INTERNATIONAL STANDARD

ISO 16684-3

First edition
2021-08

Graphic technology — Extensible metadata platform (XMP) specification —

Part 3: JSON-LD serialization of XMP

Reference number
ISO 16684-3:2021(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 171, *Document management applications*, Subcommittee SC 2, *Document file formats, EDMS systems and authenticity of information*.

A list of all parts in the ISO 16684 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

The extensible metadata platform (XMP) was introduced by Adobe Systems Incorporated in 2001 and has since established itself as a critical technology for improving business efficiency in many industries. The current serialization of XMP model is available in RDF/XML format as described in ISO 16684-1. This document provides a complete specification about how XMP can be serialized to JSON-LD.

JSON-LD is a lightweight syntax to serialize Linked Data in JSON (RFC4627). Its design allows existing JSON to be interpreted as linked data with minimal changes. It is also designed to be usable as RDF. JSON-LD can distinguish between values which are simple strings and internationalized resource identifiers (IRI). JSON itself has no built-in support for hyperlinks. Types of values can be indicated by an IRI in JSON-LD. But this cannot be done naturally in JSON. It provides the ability to annotate strings with their language.

Users of this document are cautioned that they are expected to be familiar with the documents listed as normative references and the terms used within those documents.[1]

[1] The PDF Association maintains an ongoing series of application notes for guiding developers and users of this document. It also retains copies of the specific non-ISO normative references of this document which are publicly available electronic documents.

INTERNATIONAL STANDARD ISO 16684-3:2021(E)

Graphic technology — Extensible metadata platform (XMP) specification —

Part 3: JSON-LD serialization of XMP

1 Scope

This document defines how the XMP data model can be serialized to JSON-LD.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

JSON-LD 1.1, A JSON-based Serialization for Linked Data, W3C Recommendation, 16 July 2020, https://www.w3.org/TR/json-ld11

BCP-47, Tags for Identifying Languages. A. Phillips; M. Davis. IETF. September 2009. IETF Best Current Practice. https://tools.ietf.org/html/bcp47

3 Terms and definitions

No terms and definitions are listed in this document.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

4 XMP – JSON-LD serialization

4.1 Namespace prefix serialization

Many of the terms used in this document come from JSON-LD 1.1.

According to ISO 16684-1, all names in XMP are XML expanded names, consisting of a namespace uniform resource identifier (URI) and a local name. The shortcut term of the namespace URI is defined inside the *@context* key in a JSON-LD document. All JSON-LD serializations of XMP shall contain a @*context* key.

EXAMPLE 1 XMP serialized as RDF/XML and its corresponding serialized JSON-LD with namespace prefixes.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/"
    xmlns:tiff="http://ns.adobe.com/tiff/1.0/">
    <rdf:Description rdf:about="">
            <xmp:Rating>4</xmp:Rating>
            <tiff:Orientation>1</tiff:Orientation>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "xmp": "http://ns.adobe.com/xap/1.0/",
        "tiff": "http://ns.adobe.com/tiff/1.0/"
    },
    "xmp:Rating": "4",
    "tiff:Orientation":"1",
    "@id": ""
}
```

Contexts can either be directly embedded into the document or be referenced. It is also possible to have more than one context at different points in a document. The set of contexts defined within a specific JSON object are referred to as local contexts.

NOTE The value of the *@context* key can either be a simple string (mapping the term to an IRI) or a JSON object.

If all the contexts in EXAMPLE 1 were combined into a single file that could be retrieved at the @*context* link shown in EXAMPLE 2 below, it can be referenced in the JSON-LD by adding a single line thus allowing the JSON-LD document to be expressed much more concisely.

EXAMPLE 2 Use of externally referenced *@context*.

```
{
    "@context": "http://www.example.com/contexts/context1.jsonld",
    "xmp:Rating": 4,
    "tiff:Orientation": 1,
    "@id": ""
}
```

4.2 Property Names in JSON-LD

Property names are represented as keys in JSON-LD document. These keys shall be of the form compact internationalized resource identifier (IRI). Compact IRI in JSON-LD is in the form of *prefix:suffix* where *prefix* refers to the namespace term defined in the context and *suffix* is local name in that namespace.

4.3 rdf:about serialization in JSON-LD as @id

ISO 16684-1:2019, 6.1 specifies that an XMP packet may contain a URI, called the AboutURI, that identifies the resource that the packet describes. Additionally, ISO 16684-1:2019, 7.4 adds that if the XMP data model has an AboutURI, that same URI shall be the value of an *rdf:about* attribute in each top-level *rdf:Description* element. Otherwise, the *rdf:about* attributes for all top level rdf:Description elements shall be present with an empty value. The *rdf:about* attribute shall not be used in more deeply nested *rdf:Description* elements.

EXAMPLE Serialized XMP and its corresponding serialized JSON-LD with AboutURI.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/">
    <rdf:Description rdf:about="http://www.example.com/abouturiEg/">
        <xmp:Rating>4</xmp:Rating>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "xmp": "http://ns.adobe.com/xap/1.0/"
    },
    "@id": "http://www.example.com/abouturiEg/",
    "xmp:Rating": 4
}
```

NOTE It is possible for an XMP packet to not contain an AboutURI and not have a physical association with the resource. Instead, there can be an external means of association.

