

# PICA XML Version 1.0

- Encoding PICA+ in XML
  - Design Considerations
  - PICA+ XML documentation
    - Informal description
    - PICA DTD
    - PICA XML Schema
    - Notes
  - Validation
  - Tools & Utilities
  - References
- Appendices
  - Example Document

## Encoding PICA+ in XML

This document defines a standard to encode **PICA+ in XML** (in short: *pica-XML*) for working with PICA+ data in XML environments. PICA+ XML allows transformations between PICA+ and other metadata formats, as well as presentation, validation, analysis, and editing of PICA+ data with XML-based tools.

- **Editors:** Jakob Voss <jakob.voss@gbv.de>
- **Status:** Version 1.0 (2008-07-03)

## Design Considerations

More and more interchange formats that are used in the library world are based on XML (MODS, METS...) or have a representation in XML (MARCXML for MARC, MABxml for MAB...). This document defines a representation to encode PICA+ in XML. PICA+ is the internal metadata format that is used in the CBS and LBS software by OCLC PICA. The conversion between PICA+ and PICA+ XML is lossless. By using XML as the structure for PICA+ records, users can more easily write their own tools to consume, manipulate, validate, and convert PICA+ data. PICA+ XML is not meant to replace normalized PICA+ but can act as its counterpart in the XML world.

## PICA+ XML documentation

### Informal description

PICA+ XML records contain of a record element that includes a number of datafield elements. Each datafield has a tag attribute and may have an additional occurrence attribute. Fields contain one or more subfield elements, that each have a code attribute. In subfields text is allowed. Multiple PICA+ XML records can be combined with a collection parent element. There are additional restrictions for attribute values of tag, occurrence, and code. The character set of subfield content is always full Unicode. The XML namespace for PICA XML is info:srw/schema/5/picaXML-v1.0

### PICA DTD

The structure of PICA XML can be defined with a simple Document Type Definition (DTD). The DTD defines a superset of PICA XML without restriction on attribute values and without namespace requirement.

```
<!ELEMENT collection (record+)>
<!ELEMENT record (datafield+)>
<!ELEMENT datafield (subfield+)>
<!ATTLIST datafield
  tag CDATA #REQUIRED
  occurrence CDATA #IMPLIED>
<!ELEMENT subfield (#PCDATA)*>
<!ATTLIST subfield code CDATA #IMPLIED>
```

Please note that this DTD is not official and does not include the namespace declaration (xmlns="info:srw/schema/5/picaXML-v1.0").

### PICA XML Schema

The XML Schema contains the full definition of PICA XML. Attribute values are restricted as follows:

- Tag codes (attribute tag of element datafield) must match the pattern [0-9][0-9][0-9][A-Z@], that is three digits followed by an upper case letter or the ampersand sign '@' (The first digit is also known as 'level', the second and third digit are also known as 'type', and the last character is also known als 'indicator')
- Occurrences (attribute occurrence of element datafield) must match the pattern [0-9][0-9], that is two digits.
- Subfield codes (attribute code of element subfield) must match the pattern [0-9a-zA-Z], that is a digit or a letter.

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  PICA XML 1.0 - XML Schema for XML representation of PICA data
  Author: Jakob Voss <jakob.voss@gbv.de>
  Date: 2009-07-06
-->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="info:srw/schema/5/picaXML-v1.0"
  xmlns="info:srw/schema/5/picaXML-v1.0">

  <xs:element name="collection">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="unbounded" ref="record"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="record" >
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="unbounded" ref="datafield"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="datafield">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="unbounded" ref="subfield"/>
      </xs:sequence>
      <xs:attribute name="tag" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="[0-9][0-9][0-9][A-Z@]"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="occurrence" use="optional">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="[0-9][0-9]"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>

  <xs:element name="subfield">
    <xs:complexType mixed="true">
      <xs:attribute name="code" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="[0-9a-zA-Z]"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

A copy of the XML Schema can be found at <http://www.loc.gov/standards/sru/recordSchemas/pica-xml-v1-0.xsd>

## Notes

The old SRU interface of PSI already produced some XML representation of PICA+ data with seems to conform to this standard - but the data inside the <srw:recordData> element misses a namespace. See [this example](#)

The namespace is now officially listed at <http://www.loc.gov/standards/sru/recordSchemas/>.

