Novi Sad J. Math.
Vol. 41, No. 2, 2011, 131-148

XML IN THE BISIS LIBRARY MANAGEMENT SYSTEM[1]

Danijela Tešendić[2], Danijela Boberić Krstićev[2], Bojana Dimić Surla[2]

Abstract

The paper describes an application of the XML technologies in the development of the BISIS library system. XML can be used in different applications in the development of library systems, for example, for description, storage and exchange of bibliographic data. In the BISIS system, XML is used for description of the bibliographic formats, as well as for description of bibliographic record structure. Besides, XML is used for description of the queries used for searching and downloading records from distant repositories. In order to enable circulation processes in a consortium, XML is used for data exchange related to the library users.

AMS Mathematics Subject Classification (2010): 68N01

Key words and phrases: XML, BISIS, bibliographic records, NCIP, Z39.50

1 Introduction

Since the appearance of XML (eXtensible Markup Language), there has been a widespread interest in its application in the development of library information systems. There are numerous projects on the application of XML technology in library information systems and the number of libraries that store bibliographic records in XML is increasing. So it is not surprising that the mailing list on this topic appeared [32]. The subject of more important research on the application of XML in libraries is creating different types of XML documents using XML technologies such as Document Type Definition (DTD) and XMLSchema. Another subject relates to all types of XML transformations by using XSLT technology. The DTD and XMLSchema are languages for describing structure and content of XML documents. XSLT (Extensible Stylesheet Language Transformations) is an XML-based language by which an XML document is transformed into the another XML document. The XML applications have different purposes in libraries, such as, representation of the bibliographic records, records transfer and validation as well, as transfer of other data used in the library management systems. The importance of the XML technologies for the library system functioning is stressed in [11] and the application of the XML

[1]Authors would like to acknowledge the support of the Serbian Ministry of Science, project OI–174023 – Intelligent techniques and their integration into wide-spectrum decision support and project III–47003 – Infrastructure for technology enhanced learning in Serbia.

[2]University of Novi Sad, Faculty of Sciences, Trg D. Obradovića 4, 21000 Novi Sad, Serbia

within the bibliographic standards such as MARC, Z39.50, ISO interlibrary loan (ILL) and NISO circulation interchange protocol (NCIP) is described.

This paper gives description of the application of XML technology in the implementation of the BISIS library software system. The system has been developing at the Novi Sad University since 1993. It has a number of versions and the actual one is the fourth one. The BISIS system has been implemented in the *Java* programming environment. The fourth version of the BISIS system contains the XML technology elements in all parts of the library system. The part related to the development of cataloguing editor is described in [7–9, 19]. The development of the editor is based on the XML technologies. The part of the system related to the use of the library material, that is circulation, is described in [21,27–29]. In [1,2,20], an editor for query creating under the Z39.50 protocol is described. In [25], a solution for the creation of cataloguing cards in the form of an independent software package is proposed. A main subject in the paper [26] is the application of the XML technologies for bibliographic records conversion into the MARC 21 format.

Within the BISIS project for the needs of Novi Sad University, as well as on the recommendations of the euroCRIS nonprofit organization, the CRIS (Current Research Information System), an information system of the scientific and research work [13, 22] is developed [10]. In [14] is given the CERIF proposition [6] of the compatible data model based on the MARC 21 format. In this data model, a part of the CERIF data model related to the research results is mapped on the MARC 21 format data model. In the paper [15], an extension of the CERIF data model for the evaluation of published scientific results of the researchers is given. The model is verified against the Rule book of evaluating and quantitative stating of scientific results of the researchers from the University of Novi Sad.

UNIMARC and MARC 21 formats are the most known bibliographic formats. In order to develop a bibliographic system based on the XML technologies, it is necessary to create the XML schemas for description of the formats, as well as corresponding XML schemas for bibliographic records. Some of these schemas are described in Section 2.

The XML technology application is present in the development of the software systems for downloading of bibliographic records that provide, first of all the interoperability to the other systems. The interoperability is attained by the Z39.50 and SRU standards. The XML schemas of the query languages defined by the Z39.50 standard are described in Section 3.

The NCIP (NISO Circulation Interchange Protocol) or the Z39.83 is the protocol for exchanging data about users. The syntax and semantics of the messages exchanged by the applications are defined by this protocol in order that functions of circulation of library holdings and access control to electronic resources could be enabled. The protocol syntax is described by the XML schema and messages are changed into an XML format. The implementation of the protocol within the BISIS system is presented in Section 4.

2 XML in bibliographic data processing

Library information systems are based mainly on bibliographic formats of which the UNIMARC and MARC 21 are the most known. For development of the systems based on the XML technologies it is necessary to form the XML schemas for the formats and the XML schemas for bibliographic records within these formats.