If there is no AboutURI present (this can occur as XMP tolerates missing *rdf:about* attribute for compatibility with very early XMP), then an empty *@id* shall be included in the root document.

4.4 Use of @type JSON-LD keyword

JSON-LD has *@type* keyword that can be used to specify node type and value type. A node type specifies the type of thing that is being described. A value type specifies the data type of a particular value, such as an integer, a floating-point number or a date.

In the RDF/XML serialization, the value type is expressed by *rdf:datatype*. However, this attribute is not allowed in the JSON-LD serialization. The JSON-LD serialization allows for the use of *rdf:node* that corresponds to node type (ISO 16684-1:2019, 7.9.2.5). In addition, the *rdf:type* property can be used (but not a typed node, see ISO 16684-1:2019, 7.9.2.5).

EXAMPLE 1 Below is an example of *rdf:type* and its corresponding serialization in JSON-LD.

Serialized XMP Packet in RDF/XML:

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xe="http://ns.adobe.com/xmp-example/"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/">
    <rdf:Description rdf:about="">
        <rdf:type rdf:resource="http://ns.adobe.com/xmp-example/myType"/>
        <xmp:Rating>3</xmp:Rating>
     </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "xmp": "http://ns.adobe.com/xap/1.0/",
        "xe": "http://ns.adobe.com/xmp-example/"
    },
    "@id": "",
    "@type": "http://ns.adobe.com/xmp-example/myType",
    "xmp:Rating": 3
}
```

In RDF/XML serialization of XMP, the use of an inner typed node in XMP attaches an *rdf:type* qualifier to the containing element. The value of the *rdf:type* qualifier is a URI consisting of the typed node element's namespace URI concatenated with the local name. In the JSON-LD serialization, the equivalent of *rdf:type* is the *@type* keyword.

EXAMPLE 2 Inner typed node interpretation in XMP RDF/XML serialization.

```
<rdf:RDF
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:xe="http://ns.adobe.com/xmp-example/">
      <rdf:Description>
            <xe:Prop2>
              <rdf:Description>
                 <rdf:type
                rdf:resource="http://ns.adobe.com/xmp-example/myType"/>
                 <rdf:value rdf:parseType="Resource">
                      <xe:Field>value</xe:Field>
                 </rdf:value>
              </rdf:Description>
          </xe:Prop2>
      </rdf:Description>
</rdf:RDF >
```

Corresponding JSON-LD serialization

```
{
    "@context": {
        "xe": "http://ns.adobe.com/xmp-example/",
        "rdf":"http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    },
    "xe:Prop2": {
        "@type": "http://ns.adobe.com/xmp-example/myType",
        "rdf:value": {
            "xe:Field": "value"
        }
    }
}
```

Instead of *@type*, it is also acceptable to use *rdf:type* for consistency with the XML serialization.

EXAMPLE 3 Explicit *rdf:type* field in an XMP structure.

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xe="http://ns.adobe.com/xmp-example/">
    <rdf:Description>
    <!-- In XMP this is not related to the Typed Node usage. -->
        <xe:Prop3>
        <rdf:Description>
            <rdf:type
                rdf:resource="http://ns.adobe.com/xmp-example/myType"/>
            <xe:Field>value</xe:Field>
        </rdf:Description>
        </xe:Prop3>
    </rdf:Description>
</rdf:RDF>
```

Corresponding JSON-LD serialization

```
{
  "@context": {
    "xe": "http://ns.adobe.com/xmp-example/",
    "rdf":"http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  },
  "xe:Prop3": {
    "rdf:type": "http://ns.adobe.com/xmp-example/myType",
    "xe:Field": "value"
  }
}
```

4.5 XMP property forms serialization in JSON-LD

4.5.1 General

As per ISO 16684-1, values in the XMP data model have one of three forms:

- simple;
- structure;
- array.

The array form can be further classified as: unordered array, ordered array and alternative array. The fields in structures and the items in arrays can have any value form. There is no fixed bound on the complexity of XMP data modelling.

