <element_name attribute_specification_N>
         element_content_N
      </element_name>
   [/]
</@element_name>
```

where `element_content_i`, with $1 \leq i \leq N$ is the content of the (multidimensional) element for the context specified by [`context_specifier_i`]. The same element name is used for all *context elements* while the *proxy element* groups context elements and is denoted by preceding the element name with the special symbol "@".

A *context specifier* is of the form:

```
dimension_1_specifier, ..., dimension_m_specifier
```

where `dimension_i_specifier`, for `i` = 1 to `m` is a *dimension specifier* of the form:

```
dimension_name specifier_operator
     dimension_value_expres|sion
```

A *specifier_operator* is one of =, ! =, `in`, `not in`. If the *specifier_operator* is = or ! =, the *dimension_value_expression* consists of a single atomic value. Otherwise, if the *specifier_operator* is either `in` or `not in`, the *dimension_value_expression* is a set of values.

An attribute may also depend on dimensions. In this case, `attribute_specification` is of the form:

```
attribute_name = [context_specifier_1]
                    attribute_value_1 [/]
                        . . .
                  [context_specifier_n]
                    attribute_value_n [/]
```

Context specifiers of a multidimensional entity must be mutually exclusive. That is, for a single *world* - derived by assigning a specific value to each dimension - no more than one context element/attribute may hold in a multidimensional element/attribute. An important issue,

further discussed in section 4, concerns the inheritance of contexts throughout the paths of the document graph, as well as the reduction of an MXML document into a conventional XML document holding under a specific world.

3. Time and MXML

3.1. Properties of Time Domains

Time is an important aspect of all real-world phenomena. The real-world is dynamic and the facts and phenomena which happen in it occur over time. Various notions of time have been studied in the context of databases over the past two decades [6, 3] concerning the structural model and the properties of time. More specifically, time can be *linear* in the sense that it advances from the past to the future in a totally ordered fashion, or *branching* that flows from the past to the future in a tree-like way. Concerning time domains, time can be mapped to the set of integers $\mathcal{Z}$, to the rational numbers $\mathcal{Q}$, or to the real numbers $\mathcal{R}$, and it is said to be *discrete*, *dense* or *continuous* respectively.

Boundness is another important aspect of time. Time can be *bounded* or *unbounded*. We say that time is unbounded to the future or to the past if every time is succeeded by a later or earlier time respectively. Otherwise, if there exists a last or first time, we say that time is bounded in the future or in the past.

The notion of *clock* as a means for measuring time and the notion of *granularity* of time, which determines the distance between two consecutive time points when time is discrete, are often used in temporal systems.

Finally, various notions of time, such as *valid time*, *transaction time* and *user-defined time*, have been introduced in temporal databases. The *valid time* of a fact is the time when the fact is true in the modelled reality, while the *transaction time* of a fact is the time when a fact is stored in the database [5].

In MXML we consider linear and discrete time. Moreover, we assume time domains to be finite, i.e. time is bounded in both to the future and to the past.

3.2. Temporal Dimensions in MXML

MXML dimensions can be used to express time and associate it with elements and attributes of XML. In the following example we give a part of an MXML document that describes the (on-line) catalog of a restaurant. The complete catalog would contain a series of special comprehensive menus as well as independent choices for first dishes, main dishes and fruits, which are offered at specific dates, time periods, and days of the week.

The availability of the various catalog entries depends on the values of three temporal dimensions namely date, week_day and time. The dimension date ranges from 01/01/2000 to 31/12/2000, while the dimension time ranges from 00:00 to 23:59. Finally, the domain of the dimension week_day is the set of the names of the days of the week.