The XML schema that describes both MARC 21 and UNIMARC formats, as well as the object model arrising on the basis of the schema, is described in Section 2.1. An instance of the schema is used in the editor for cataloguing of the BISIS system. Section 2.2 describes the XML schemas for bibliographic records. Based on the records, an object model of bibliographic record that is used in the BISIS system is created.

2.1 XML schema of bibliographic formats

Schemas for the UNIMARC and MARC 21 format are developed within the BookMARC project [3]. A research result in this project that is related to the UNIMARC format is given in the paper [4], whereas the entire schema of the UNIMARC format is given in the paper [31]. The subject of the paper [5] is the creation of an XML schema for the MARC 21 format, whereas the entire schema of the MARC 21 format that is developed within the same project is given in [30].

Structures of the UNIMARC and MARC 21 formats are similar. The XML schema that describes MARC 21 and UNIMARC format is given in [9]. An occurrence of the scheme is an XML document of the corresponding format that contains all data referred to the bibliographic UNIMARC and MARC 21 formats. This XML document is used as input information for the cataloguing editor of the BISIS system. Because of the volume of the paper, only the part of the schema by which is modeled its root element containing basic concepts of the format is presented in Figure 1. These basic concepts of the format are:

- leader (MARC21), record label (UNIMARC),
- directory,
- control fields,
- fields,
- code books.

Record label, which is modeled by the *leader* element, is placed at the beginning of each record and it contains data for record processing. It consists of 24 characters that are designated by 00 through 23. Character position indicates data that it contains. The directory (*directory* element) consists of a three-digit tag for each field, its length and beginning position of a character in relation to the first data field.

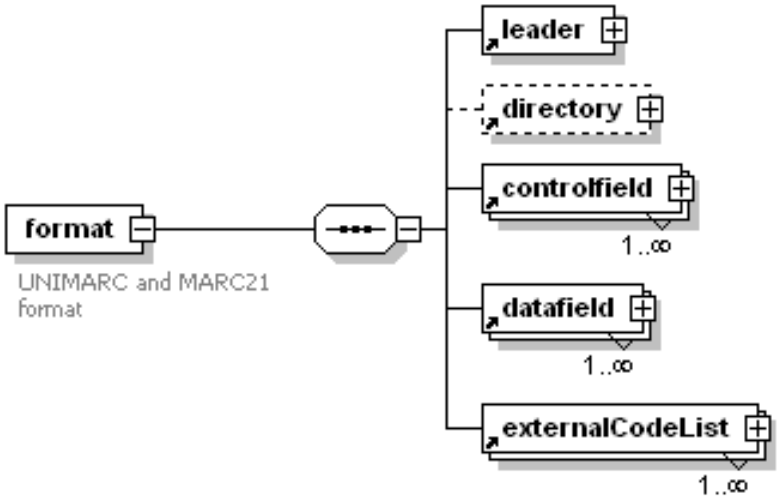

Figure 1: Root element of the XML schema for UNIMARC and MARC 21 formats

Control field (*controlfield* element) is defined by its name, description of the data that is placed in the field and repeatability feature. Control fields within the UNIMARC format do not exist under that name as it is the case in the MARC 21 format, but there are the fields that have structure and characteristics of the control fields, and these fields have the characteristics of mandatoriness.

Fields (*datafield* element) can be information carriers by themselves or can contain two indicators and a definite set of subfields at the most. The indicators additionally define content of a field, and subfields are information carriers in that case. The fields are described by the field names, descriptions and repeatability characteristics. Within the UNIMARC format, the fields are related to the characteristics of mandatoriness.

Code book (*externalCodeList* element) is a code list that is prescribed by a standard or it is introduced by an institution. It can be related to only one or more format elements and it consists of a set of codes with concatenated meanings.

The above concepts are complex as they can contain other format concepts.

An object model of bibliographic format is created on the basis of a given XML schema. The model is used in the editor for cataloguing of bibliographic records in the BISIS system. A class diagram of the object model is presented on Figure 2. Each schema element is presented by a special class on the diagram. Class names correspond to the names with an F prefix in the XML schema, designating that it is related to the format. The class attributes that correspond to the attributes of the XML schema are presented on the class diagram in Figure 2. The class operations consist of the *set* and *get* methods for each attribute by

