Structural Organization of Anatomical Data Using XML Technologies

Goran S. Krstic[1], and Zoran Z. Stanković[2]

Abstract - **This paper describes the basics necessary for building an educational system in the field of anatomy using XML technologies. It contains the original XML schema for hierarchical structuring of anatomical data. It also contains a discussion on the possibilities of implementation.**

Keywords - **XML, XML Schema, XPath, XSL, SVG, Anatomical Data.**

I. Introduction

XML (*Extensible Markup Language*) is able to represent any kind of structured and semi-structured data. As a meta-signifying language, it has enabled the development of many other new signifying languages (vocabularies). These vocabularies are defined by XML schemas (*XML Schema Language*). XML schemas are used to specify the structure of XML documents and requests that need to be answered. The wide range of XML's possibilities mostly stem from the strict rules under which the documents can be created. Languages based on XML are independent from the platform, simple for lexical analysis, with well-defined syntax, easily read by people, and there are a lot of tools for lexical analysis and manipulation of XML data.

Furthermore, xml belongs to a family of technologies which rapidly develop. Today, XML technologies have a wide range of appliance, from document development, through developing multi-layer (*n-tier*) applications and Web services, to semantic Web, in all areas of human functioning (business, medicine, education, government organizations etc.)

This paper gives the basics needed for the realization of educational system in the field of anatomy. XML is considered to be a mature technology, which helps realize an advanced educational system. At the end of the paper, the analyzed possibilities and ways of implementation are summed up. We have also pointed out some conceptual advantages that the displayed solution is expected to have in relation to the existing programs in this area, which primarily rely on the multimedia aspect.

[1]Goran Krstic is with the College of Agriculture and Food Technology, Cirila and Metodija 1, Prokuplje, Serbia, E-mail: gorca@ni.ac.rs.

[2] Zoran Stankovic is with the Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Nis, Serbia, E-mail: zoran.stankovic@elfak.ni.ac.rs.

II. XML Schema

All living systems are hierarchically organized [1]. The human organism is composed of organ systems, nine of them altogether. Organic systems, as indicated by the term, are composed of several organs, which are composed of various tissues built of one or more types of cells. The human organism, anatomically seen, may represent the best example of hierarchical organization and order in nature (Figure 1). Therefore, it is the best to represent anatomical data by a hierarchical structure provided by XML.

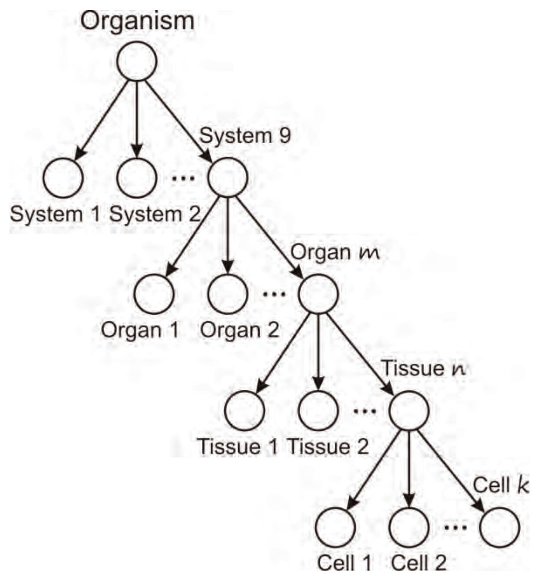

Fig.1. *Hierarchical organization of human organism.*

The numbers *m*, *n*, *k* relate to the number of types of entities, instead of their total number.

XML schema language is able to model any hierarchy that is generally plausible to specify [2, 3, 4]. The XML *Anatomy* schema follows the hierarchical structure of an organism:

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <!-- XML Schema Anatomy.xsd -->
    <xsd:element name="Organism">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Latin-name"/>
                <xsd:element ref="English-name"/>
                <xsd:element ref="Serbian-name"/>
                <xsd:element ref="Description"/>
                <xsd:element ref="System" maxOccurs="9"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="System">
```