The multidimensional element menu of the example below consists of a single context element

Example 3.1. *A part of an (on-line) restaurant catalog.*

```
<@catalog>
  [week_day = Monday]
    <catalog>
      <comment> This day the restaurant is closed </comment>
    <\catalog>
  [/]
  [week_day in {Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday}]
    <catalog>
      <@menu>
        [time in {12:00..21:00}, week_day in {Saturday, Sunday},
              date in {01/01/2000..31/12/2000}]
          <menu>                              <!-- point 1 -->
            <@firstDish>
              [time in {12:00..18:00}]
                <firstDish>
                  <@salad>
                    [date in {01/05/2000..30/09/2000}]
                      <salad>  season salad </salad>          <!-- point 2 -->
                    [/]
                    [date in {01/01/2000..30/04/2000, 01/10/2000..31/12/2000}]
                      <salad>  c|hef salad </salad>
                    [/]
                  </@salad>
                </firstDish>
              [/]
              [time in {18:01..21:00}]
                <firstDish> <soup> c|hicken soup </soup> </firstDish>
              [/]
            </@firstDish>
            <maindish type =
                [date in {1/1/2000..30/4/2000, 1/9/2000..31/12/2000}]"frozen"[/]
                [date in {1/5/2000..31/8/2000}] "fresh" [/]>
              sword fish filet with vegetables
            </maindish>
            <@fruit>
              [date in {15/07/2000..31/08/2000}]
                <fruit> watermelon </fruit>
              [/]
              [date in {01/01/2000..30/04/2000}]
                <fruit> apple  </fruit>
              [/]
              [default] <fruit> fruit salad </fruit> [/]
            </@fruit>
          </menu>
        [/]
      </@menu>
      .... <!-- other 'menu' elements including salads, main dishes, fruits etc.>
    </catalog>
  [/]
</@catalog>
```

which holds only during weekends from 12:00 to 21:00 for the year 2000. It contains a first dish that, depending on the hour of the day, can be either salad or soup. The type of salad depends on the season of the year which is indicated by the value of dimension `date`. The menu also contains a main dish which is the same for all values of the dimensions for that menu. Finally, the menu contains a fruit which also depends on the dimension `date`. The value `default` for the dimension `date` means that fruit salad is available whenever no other fruit element holds for the dimension `date`. Temporal dimensions that appear in context specifiers may not only affect the structure or the value of elements, but they can determine the attribute values as well. For example, the attribute `type` in the element `main_dish` is a multidimensional attribute whose value is either `"frozen"` or `"fresh"` depending on the value of dimension `date`.

3.3. Representing Time Domains in MXML

In MXML each dimension takes value from a specific domain. Dimension specifiers determine the values of dimensions by assigning them subsets of the dimension domains. In example 3.1 we have three different dimensions namely `week_day`, `date` and `time`. All these dimensions express time. The domain of the dimension `week_day` is assumed to be the set of names of the days of the week. The domain of dimension `time` is the clock time. In the example we have adopted a representation of time that is of the form *hours:minutes*, assuming that the domain of time is discrete and includes every minute in an hour. The representation could also take into account seconds, and this poses the problem of different granularities while measuring time. Finally the domain of dimension `date` is the dates of year 2000 and its values are of the form *Day/Month/Year*.

It is worth noting that dimension `date` can be seen as a dimension representing absolute time, while the time in dimensions `week_day` and `time` can be seen as relative time.

Although the possible values of dimension `week_day` are easy to be explicitly given, this is not practical for the other two dimensions. For dimensions `date` and `time` a more compact notation is used. As it is shown in the example, the dots between starting and ending time

instances eliminate the need to write explicitly all the possible values of the dimension. This representation exploits the physical ordering of time and dates and it is a syntactic shorthand that allows for a compact representation of a set. The sets represented that way are considered to be totally ordered. Note that a dimension value can contain more that one such expressions separated by commas. In example 3.1, assuming a granularity of minute for `time` and a granularity of day for `date`, the number of possible worlds for which the context element `menu` at *point 1* holds is `2 x minutesIn(12:00, 21:00) x daysIn(01/01/2000, 31/12/2000)`, while the number of possible worlds for `salad` at *point 2* is `2 x minutesIn(12:00, 18:00) x daysIn(01/05/2000, 30/09/2000)`.