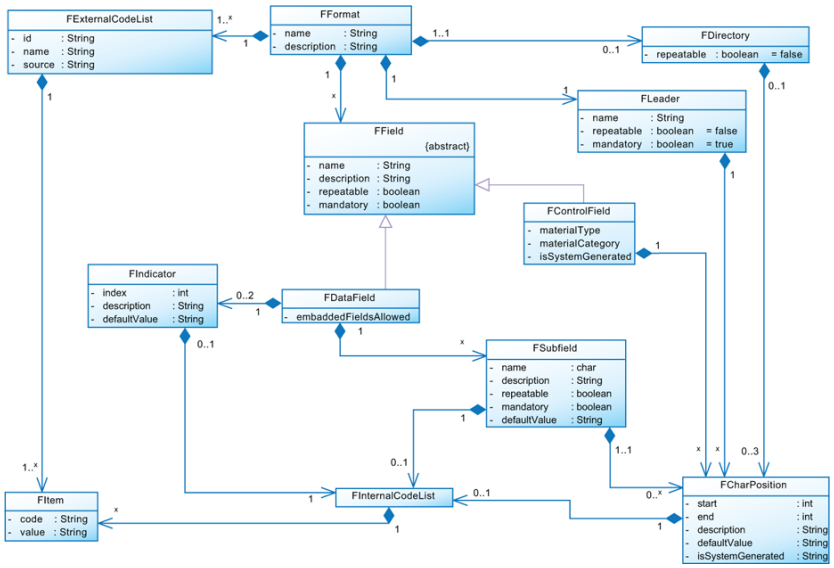

Figure 2: Class diagram of the bibliographic format object model

which it is possible to update data about the format within the object model.

2.2 XML schema of bibliographic records

One of the projects of introducing XML into libraries is MARC XML developed in the Library of Congress and described in [17]. The article stated that the project is launched in order to standardize the exchange of MARC records in XML format and to give the XML schema of records [18] that was created within the same project.

Also, the *Open Archives Initiative* organization proposed [24] a solution for modeling bibliographic records in the MARC 21 standard and in the XMLSchema language. The XML Schema supports all concepts of the MARC 21 records and it is similar to the schema defined within the MARC XML project.

Creation of the XML schema of bibliographic records in the UNIMARC standard is reduced to updating one of two XML schemas of the MARC 21 bibliographic records by introducing elements that model the existence of secondary fields. That XML schema of bibliographic record is used in the BISIS system.

A basic role of object models of the bibliographic records is the representation of data about record in an application. The data about record can be read into an object model from the XML documents and also can be stored from the object model in these XML documents. In order that correct mapping data of the XML documents could be performed onto the object model, and vice versa, it is necessary that the object model corresponds to the model of the XML schema according to which the XML bibliographic records are created.

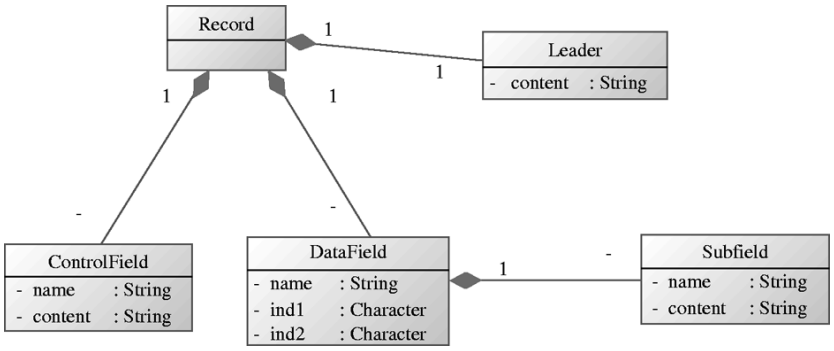

Figure 3: Class diagram of a bibliographic record object model within MARC 21 standard

A class diagram of bibliographic record object model within the MARC 21 format is given in Figure 3. The object model is created in accordance with the bibliographic record schema developed in the scope of the MARC XML project.

A bibliographic record is presented by the *Record* class. One record contains exactly one leader that is presented by the *Leader* class the attribute *content* of which represents the content of the record leader, and its length is 24 characters. The bibliographic record within the MARC 21 standard contains two kinds of fields, control fields and fields containing subfields. Control fields are modeled by *ControlField* class that has two attributes, *name* attribute that presents field tag consisted of three digits and *content* that represents the content of a control field. Record fields that contain the first and the second indicator, *ind1* and *ind2,* and a set of subfields presented by the *Subfield* class that is connected with *DataField* class by a composition relation, are represented by *DataField* class. A subfield is determined by its designation (*name* attribute) and content (*content* attribute).

The structure of a bibliographic record within the UNIMARC format is similar to the record structure within the MARC 21 format. A record leader, control fields and fields that contain subfields appear in both formats. Besides, there are defined secondary fields in the UNIMARC format and these appear as the subfield content. The object model of the UNIMARC record can be obtained on the base of the object model of the MARC 21 record adding another relation between the *DataField* and *Subfield* classes by which a secondary field is presented. The relation is presented in Figure 4.