Subclauses 4.5.2 to 4.5.5 present the rules which shall be followed by conforming applications while serializing any XMP to JSON-LD.

4.5.2 Simple valued XMP Properties

A simple value may be either a string of unicode text as defined in ISO/IEC 10646, a number, or one of the three literal names defined in RFC 8259.

Any string value may be empty. A non-empty string value may be a URI, which when used as the value of the *@id* key shall be interpreted as an IRI.

EXAMPLE 1 Serialization for non-URI simple value in RDF/XML format and JSON-LD format.

Serialized XMP in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/">
    <rdf:Description rdf:about="">
        <xmp:Rating>4</xmp:Rating>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP in JSON-LD

```
{
    "@context": {
        "xmp": "http://ns.adobe.com/xap/1.0/"
    },
    "@id": "",
    "xmp:Rating":4
}
```

EXAMPLE 2 Serialization for URI simple value in RDF/XML format and JSON-LD format.

Serialized XMP in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/">
    <rdf:Description rdf:about="">
        <xmp:BaseURL rdf:resource="https://www.adobe.com/"/>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP in JSON-LD

```
{
    "@context": {
        "xmp": "http://ns.adobe.com/xap/1.0/"
     },
    "@id": "",
     "xmp:BaseURL": {
        "@id": "https://www.adobe.com/"
     }
}
```

4.5.3 Array valued XMP properties

4.5.3.1 General

As per ISO 16684-1, XMP property with an array value can be one of the three possible types: unordered, ordered and alternative. Arrays are unordered by default in JSON-LD.

4.5.3.2 Unordered array value

Unordered values are represented by *@set* keyword in JSON-LD. XMP unordered array values shall be serialized in the form of a key-value pair where the key shall be the property name represented as mentioned in 4.1 and its value shall be a JSON object. This JSON object shall have a property *@set* and its value shall be a JSON array of items.

EXAMPLE Serialization for unordered array value in RDF/XML format and JSON-LD format.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
    <rdf:Description rdf:about="">
        <dc:subject>
            <rdf:Bag>
                <rdf:li>XMP</rdf:li>
                <rdf:li>metadata</rdf:li>
                <rdf:li>ISO standard</rdf:li>
            </rdf:Bag>
        </dc:subject>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/"
     },
    "@id": "",
     "dc:subject": {
            "@set": ["XMP", "metadata", "ISO standard"]
     }
}
```

Since arrays in JSON-LD are considered to be unordered, below serialization may also be used.

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/"
     },
     "@id": "",
     "dc:subject": ["XMP", "metadata", "ISO standard"]
}
```

4.5.3.3 Ordered array value

Ordered array values shall be serialized in the form of a key-value pair where the key shall be the property name represented as mentioned in 4.1 and its value shall be a JSON object. This JSON object shall have a property *@list* (which is used for ordered arrays in JSON-LD) and its value shall be a JSON array of items.

EXAMPLE Serialization for ordered array value in RDF/XML format and JSON-LD format.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
    <rdf:Description rdf:about="">
        <dc:subject>
            <rdf:Seq>
                <rdf:li>XMP</rdf:li>
                <rdf:li>metadata</rdf:li>
                <rdf:li>ISO standard</rdf:li>
            </rdf:Seq>
        </dc:subject>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/"
      },
    "@id": "",
    "dc:subject": {
        "@list": ["XMP", "metadata", "ISO standard"]
    }
}
```

4.5.3.4 Alternative array value

Alternative array values shall be serialized in the form of a key-value pair where the key shall be the property name as described in 4.1.

It is not syntactically possible in JSON-LD to represent alternative arrays as it does not have serializations for RDF containers. Instead, use *rdf:Alt* to indicate the type of the property using *@type* keyword of JSON-LD.

EXAMPLE Serialization for alternative array value in RDF/XML format and JSON-LD format.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:ex="http://ns.example.com/example/">
    <rdf:Description rdf:about="">
        <ex:prop1>
            <rdf:Alt>
                <rdf:li>100</rdf:li>
                <rdf:li>120</rdf:li>
                <rdf:li>130</rdf:li>
            </rdf:Alt>
        </ex:prop1>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "ex": "http://ns.example.com/example/",
        "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    },
    "@id": "",
    "ex:prop1": {
        "@type": "rdf:Alt",
        "rdf:_1": 100,
        "rdf:_2": 120,
        "rdf:_3": 130
    }
}
```

NOTE Apart from alternative array, XMP also supports language alternative array whose serialization is discussed in the qualifier serialization section (4.5.5).