In this paper, time is assumed to be linear, discrete, and bounded (i.e. time domains are finite). However, it is easy to consider unbounded time by slightly extending the syntax. For example, we could write an expression of the form `30/6/2000..INFTY` as a way to represent the infinite set of dates that follow the date `30/6/2000`.

4. Propagation of Context

In MXML every element and attribute, be it conventional or multidimensional, is associated with a context. The context of an element/attribute defines the worlds for which that element/attribute holds, by assigning values to dimensions.

In general, subelements inherit the context of their parent elements. The principle behind context propagating in this way is that an (sub)element should in general hold for all worlds included in the context of its parent element (i.e. for the worlds under which its parent element holds), unless the subelement further specializes the context it inherits through its context specifier.

In example 3.1 there are various cases that illustrate context inheritance. The context element `menu` at *point 1* becomes the holding facet on Saturdays and Sundays between 12:00 and 21:00 provided the date is within the year 2000. The subelement `firstDish` consists of two context elements that inherit the context of `menu` and

further specialize it by narrowing down the values of dimension `time`. One of the `firstDish` context elements contains the multidimensional element `salad`, which in turn comprises two facets that depend on `date` — we assume that the salad suggestion changes according to the availability of ingredients throughout the year.

By associating element and attribute variants with a context, it is possible, given a world w, to reduce an MXML document G to an XML document G', such that G' is the holding facet of G in the world w. For example, in the case of a single time dimension this reduction can give the XML instance of G holding at a time point t.

5. Conclusions and Future Work

In this paper, we argued that MXML dimensions can be used to represent temporal information. We examined the properties and discussed semantic issues of time domain. We illustrated through an example how time-varying data can be represented in a robust and concise way.

MXML seems very promising for representing information that depends on temporal dimensions, however, a lot of issues remain open: (a) DTDs for Multidimensional XML should support the definition of dimensions whose domain is discrete, ordered and infinite. Support for defining different granularites for the time domains is also needed. (b) Research work should also be carried out towards defining a query language for MXML that will allow operations over time domains.

References

[1] S. ABITEBOUL, P. BUNEMAN, AND D. SUCIU, *Data on the Web: From Relations to Semistructured Data and XML*, Morgan Kaufmann Publishers, 2000.

[2] T. AMAGASA, M. YOSHIKAWA, AND S. UEMURA, A Data Model for Temporal XML Documents, in M. T. Ibrahim, J. Kung, and N. Revell, editors, *Database and Expert Systems Applications, 11th International Conference, DEXA 2000, London, UK, Sept. 4–8, Proceedings*, Lecture Notes in Computer Science (LNCS) 1873, pages 334–344. Springer-Verlag, Sept. 2000.

[3] M. BAUDINET, J. CHOMICKI, AND P. WOLPER, Temporal deductive databases, In L. Farinas del Cerro and M. Penttonen, editors, *Temporal Databases: Theory, Design, and Implementation*, pages 294–320. The Benjamin/Cummings Publishing Company, Inc, 1993.

[4] T. BRAY, J. PAOLI, AND C. M. SPERBERG-MCQUEEN, Extensible markup language (XML) 1.0 (second edition), http://www.w3.org/TR/2000/REC-xml-20001006, October 2000.

[5] C. S. JENSEN, J. CLIFFORD, S. K. GADIA, A. SEGEV, AND R. T. SNODGRASS, A Glossary of Temporal Database Concepts, *SIGMOD RECORD*, 21(3):35–43, September 1992.

[6] G. OZSOYOGLU AND R. T. SNODGRASS, Temporal and real-time databases: A survey, *IEEE Transactions on Knowledge and Data Engineering*, 7(4):513–532, August 1995.

