

XML Model for Legal Documents

Vanyo Peychev

Faculty of Mathematics and Informatics, Sofia University, 1164 Sofia

E-mail: vanyo@fmi.uni-sofia.bg

Introduction

Information is everywhere around us. We are diving in an informative ocean. The structured presentation of information improves its adoption and gives us possibilities to define operation above the document. Defining a structure for information is a process. This process introduces metadata which improves both visualization and operations over data. The text information is distributed and processed as document form.

Every document is a set of paragraphs. Document in which the paragraphs have a name and predefined order we call *structured document*. The paragraph naming and predefined ordering process we call *document tagging*. Document tagging means the diving of document into a set of sections. Each section begins with tag, has a text and ends with an end tag. It is possible the document to have more than one repeating text part.

Structured documents can be processed in different ways. Document processing and management require the documents to be saved in specific data structures. Some methods for structured documents saving and management are:

- formatting languages as SGML, XML;
- database for saving document structure and data;
- hypertext management systems.

SGML [5] is used as universal means for data formatting for many different purposes. Usually it is applicable in huge document management systems [2]. When we talk about huge complex data, we should define a lot of additional requirements to the systems that describe the data. As a result a SGML description of such systems becomes very complicated.

XML language is a subset of SGML created for the same purposes – formatting unspecified data types. Unlike SGML that contains a great number of complicated constructions, XML is simplified as much as possible [3]. The ability for describing unspecified information is XML power. XML is extremely flexible in regard to structure

the data. The fast developing of XML accelerates the usage of formatting languages for documents description and management.

The database solution for saving and management of structured documents is different approach giving some advantages. Concerning saving and management of documents, the major advantage of database is the possibility to use operations from relational algebra and relational calculus for describing document operations.

Hypertext systems for document management consider the relationships between documents. In practical, this class of systems saves both document structure and hyperlinks to other documents. The hypertext systems aim visualization of linked information.

The article investigates legal documents. The aim is to model legal documents by XML. For creating a valid XML document, Document Type Definition (DTD) is introduced. There are many reasons to model legal documents as a document type [7], but the more important ones are:

- the predefined and stable structure;
- the presentation of huge text information;
- the unified presentation of a class of legal documents improves the execution of operations over them;
- the usage of well-known methods for managing and processing.

The article has two parts. First one presents XML model of legal document. The second one describes XQuery and how we can query legal documents.

Legal document structure

The building of legal documents information management systems is a huge and complex problem. It is a problem of organization structure and process flow modeling for the enterprise. It is a legal knowledge presentation problem too. It is a problem of converting and saving of legal information in electronical form. Every legal document consists of elements with hierarchical structure [4], as it is presented in Fig. 1.

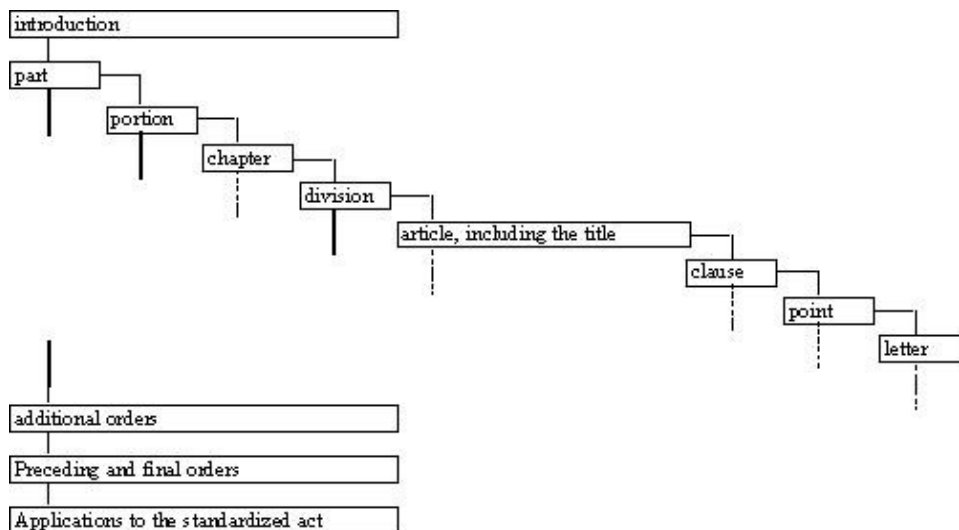


Fig. 1

Legal documents organization structure is built according specific rules. The rules define relations between document elements. The relations are introduced by links. *The links* are specific language constructions which describe relationship between legal texts. The links are special extend to syntax of document. Their existance is required by the need of clear definitions for relationships between different components of legal documents (typically the components are clauses or terms). The links effect into concentration of legal text. This way every legal norm can be formulated using the texts of already defined legal norms.