3 Application of XML technologies for bibliographic record exchange

The results achieved in the scope of the development of the BISIS library software system are presented in this section and especially of the part related to the

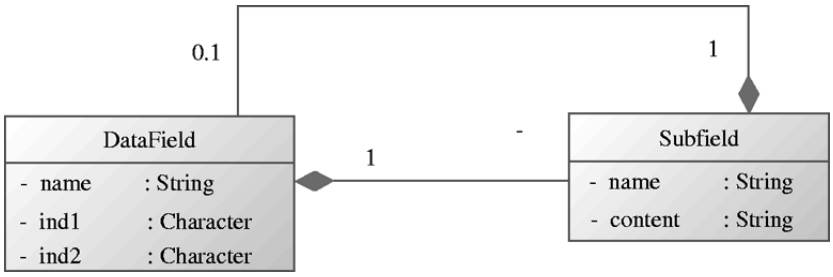

Figure 4: Secondary field

application of the XML (*Extensible Markup Language*) technologies for exchange of the bibliographic records in the electronic form. In order that the searches of different databases accessible via Internet and download of bibliographic records would be enabled it is necessary to have a communication protocol that is independent of the environment (database, platform, programming language). One of such protocols, that is primarily intended for the communication among library information systems, is described by the Z39.50 standard [12].

One of the main tasks defined by the standard is database searching. Six types of queries are defined by the standard:

1. Type-0 can be used if there exists preliminary agreement out of the standard scope.

2. Type-1 is represented by the reverse Polish notation and it is defined as obligatory query type in accordance with the standard Z39.50.

3. Type-2 is the query defined in accordance with the standard ISO8777.

4. Type-100 is the query known under the name *Common Command Language* and it is not defined in accordance with the standard Z39.50.

5. Type-101 represents broadening of the Type-1 query. The query is similar to the query Type-1, but it is not dependent on the Z39.50 protocol version.

6. Type-102 (*Ranked List Query*) should be defined in the following versions of the standard.

In addition to the basic services that a client/server should satisfy, the standard restricts the implementers of the Z39.50 protocol by having to support the Type-1 query. However, this does not mean that all operators or all possibilities for operands have to be supported. The Type-1 query consists of several isolated terms that are searched for and, for each term, certain attributes are defined. In the Type-1 query several terms can be related by AND, OR and NOT logical operators. The terms and operators are expressed in Reverse Polish notation (postfix records).

The Z39.50 query languages can be represented by using XML and as a part of the JAFER [16] project a proposal of the XML schema of the Type-1 query language is given. A proposal of the XML schema of the Type-100 query language is given in the next section. The representation of a query language by the XML schema is motivated by the fact that a particular query that a user enters into the application for searching bibliographic records via the Z39.50 protocol can be represented as an instance of the corresponding schema. Then, it is possible to validate that query in relation to the corresponding schema and, additionally, XML documents are much more suitable for further processing than an arbitrary text.

3.1 Type-100 Z39.50 query

This query is known under the name *Common Command Language* and is defined in the scope of the ANSI/NISO Z39.58 standard *Common Command Language for Online Interactive*. The primary goal of the query language was to enable query input from the command line on the basis of which it was named. The syntax of this type of query is given as follows.

CCL-Find::= CCL-Find Op Elements | Elements
Op::="and" | "or" | "not"
Elements::= '(' CCL-Find ')' |
Set |
Terms |
Qualifiers Relation Terms |
Qualifiers Relation '(' CCL-Find ')' |
Qualifiers '=' string '-' string
Set::= 'set' = string
Terms::= Terms Prox Term | Term
Term::=Term string | string
Qualifiers::= Qualifiers ',' string | string
Relation::= '=' | '≥' | '≤' | '¡¿' | '¿' | '¡'
Prox::= '%' | '!'

Similarly as the Type-1 query, this query can also consist of many elements related by logic operators. The operators can be AND, OR and NOT. Besides the operators, in this query type also appears *Prox* operator, that explains in more details the relation between two terms that are searched. The sequence of words that are searched or the distance between one expression to the another can be defined.

An operand can be an earlier obtained set of results that has its name. It can also be a sequence of strings, without any prefixes, which is a novelty compared to the Type-1 query. With the operand of this type, the user of the search system specifies only expressions by which he/she wants to search, but he/she does not specify any prefix. The system that implements this type of queries defines some basic indexes and then this search is done only by the indexes.