[7] Y. STAVRAKAS, M. GERGATSOULIS, AND T. MITAKOS, Representing context-dependent information using Multidimensional XML, In J. Borbinha and T. Baker, editors, *Research and Advanced Technology for Digital Libraries, 4th European Conference ECDL'2000*, Lecture Notes in Computer Science (LNCS) 1923, pages 368–371. Springer-Verlag, 2000.

[8] Y. STAVRAKAS, M. GERGATSOULIS, AND P. RONDO-GIANNIS, Multidimensional XML, In P. Kropf, G. Babin, J. PLaice, and H. Unger, editors, *Distributed Communities on the Web, Third International Workshop (DCW'2000)*, Lecture Notes in Computer Science (LNCS) 1830, pages 100–109, Springer-Verlag, 2000.

Contact address:
Theodoros Mitakos, Manolis Gergatsoulis,
Yannis Stavrakas, Efstathios V. Ioannidis
Institute of Informatics & Telecommunications
National Centre for Scientific Research (N.C.S.R.) "Demokritos"
153 10, Aghia Paraskevi Attikis, Greece
e-mail: {teo}@softlab.ntua.gr
e-mail: {manolis,ystavr,stathis}@iit.demokritos.gr

DR. THEODOROS MITAKOS received his Diploma in Computer Engineering in 1992 from the University of Patras. He received his Ph.D. degree in Computer Science in 1998 from the National Technical University of Athens (NTUA), Greece. Since 1998 he is a researcher at NTUA and since 1999 a collaborating researcher at the National Centre for Scientific Research (NCSR) "Demokritos", Athens. Since March 2001 he is a visiting assistant professor at Technological Educational Institute of Halkida, Evia. He is also the Technical Director of InfoWise Co, Halkida Greece. His research interests include logic and modularity, data mining, deductive and object-oriented databases, XML and semistructured data.

DR. MANOLIS GERGATSOULIS received his Diploma in Physics in 1983, his M.Sc. and Ph.D. degrees in Computer Science in 1986 and 1995 respectively, all from the University of Athens, Greece. Since 1996 he is a Research Associate in the Institute of Informatics and Telecommunications, of the National Centre for Scientific Research (NCSR) "Demokritos", Athens. Since September 1999, he is a visiting assistant professor at the Department of Technology Education, University of Piraeus. His research interests include logic and temporal programming, program transformations and synthesis, deductive databases, XML and semistructured data.

YANNIS STAVRAKAS received his Degree in Physics in 1989 from the University of Athens, Greece. In 1992 he received the M.Sc. with Distinction in Computer Science from the University College London. He worked as a software engineer for a period of five years, and was involved in a number of research and commercial projects. He is a member of the Knowledge and Database Systems Laboratory at the National Technical University of Athens, where he pursues a Ph.D. Since 1998 he is an associate researcher at the Institute of Informatics and Telecommunications of the National Center for Scientific Research "Demokritos". His research interests include semistructured data, XML, representation of context-dependent information, and collaborative filtering.

EFSTATHIOS V. IOANNIDIS received his BA degree in Management & Administration from the University of Piraeus in 1967. During 1968–1971 he was on a scholarship at the Computer Department of the National Center for Scientific Research (NCSR) "Demokritos", Athens, Greece. From 1971 to 1990 he participated, as analyst-programmer and sub-manager of EDP department team of NCSR "Demokritos", in the design, development and maintenance of more than twenty information systems (on Bibliographies, Economics, Mathematics etc.) From June 1990, as a member of the "Software & Knowledge Engineering Lab" (SKEL) of the Institute Informatics & Telecommunication (IIT) of NCSR "Demokritos", he has the responsibility as manager for the Local Network and as DBA for all RDBMS (INGRES, SYBASE, ORACLE etc.). His current research interests are in the areas of temporal representation & reasoning, temporal and/or deductive database, models & methodologies for software engineering and new informatics technologies (information extraction, data mining etc.)