Links information is used for creation of a legal text as well for revelling relationships between legal norms for establish the legal structure. The links, as legal term, are equivalent to hyperlinks, as informatics term. The term hyperlink contains the meaning of the term link. Hyperlink adds more meaning to the links eventually to meet particular information necessities.

Other special feature of legal documents is the dynamic of data. This dynamic reflects the perpetual development of legal documents.

Legal documents contain a data which can be changed periodically. Legal document life cycle passes through the following steps:

- creating a project for legal document;
- accepting;
- add new texts and change existing ones;
- abrogate parts from document or whole document.

Every life cycle phase effect document structure and content. The changes affect either whole document or some of its parts as it is shown on Fig. 1 development of legal documents defines the next data groups:

- projects for legal texts;
- legal texts which are accepted and appear soon;
- currently working legal texts;
- legal texts have been abrogated.

Information meaning about first three data groups is indisputable. The data for abrogate legal texts is necessary not only for legal-historical analysis but it is important to solve legal problems based on abrogate texts.

XML presentation of legal document

For saving and management of legal documents is used either formatting languages or database. The chosen method for manipulating legal documents is using formatting languages, particular XML. Legal document structure is described with XML, which is good solution for data transfer from/to database. Based on legal document structure, as it is described on Fig. 1, XML with DTD is created.

Document Type Definition (DTD) [1] is standardized method for describing XML node types. DTD describes the document structure, document nodes, element data types and default values. DTD allow group of XML documents to share one and the same XML dictionary. For describing legal document structure the following DTD has been created:

```
<!ELEMENT Documents (Law?) >
<!ELEMENT Law (LawName, Introduction, LawParts) >
<!ELEMENT LawName (#PCDATA) >
<!ELEMENT Introduction (#PCDATA) >
```

```

<!ELEMENT LawParts (Part) >
<!ELEMENT Part (partName, Stakes) >
<!ELEMENT partName (#PCDATA) >
<!ELEMENT Stakes (stake)>
<!ELEMENT stake (stakeName, Heads)>
<!ELEMENT stakeName (#PCDATA) >
<!ELEMENT Heads (Head?)>
<!ELEMENT Head (headName, Sections)>
<!ELEMENT headName (#PCDATA) >
<!ELEMENT Sections (Section)>
<!ELEMENT Section (sectionName, Clauses)>
<!ELEMENT sectionName (#PCDATA) >
<!ELEMENT Clauses (Clause)>
<!ELEMENT Clause (clauseName, clauseText, Terms)>
<!ELEMENT clauseName (#PCDATA) >
<!ELEMENT clauseText (#PCDATA) >
<!ELEMENT Terms (Term)>
<!ELEMENT Term (termName, termText, Points)>
<!ELEMENT termName (#PCDATA) >
<!ELEMENT termText (#PCDATA) >
<!ELEMENT Points (Point)>
<!ELEMENT Point (pointNumber, pointName, Leters)>
<!ELEMENT pointNumber (#PCDATA) >
<!ELEMENT pointName (#PCDATA) >
<!ELEMENT Leters (Leter)>
<!ELEMENT Leter (leterNumber, leterName)>
<!ELEMENT leterNumber (#PCDATA) >
<!ELEMENT leterName (#PCDATA) >

```

With this DTD we can create XML documents for different legal documents. DTD guarantees validation rules for XML. These rules include structure and text control according legal document structure, which is described on Fig. 1. The model considers that the links are internal data parts, i. e. links belong to section # PCDATA.

This XML model for legal documents can be used as input/output format for data transportation between different databases. Saving temporal structure is realized with saving different instances of the legal document. For searching in XML based data we prefer XML querying languages.