```
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Latin-name"/>
                <xsd:element ref="English-name"/>
                <xsd:element ref="Serbian-name"/>
                <xsd:element ref="Description"/>
                <xsd:element ref="Organ"
                 maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
. . .
    <xsd:element name="Cell">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Latin-name"/>
                <xsd:element ref="English-name"/>
                <xsd:element ref="Serbian-name"/>
                <xsd:element ref="Description"/>
                <xsd:element ref="Part" minOccurs="0"
                 maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="Part">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Latin-name"/>
                <xsd:element ref="English-name"/>
                <xsd:element ref="Serbian-name"/>
                <xsd:element ref="Description"/>
                <xsd:element ref="Sub-part" minOccurs="0"
                 maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
. . .
    <xsd:element name="Latin-name" type="xsd:string"/>
    <xsd:element name="English-name" type="xsd:string"/>
    <xsd:element name="Serbian-name" type="xsd:string"/>
    <xsd:element name="Description" type="xsd:string"/>
</xsd:schema>
```

Elements of the schema correspond to the entities composing the organism. The *Description* element is used as the container element and is supposed to embody the main part of the educational content. Moreover, the *Description* element should include links to other elements in the document, which provides a mixed structured of this element and an XML oriented towards (*document-centric*, *document-oriented)*. This brings about new issues in the implementation phase. For example, relational data bases can only accept a data-oriented XML. More about it at the end of work.

The role of the *Part* element is dual. On the one hand, it includes parts of entities which comprise the hierarchical structure, and on the other hand it includes all the macroscopically visible elements that are not directly incorporated in the hierarchical structure (upper part, lower part, front, back).

For reasons of clarity and visibility, in the first illustration, the schema is not burdened with details. In the implementation phase, it is possible to add various restrictions, which provides better control and validity of the document. For example, it can be ensured that an element is not left empty during the insertion of data into the XML document, i.e. – that it contains at least one character.

```
<xsd:element name="Latin-name">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:minLength value="1"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
```

Needless to say, the schema is not final; it is open to development and upgrading before implementation, as well as in the implementation phase itself.

III CREATING A DOKUMENT

In XML terminology, there are two terms related to the accuracy of the document, i.e. the XML document can be well-formed and valid. A well-formed document fulfills the basic XML syntax. A valid document is a well-formed document that also responds to the structure defined by the XML schema.

All documents must be well-formed, but not necessarily valid. However, a valid document has certain advantages. Checking the validity ensures that the document responds to a certain structure and satisfies the standards demanded. If a part of the document does not respond to the specification in the schema, the processor shows an error notification, and therefore the necessary corrections can be made. The following document is a valid one that fulfills the standards specified by XML *Anatomy* schema:

```
<?xml version="1.0"?>
<?xml:stylesheet type="text/xsl" href="Anatomy.xsl"?>
<!-- Dokument Anatomy.xml -->
<Organism xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Anatomy.xsd">
    <Latin-name>Human</Latin-name>
    <English-name>Man</English-name>
    <Serbian-name>Čovek</Serbian-name>
    <Description>The most perfect organism on earth
...</Description>
    <System>
        <Latin-name>Sceletus</Latin-name>
        <English-name>Skeletal system</English-name>
        <Serbian-name>Koštani sistem</Serbian-name>
        <Description>Every living thing has its own form and
          structure. The skeletal system is ...</Description>
        <Organ>
            <Latin-name>Femur</Latin-name>
            <English-name>Femur</English-name>
            <Serbian-name>Butna kost</Serbian-name>
            <Description>Femur is the largest bone in the human
            body. Belongs to ...</Description>
            <Part>
                <Latin-name>Trochander major</Latin-name>
                <English-name>Big wheel</English-name>
                <Serbian-name>Veliki kotur</Serbian-name>
                <Description>Forms part of ...</Description>
            </Part>
            <Part>
                <Latin-name>Trochander minor</Latin-name>
                <English-name>Smal wheel</English-name>
                <Serbian-name>Mali kotur</Serbian-name>
                <Description>Forms part of ...</Description>
            </Part>
            <Tissue>
                <Latin-name>Bone tissue</Latin-name>
                <English-name>Bone tissue</English-name>
                <Serbian-name>Koštano tkivo</Serbian-name>
```