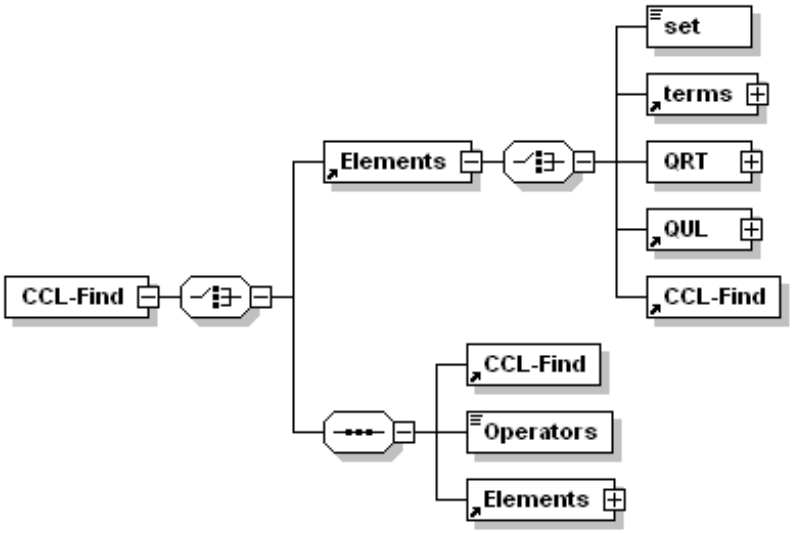

Figure 5: *CCL-Find* element

The next option for the operand is a qualified name that is connected to a term or a new query by a particular relation. The qualified name represents some predefined prefix, for example, author name, title, etc. With this type of query it is possible to set also a certain range of values as an operand, but only for the corresponding qualified names, something that was not possible in the query of Type-1. For example, we can set the query that will retrieve all books for the period of 1990 up to 2000.

3.2 Query XML schema

In this section we propose an XML schema that models the Type-100 query. The root element is the **CCL-Find** element that is shown in Figure 5.

The **Operators** element represents the logical operators that can be used in a query. It is of basic type, and its values can be AND, OR or NOT.

The **Elements** element represents the operands that can be used in a query. The **set** element is of basic type and contains information about the name of the previously obtained result set.

The **terms** element is of the composite type, and it is shown in Figure 6. The concrete values of the expressions that are searched are saved in the **term** element. The **proxy** element is of basic type, and it represents the operator that more closely designates the relation between two terms. This element can have two values: the value %, which designates that the order of occurrences of searched expressions is not relevant, and the value !, which designates that the order of occurrences of searched expressions is specified.

The **QRT** element is shown in Figure 7. The **Qualifiers** element represents

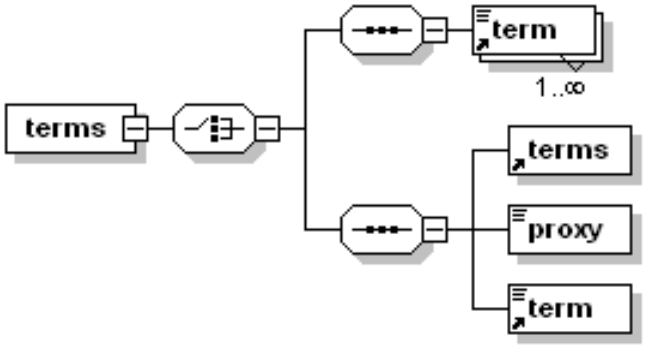

Figure 6: *terms* element

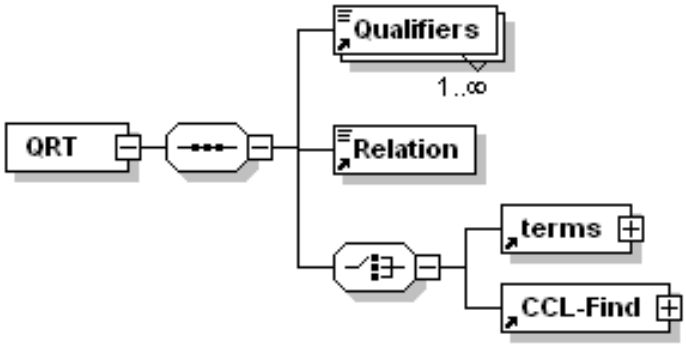

Figure 7: *QRT* element

a qualified name that will be used in the search, for example, an AU that represents an author. The **Relation** element is of basic type, and its values can be =, >=, <=, <>, > and <. It designates a corresponding relation between an expression that is searched and a qualifier.

The **QUL** element shown in Figure 8 ensures that a query operand can represent a qualifier whose value can be from a particular range. The lower limit of the range is defined by the **Lowstring** element, whereas the upper limit of the range is defined by the **Uperstring** element.

It is possible to define in a similar way the XML schemas of other query languages defined by the Z39.50 standard.

4 XML in the circulation system

NCIP (NISO Circulation Interchange Protocol) or Z39.83 is a protocol for exchange of data about users that is standardized by *NISO National Information*

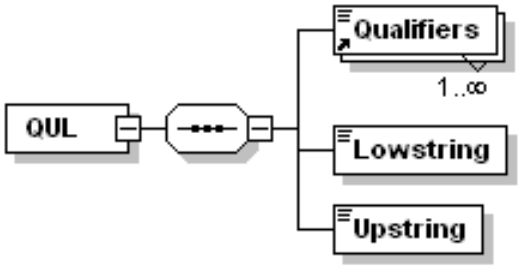

Figure 8: *QUL* element

Standards Organization). The syntax and semantics of the messages are defined by the NCIP protocol in order that the functions of checking out and discharging books as well as of controlling access to electronic resources be enabled. The sets of objects, services, messages and elements in the messages exchanged by the applications are defined by the protocol.