## Validation

Validation is a fundamental process to ensure quality of data. Early and frequent validation prevents errors that are difficult to track and repair in later steps of data transformation. A standard that cannot be used to automatically test objects against it, is pretty useless and will surely be disregarded. To ensure conformance of PICA XML several levels of validation are possible:

1. Basic XML validation according to the PICA DTD  
This can be done by adding the DTD to an XML document and parse it with a validating XML parser

2. Basic XML validation according to the PICA XML Schema  
This can be done by adding the XML Schema to an XML document and parse it with a validating XML parser
3. Validation of PICA+ datafield and subfield structure  
This requires lists of required and allowed datafields and subfields, depending on the cataloguing rules.
4. Validation of PICA+ record content  
This requires a deeper look into cataloguing rules.

## Tools & Utilities

- You need a validating XML parser.
- Readers and Serializers to convert from and to normalized PICA+ data are being worked on.

## References

- **MABxml** - MAB in XML. <http://www.ddb.de/standardisierung/formate/mabxml.htm>
- **MARCXML** - MARX 21 XML Schema. <http://www.loc.gov/standards/marcxml/>
- **OAI-PMH** - Open Archives Initiative - Protocol for Metadata Harvesting v. 2.0. <http://www.openarchives.org/OAI/openarchivesprotocol.html>
- **XML** - Extensible Markup Language (XML) 1.1 (Second Edition). <http://www.w3.org/TR/2006/REC-xml11-20060816/>
- **XMLNS** - Namespaces in XML 1.0 (Second Edition). W3C Recommendation 16 August 2006. <http://www.w3.org/TR/2006/REC-xml-names-20060816>

## Appendices

### Example Document

```
<?xml version="1.0" encoding="UTF-8"?>
<record xmlns="info:srw/schema/5/picaXML-v1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="info:srw/schema/5/picaXML-v1.0 http://www.oclc.org/xml/picaplus.xsd">
  <!-- please change http://www.oclc.org/xml/picaplus.xsd to your local copy of the XML schema or
  use a catalog file! -->
  <datafield tag="001@">
    <subfield code="0">0917:14-03-05</subfield>
  </datafield>
  <datafield tag="001B">
    <subfield code="0">0917:23-03-05</subfield>
    <subfield code="t">16:15:13.000</subfield>
  </datafield>
  <datafield tag="001D">
    <subfield code="0">0917:23-03-05</subfield>
  </datafield>
  <datafield tag="001X">
    <subfield code="0">0</subfield>
  </datafield>
  <datafield tag="002@">
    <subfield code="0">Aau</subfield>
  </datafield>
  <datafield tag="003@">
    <subfield code="0">481592954</subfield>
  </datafield>
  <datafield tag="004A">
    <subfield code="0">3774250936</subfield>
  </datafield>
  <datafield tag="011@">
    <subfield code="a">2004</subfield>
  </datafield>
  <datafield tag="021A">
    <subfield code="a">Der Hamster</subfield>
    <subfield code="d">artgerecht halten, gesund ernähren, richtig verstehen</subfield>
    <subfield code="h">Peter Hollmann</subfield>
  </datafield>
  <datafield tag="028A">
    <subfield code="d">Peter</subfield>
    <subfield code="a">Hollmann</subfield>
  </datafield>
  <datafield tag="032@">
    <subfield code="a">5. Aufl</subfield>
  </datafield>
  <datafield tag="033A">
    <subfield code="p">München</subfield>
    <subfield code="n">Gräfe und Unzer</subfield>
  </datafield>
  <datafield tag="034D">
    <subfield code="a">127 S</subfield>
  </datafield>
  <datafield tag="034M">
```

```
<subfield code="a">zahlr. Ill</subfield>
</datafield>
<datafield tag="036E">
  <subfield code="a">Mein Heimtier</subfield>
</datafield>
<datafield tag="044K">
  <subfield code="a">Ratgeber</subfield>
</datafield>
<datafield tag="044L">
  <subfield code="S"> </subfield>
  <subfield code="a">Ratgeber</subfield>
</datafield>
<datafield tag="044L" occurrence="01">
  <subfield code="S"> </subfield>
  <subfield code="a">Hamsterhaltung</subfield>
</datafield>
<datafield tag="045B">
  <subfield code="a">Xbp 3</subfield>
</datafield>
</record>
```

Diese Seite wurde zuletzt am 28. März 2014 um 09:36 Uhr geändert.