Searching in legal document through XQuery

XQuery [8] is XML based language. It allows querying XML documents. Example of XQuery expression for searching in XML document is the following:

```

for $point in doc("law.xml")//Clauses
where $point/clause/ClauseName = X
and $point/clause/Terms/termName = Y
and $point/clause/Terms/Points/pointNumber = Z
return $point/clause/Terms/Points/pointName

```

The query assigns the variable \$point to every section Clauses and set of tuples is created. Every tuple assigns the variable \$point to single Clauses node. After that the defined filter is applying. As result of execution this query text for point Z is extracted. The point with number Z is subpoint to term with number Y which is a subterm for article with number X, as it is shown on Listing 1.

Listing 1 Result query for point Z

```

<clause>
  <clauseName>X</clauseName>
  <clauseText>Some clause text</clauseText>
  <Terms>
    <term>
      <termName>Y</termName>
      <termText>Some term text</termText>
      <Points>
        <point>
          <pointNumber>Z</pointNumber>
          <pointName>Some point text</pointName>
          <Leters>
            <leter>
              <leterNumber>Z1</leterNumber>
              <leterName>Some leter text if exists</leterName>
            </leter>
          </Leters>
        </point>
      </Points>
    </term>
  </Terms>
</clause>

```

Using XQuery we have a freedom to prepare result data in appropriate type for future processing. Creating well formatted results with XQuery is useful for presenting result data in appropriate view. Usually XQuery return a document node, if no additional formatting is assigned. This feature allows creatinon of different and user like results. For example to visualize the text of point Z from legal document through WAP [9] essay way is generating WML [6] result.

```

for $point in doc("law.xml")//Clauses
where $point/clause/ClauseName = X
  and $point/clause/Terms/termName = Y
  and $point/clause/Terms/Points/pointNumber = Z
return
<wml>
  <card id="Text_1" title="SomeTextElement">
    <p>
      Член No: { $point/clause/ClauseName } <br/>
      Алинея No: { $point/clause/Terms/termName } <br/>
      Точка No: { $point/clause/Terms/Points/pointNumber }
    </p>
  </card>

```

```
<card id="Text_2" title="SomeTextElement">
  <p> { $point/clause/Terms/Points/pointName } </p>
</card>
</wml>
```

Conclusion

The entire economic life needs information systems for legal documents. The Bulgarian information systems for management legal documents are limited. The reason is that all of them are using old technologies for development. The future for legal documents growth can be generalized in the conclusion: automatization of legal information will bring legal information closer to the users. The article is a step to this direction.

The article explains XML model for legal documents. The model is useful for querying legal documents. XQuery (Used language for query XML data) allows creation of user friendly formatting of the result data. These advantages make the solution wide-applicable and bring it closer to the user requirements.

References

1. Finkelstein, C. Building corporate portals with XML. McGraw-Hill, 2000.
2. Herwijnen, E. Practice SGML. International Thomson Publishing, 1995.
3. Hunter, D. Professional XML. Wrox, 2001.
4. Peychev, V, V. Dimitrov. Dynamic WAP application for information search in legal documents. – In: Proceedings of 31th spring conference of the Union of Bulgarian Mathematicians, Borovets, 2002.
5. Rubinsky, Y, M. Maloney. SGML on the Web. Prentice Hall PTR, 1997.
6. Rytönen, K. Mobile Commerce and WML. 2000.
7. Office of Justice programs, Structure and Design Issues for Developing, Implementing and Maintaining a Justice XML Data Dictionary. U.S. Department of Justice, 2002.
8. XQuery 1.0: An XML Query Language, W3C Working Draft, 2 May 2003.
<http://www.w3.org/TR/xquery>
9. [WapfINet] <http://www.wapforum.org>

XML модель нормативного акта (закона)

Ваньо Пейчев

*Факультет математики и информатики, Университет Софии, 1164 София
E-mail: vanyo@fmi.uni-sofia.bg*

(Резюме)

В современной информатике все более внимание обращается обработке текстовых данных. Тенденция состоит в том, чтобы представление и обмен данных, между приложениями тоже, реализовались с информацией, основанной на тексте. С другой стороны, текстовые документы дают возможность большей свободы при представлении информации. Цель этой работы представить XML модель нормативного акта (закона) и возможность выполнения заявок над законом при помощи языка XQuery.