```xml
                <Description> Falls into the category of connective
                 tissue... </Description>
                <Cell>
                    <Latin-name>Osteocit</Latin-name>
                    <English-name>Osteocyte</English-name>
                    <Serbian-name>Osteocit</Serbian-name>
                     <Description>Osnovna ćelija koštanog tkiva...
                      </Description>
                </Cell>
                <Cell>
                    <Latin-name>Osteoblast</Latin-name>
                    <English-name>Osteoblast</English-name>
                    <Serbian-name>Osteoblast</Serbian-name>
                    <Description>Mlade ćelije koje ...</Description>
                </Cell>
            </Tissue>
        </Organ>
    </System>
</Organism>
```

XML is not only a signifying language; it also represents an almost ideal way of structuring data. The given document is barely larger that the minimal one that satisfies the rules specified in the schema. In the realization phase, a far larger XML document is given, perhaps even the largest one so far.

IV. Advantages of using XML technologies

The existing educational programs in this area have relatively good possibilities of visualization and animation. The disadvantage is bad organization of data, which does not follow the hierarchical structure of organism. Their crucial disadvantage is the lack of browsing possibilities by using a questioning language. The data is organized into lessons accessed from a formerly established menu.

Once we have structured data in XML format, huge possibilities are achieved. Here is an attempt to systematize them and point out each of the advantages of using XML technologies:

Data storage. Even though an XML document can basically be placed in a relational or object-relational database, the Native XML Database should be primarily considered.

The source XML databases are also appropriate for *data-centric* and *document-centric* applications. They have an XML document for a fundamental unit of logical data storage, just like the relational database has syllable tables as its fundamental unit of logical storage. This group of databases includes Tamino, Ipedo, dbXML, etc. With these databases, there is no conversion, the data remain in XML format, which in this case makes them the best choice for storage of XML coded anatomical data.

Searching. The languages currently existing used in commercial products are SQL and Xpath (*XML Path Language*). It is evident that relational products will be browsed using SQL, while XML data will be browsed with XPath. In relation to this, their possibilities are to be considered.

If requests are to be made over the recursive data structure, XPath is more suitable than the “syllabic” RDBMS and SQL. For example, if we need to find out which parts (ingredients), sub-parts, or even sub-sub parts a certain element contains, languages based on XPath will handle this kind of data better than SQL-based languages. On the other hand, SQL has a far richer collection of operators for manipulating data than XPath languages. In the particular case discussed here, there are no mathematical operations with the elements, so XPath tends to be the language that is more suitable for data browsing.

Supposing we are not using XML and we want to place the data in a relational database. A relational representation could be achieved by *master-detail* organization of tables that would follow the hierarchical structure of organism in Illustration 1. However, on the lowest level, there would be a plethora of tables ($m*n*k$), which would result in too many contactenations, making the requests too long and practically unworkable. For example, if we wanted to select all types of cells in the organism, we would generally have the following situation:

$table_1.key = table_2.key$ and $table_2.key = table_3.key$ and ... and $table_{n-1}.key=table_n.key$

Xpath expression is far simpler:

```
//Cell
```

Selects all the elements in the document named *Cell*.

Data display. Not presuming the final solution, it is clear that the most natural way of displaying XML data is by using XSL stylesheets (*Extensible Stylesheets language*). By using Xlink (*XML Linking Language*) for referring and SVG (*Scalable Vector Graphics*) picture format, the use of XML technologies would be fully encompassed. SVG has the advantage over other formats, because it uses XML syntax for inscription of geometrical shapes. This provides various possibilities for working with images, such as interactivity, animation, reaction to different events, etc. [5]. Apart from that, it the data are placed in an XML database, the solutions these bases offer for storage and working with images are also available.

Application. Using the potentials of a database in which XML data are placed, it is also easy to construct a Web application, which would provide an *on-line* anatomy and enable e-learning. The vision goes so far as to also enable a *stand-alone* application that would provide a three-dimensional display of elements, their rotation and interaction. We agree that this is not so easy to achieve at the moment, but this possibility should be pursued.

Level of detailisation. It is important to notice that XML data are separated from display. The same XML document can be displayed in several ways, without making any changes in the document itself. In relation to that, it is possible to determine the level of detail of the display, where one level or degree of details would be used for students, and another one for specialists, for example - i.e., it would show even the smallest details. It is possible to further separate the sub-documents and get histology (the branch of anatomy dealing with tissues), or cytology (the branch of anatomy dealing only with cells)

Semantic upgrading. It is possible to use one of the languages of semantic Web (RDF and RDFS) to build classes that would correspond the hierarchical structures of the organism (metaphorically seen, of course, since an organ is not precisely a sub-class of the organism. XML data can be used to fill instances, which would form an information base. If RDF(S) were to be used, it would be possible, using GSS (*Graph Stylesheets Language*), to get a certain diagram display of the organism. It is also possible to further improve the system by developing ontology. By the implementation of knowledge, certain aspects of organism functioning, i.e. physiology, could be enveloped.

Multilinguality. It is possible to efficiently organize the educational content and implement multilinguality in an entirely uniform way.

The best results would be achieved by using: an XML database for storing XML documents, XPath question language for browsing, XSL stylelist to display data and SVG format for the images.

V. CONCLUSION

This paper lays out the original XML schema for hierarchical structuring of anatomical data. The basics necessary for the realization of an advanced educational system in the field of anatomy by using the advantages of XML technology are laid out. It is our opinion that the recommended solution will enable efficient structuring of content and implementation of multilinguality.

REFERENCES

[1] G. Tortora, Principles of Human Anatomy, 6th ed., HarperCollins Publishers, New York, 1992.

[2] A. Skonnard, “Designing XML Schema Libraries”, http://msdn.microsoft.com/library/

[3] XML Schema Part 1: Structures, W3C Recommendation 2 May 2001, http://www.w3.org/TR/xmlschema-1/

[4] XML Schema Part 2: Datatypes, W3C Recommendation 02 May 2001, http://www.w3.org/TR/xmlschema-2/

[5] Scalable Vector Graphics (SVG) 1.1 Specification, W3C Recommendation *14 January 2003* http://www.w3.org/TR/SVG11/