XML was chosen for encoding messages with the message structure described by the XML schema. A current version of the NCIP XML schema that defines the messages can be found on the official site of the NCIP standard [23]. The NCIP protocol is free of charge and open to public. In addition to that, the site offers information about the activities and work of the implementation group, as well as information about the standard implementation itself within different library systems.

A consortium stands for an association of libraries that mutually cooperate in some area, which results in the improvement of services that libraries offer to their users. The circulation within a consortium is a library process in which the users of a library are enabled to borrow books from other libraries within the consortium. In order to enable this activity, it is necessary to provide the exchange of user data on the consortium level. In this process, the NCIP standard provides the data exchange.

Several application profiles were created for putting protocols into practice as a support to protocol implementation. By analyzing the existing application profiles, the *User Agency Manages Transaction* (*DCB-2*) was found to be the most suitable application profile for the implementation of the protocols in the BISIS system. This profile describes the use of the NCIP protocol in the environment where the circulation transactions are managed by the library to which a user belongs. In other words, it is not the policies of the library in which the user checks out an item but the ones of his/her home library that are applicable to the patron. With the aim of implementing users' checking out functionalities within a consortium:

- publications are checked out at a location where they are located (that is, are not sent to other locations),
- publications are returned at a same location where they were checked out,

- there is no financial remuneration for checking out.

By analyzing the profiles, six services were found necessary to be implemented: *Lookup User*, *Lookup Item*, *Check Out Item*, *Item Renewed*, *Renew Item* and *Check Item In*.

While defining the protocols, different library practices regarding circulation were taken into account. Therefore, a protocol was defined so that it can be extendable in regard to the needs of a library or the consortium. It was left to the discretion of libraries to define lists of values for certain elements that can be found in these elements. These lists are accessible to the libraries that participate in the exchange and uniquely determined by the URI address. Also, a possibility is left to the libraries to add new elements to the objects in relation to the ones defined by the protocol. It was necessary to define several value lists for the implementation of the NCIP protocol within the BISIS system. These lists are defined in relation to the code books that exist in the data model.

The services defined by the NCIP protocol consist of two messages – request and response – through which data is exchanged. We will use the example of the *Lookup User* service to illustrate the implementation mode of the NCIP protocol within the BISIS system. This service is initiated by the library that is not user's parent library and with the goal of obtaining user information. The request message that is sent by this service is defined by the NCIP XML element of the *LookupUser* schema. The XML document that represents the request message of the *Lookup User* service is shown in Listing 1 and is created on the basis of the NCIP element of the XML schema.

In an XML document one can notice a structure and content of the message. The *InitiationHeader* element that represents the message header has in this case two elements – *FromAgencyId* and *ToAgencyId* – containing the information about the library sending the request and the user to whom it is sent. Both elements contain the *AgencyId* element carrying the information about a particular library. The *AgencyId* element is of the *SchemeValuePair* type. This type is defined with the aim of adapting protocols to the local library practices. The elements of this type have the *Scheme* attribute in which the URI address is quoted of the value list from which the element values are taken. In the case of the *AgencyId* element, which belongs to this type, as well as of the BISIS system, a value list of short titles of the libraries that are consortium members and exchange messages can be found at http://bisis.ns.ac.rs/NCIPschemes/AgencyID.scm. In the example shown in the listing one can notice that the message is sent by the library with the *ftnns* designation and received by the library with the *dmins* designation.

After the *InitiationHeader* element a *UserId* element follows on the basis of which the user, whose data is searched out, is identified. The element represents the sequence of the *UserIdentifierType* and *UserIdentifierValue* elements. In this case *AgencyId* element determines to which library a user belongs. The *UserIdentifierType* element defines which data is used for the user identification. The element is of the *SchemeValuePair* type and it has the *Scheme* attribute in which the URI address of the value list in which the data usable for the user

identification is defined. In the given example, it is Patron ID data that represents the user number. The *UserIdentifierValue* element presents a concrete value of data that is defined in the *UserIdentifierType* and, in this case, the number of concrete user.

The *InitiationHeader* element is followed by the *UserId* element on the basis of which the user whose data is sought is identified. This element represents a sequence of the *UserIdentifierType* and *UserIdentifierValue* elements. In this case the *AgencyId* element determines the library to which a user belongs. The *UserIdentifierType* element defines the data used for user identification. This element is of the *SchemeValuePair* type and has the *Scheme* attribute with the URI address of the value list in which the data usable for user identification is defined. In the example shown, it is the Patron ID data that represents the user's number. The *UserIdentifierValue* element stands for a particular value of the data defined in the *UserIdentifierType* element and, in this case, the number of a particular user.

Following the *UserId* element, are listed the *UserElementType* elements which define the data about the user that is being sought. The *UserElementType* element is also of the *SchemeValuePair* type and the values of these elements are taken from the defined value list. The elements of the *User* object that is defined by the protocol are found in the value list for this element. Added after the *UserElementType* elements in the example shown, the *LoanedItemsDesired* element indicates that data about the items checked out by the user is also requested.

```
<?xml version="1.0" encoding="UTF-8"?>
<ncip:LookupUser xmlns:ncip="http://www.niso.org/2008/ncip"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.niso.org/2008/ncip
       ncip_v2_0.xsd ">
<ncip:InitiationHeader>
 <ncip:FromAgencyId>
  <ncip:AgencyId
   ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/AgencyID.scm">
            ftnns
  </ncip:AgencyId>
 </ncip:FromAgencyId>
 <ncip:ToAgencyId>
  <ncip:AgencyId
   ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/AgencyID.scm">
            dmins
  </ncip:AgencyId>
 </ncip:ToAgencyId>
</ncip:InitiationHeader>
<ncip:UserId>
 <ncip:AgencyId
   ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/AgencyID.scm">
```

```
            dmins
  </ncip:AgencyId>
  <ncip:UserIdentifierType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserIdentifierType.scm">
            Patron ID
  </ncip:UserIdentifierType>
  <ncip:UserIdentifierValue>
            00000001369
  </ncip:UserIdentifierValue>
</ncip:UserId>
<ncip:UserElementType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserElementType.scm">
            Name Information
</ncip:UserElementType>
<ncip:UserElementType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserElementType.scm">
            User Address Information
</ncip:UserElementType>
<ncip:UserElementType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserElementType.scm">
            User Language
</ncip:UserElementType>
<ncip:UserElementType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserElementType.scm">
            User Privilege
</ncip:UserElementType>
<ncip:UserElementType
    ncip:Scheme="http://bisis.uns.ac.rs/NCIPschemes/UserElementType.scm">
            Block Or Trap
</ncip:UserElementType>
<ncip:LoanedItemsDesired/>
</ncip:LookupUser>
```

Listing 1 *LookupUser* message

As a response to the message within the *Lookup User* service a message defined by the NCIP XML element of the *LookupUserResponse* schema is created. Based on this element, an *XML* document is created to represent the message *LookupUserResponse*. Because of its volume, this message is omitted in the paper.

5 Conclusion

The use of XML technologies in the BISIS library system development is described in this paper. There are numerous projects that apply XML technology

to library information systems and the number of libraries that store bibliographic records in XML is increasing. The application of XML in libraries has different purposes, such as, the representation of bibliographic records, transport and validation of records well as transport of other data used in the library management system.

Library information systems are mainly based on bibliographic formats, the best known of which are UNIMARC and MARC 21. An XML schema that describes both MARC 21 and UNIMARC format, as well as the object model developed on the basis of that schema are described in this paper. An instance of this schema is used in the editor for cataloguing the BISIS system. Also, the paper describes the XML schemas for bibliographic records on the basis of which an object model of the bibliographic record used in the BISIS system is created.

The BISIS system supports searching and downloading bibliographic records via the Z39.50 protocol. In this paper, XML is used for the description of user-defined queries and a proposal of the XML schema for one of the query languages proposed by the Z39.50 protocol is given.

XML is used for data exchange in the process of circulation within a consortium. The NCIP protocol is intended for exchanging data about library users. The syntax of the protocol is described by the XML schema and messages are exchanged in the XML format. The paper gives a short description of the protocol, as well as an example of a message in accordance with the protocol.

XML is a widely accepted standard for presenting data, and there are numerous software tools for working with XML documents. Therefore, storing bibliographic format data in XML documents brings many advantages. An XML document of the bibliographic format can be used by a huge number of applications regardless of the programming language and the platform it uses. The system for cataloguing becomes independent of the bibliographic format since, all the changes affecting only the XML document that is used as the input information, can be easily adjusted for all the changes in the format. Moreover, by using XML, the system gains in its interoperability. Namely, the interoperability with other systems is achieved through the exchange of data in an XML format.

6 Acknowledgement

This paper is part of the scientific project *Abstract Models and Applications in Computer Science* no.144017A, supported by the Ministry of Science of the Republic of Serbia

References

[1] Boberić D., Milosavljević B., Generating library material reports in software system BISIS. Proceedings of the 4th International Conference on

Engineering Technologies ICET 2009. Novi Sad, 2009, 133-137.

[2] Boberić, D., Surla D., XML Editor for Search and Retrieval of Bibliographic Records in the Z39.50 Standard, The Electronic Library.

[3] BookMARC, Bibliographic Information Services. www.bookmarc.pt/tvs/ (last accessed 13 October 2010)

[4] Carvalho, J., An XML representation of the UNIMARC manual: a working prototype. World Library and Information Congress: 71th IFLA General Conference and Council, Oslo, 2005, http://www.ifla.org/IV/ifla71/papers/199e-Carvalho.pdf

[5] Carvalho, J., Cordeiro, M. I., Lopes, A., Vieira, M., Meta-information about MARC: an XML framework for validation, explanation and help systems. Library Hi Tech, 22(2) (2004), 131-137.

[6] CERIF. http://www.eurocris.org/Index.php?page=CERIFintroduction&t=1 (last accessed 13 October 2010)

[7] Dimić B., Surla D., XML Editor for UNIMARC and MARC 21 cataloguing. The Electronic Library, 27(3) (2009), 264-473.

[8] Dimić, B., Milosavljević, B., Customisation of cataloguing in library software system BISIS. Proceedings of the 4th International conference on engineering technologies ICET 2009. Novi Sad, 2009, 127-131.

[9] Dimić, B., Milosavljević, B., Surla, D., XML schema for UNIMARC and MARC 21 formats. The Electronic Library, 28(2) (2010), 245-262.

[10] euroCRIS. http://www.eurocris.org/ (last accessed 13 October 2010)

[11] Hjørgensen, P.H., XML standards and library applications. ELAG 2001, (2001) www.stk.cz/elag2001/Papers/Poul_HenrikJoergensen/Show.html

[12] Information Retrieval (Z39.50): Application Service Definition and Protocol Specification. http://www.loc.gov/z3950/agency/Z39-50-2003.pdf (last accessed 13 October 2010)

[13] Ivanović, D., Milosavljević, G., Milosavljević, B., Surla, D., A CERIF-compatible research management system based on the MARC 21 format. Program: electronic library and information systems, 44(3) (2010), 229-251.

[14] Ivanović, D., Surla, D., Konjović, Z, CERIF compatible data model based on MARC 21 format. The Electronic Library, (2010), in print.

[15] Ivanović, D., Surla, D., Racković, M., A CERIF data model extension for evaluation and quantitative expression of scientific research results. Scientometrics, doi:10.1007/s11192-010-0228-2, (2010)

[16] JAFER. http://www.jafer.org/ (last accessed 13 October 2010)

[17] Keith, C., Using XSLT to manipulate MARC metadata. Library Hi Tech, 22(2) (2004), 122-130.

[18] MARC21Slim XML Schema. http://www.loc.gov/standards/marcxml/schema/MARC21slim.xsd (last accessed 13 October 2010)

[19] Milosavljević B., Dimić B., XML schema of UNIMARC format variant and bibliographic record in BISIS software system. Novi Sad Journal of Mathematics, Vol. 37 No. 1 (2007), 115-128.

[20] Milosavljević, B., Boberić, D., Surla, D., Retrieval of Bibliographic Records Using Apache Lucene. The Electronic Library, 28(4) (2010), 525-539.

[21] Milosavljević, B., Tešendić, D., Software Architecture of Distributed Client/Server Library Circulation System. The Electronic Library, 28(2) (2010), 286-299.

[22] Milosavljević, G., Ivanović, D., Surla, D., Milosavljević, B., Automated Construction of the User Interface for a CERIF-Compliant Research Management System. The Electronic Library, (2010), in print.

[23] NCIP Implementation Group website. http://www.ncip.info/ (last accessed 13 October 2010)

[24] Open Archives Initiative. http://www.openarchives.org/ (last accessed 13 October 2010)

[25] Radjenović, J., Milosavljević, M., Surla, D., Modelling and implementation of catalogue cards using FreeMarker. Program: electronic library and information systems, 43(1) (2009), 63-76.

[26] Rudić G., Surla, D., Conversion of bibliographic records to MARC 21 format. The Electronic Library, 27(6) (2009), 950-967.

[27] Tešendić, D. (2009), Dynamic model of the library circulation system. Proceedings of the 4th International Conference on Engineering Technologies ICET 2009, Novi Sad, 2009, 139-144.

[28] Tešendić, D., A database model for library material usage. Novi Sad Journal of Mathematics, Vol. 37 No. 1 (2007), 155-162.

[29] Tešendić, D., Milosavljević, B., Surla, D., A Library Circulation System for City and Special Libraries. The Electronic Library, 27(1) (2009), 162–186.

[30] XML Schema Marc21. http://www.bookmarc.pt/documentation/marcdoc/xsd.html (last accessed 13 October 2010)

[31] XML Schema Unimarc. http://www.bookmarc.pt/unimarc/schema/FORMAT.html (last accessed 13 October 2010)

[32] XML4Lib: XML in Libraries Mailing List. `http://lists.webjunction.org/xml4lib/` (last accessed 13 October 2010) 27(3) (2009), 474-495.

Received by the editors October 27, 2010