4.5.4 Structure valued XMP properties

The element content for each field in the structure shall follow the rules for properties, varying according to the form of the XMP value being serialized (simple, structure, or array), and whether the XMP value has qualifiers.

Structure property shall be serialized in the form of a key-value pair where the key shall be the property name and its value shall be JSON object. The content of the JSON object shall be serialized as per the rules described for JSON-LD serialization of simple, array or structure property.

EXAMPLE Serialization for Structure value in RDF/XML format and JSON-LD format.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xmpTPg="http://ns.adobe.com/xap/1.0/t/pg/"
    xmlns:stDim="http://ns.adobe.com/xap/1.0/sType/Dimensions#">
    <rdf:Description rdf:about="">
        <xmpTPg:MaxPageSize>
            <rdf:Description>
                <stDim:h>11.0</stDim:h>
                <stDim:w>8.5</stDim:w>
                <stDim:unit>inch</stDim:unit>
            </rdf:Description>
        </xmpTPg:MaxPageSize>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "stDim": "http://ns.adobe.com/xap/1.0/sType/Dimensions#",
        "xmpTPg": "http://ns.adobe.com/xap/1.0/t/pg/"
    },
    "@id": "",
    "xmpTPg:MaxPageSize": {
        "stDim:h": 11.0,
        "stDim:w": 8.5,
        "stDim:unit": "inch"
    }
}
```

4.5.5 Qualifiers

4.5.5.1 General

XMP qualifiers may be used to attach annotations to any XMP value, without changing the form of that value. The value of a qualifier may be any XMP value form (i.e. simple, array or structure).

4.5.5.2 Simple valued XMP property with qualifier

Simple valued XMP property with qualifiers shall be serialized as a key-value pair where the key shall be the property name and its value shall be a JSON object. The content of the JSON object shall consist of two key-value pairs. The first pair shall be the serialization of qualifier. The second pair shall have key *rdf:value* taking this from RDF vocabulary. Its value shall be JSON-LD serialization of simple value.

EXAMPLE Serialization for qualified simple value in RDF/XML format and JSON-LD format.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:xe="http://ns.adobe.com/xmp-example/">
    <rdf:Description rdf:about="">
        <dc:source>
            <rdf:Description>
                <rdf:value>Adobe XMP Specification, April 2010</rdf:value>
                <xe:qualifier1>artificial example1</xe:qualifier1>
            </rdf:Description>
        </dc:source>
        <xmp:BaseURL>
            <rdf:Description>
                <rdf:value rdf:resource="https://www.adobe.com/"/>
                <xe:qualifier2>artificial example2</xe:qualifier2>
            </rdf:Description>
        </xmp:BaseURL>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/",
        "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
        "xe": "http://ns.adobe.com/xmp-example/",
        "xmp": "http://ns.adobe.com/xap/1.0/"
    },
    "@id": "",
    "dc:source": {
        "rdf:value": "Adobe XMP Specification, April 2010",
        "xe:qualifier1": "artificial example1"
    },
    "xmp:BaseURL": {
        "rdf:value": {
            "@id": "https://www.adobe.com/"
        },
        "xe:qualifier2": "artificial example2"
    }
}
```

Resources generally do not have language qualifiers. However, if such language qualifiers are present, they shall be ignored by a processor. If other qualifiers are required, they are encoded as described above, and may also be ignored by a processor.

4.5.5.3 Array valued XMP property with qualifier

Array valued XMP property with qualifier shall be serialized as a key-value pair, where the key shall be the property name and its value shall be a JSON object. The content of the JSON object shall consist of two key-value pairs. The first pair shall be the serialization of the qualifier. The second pair shall have the key *rdf:value*. Its value shall be the serialization of array values as discussed in 4.5.5.2.

EXAMPLE Serialized XMP and its corresponding serialized JSON-LD for array valued XMP property with qualifier.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:xe="http://ns.adobe.com/xmp-example/">
    <rdf:Description rdf:about="">
        <xe:source rdf:parseType="Resource">
            <rdf:value>
                <rdf:Bag>
                    <rdf:li>XMP</rdf:li>
                    <rdf:li>metadata</rdf:li>
                </rdf:Bag>
            </rdf:value>
            <xe:qualifier>artificial example</xe:qualifier>
        </xe:source>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/",
        "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
        "xe": "http://ns.adobe.com/xmp-example/"
    },
    "@id": "",
    "xe:source": {
        "rdf:value": {
            "@set":["XMP","metadata"]
        },
        "xe:qualifier": "artificial example"
    }
}
```

4.5.5.4 Use of @language JSON-LD keyword

In JSON-LD, to allow the identification of content language, the JSON-LD keyword *@language* is used as a key. In accordance with IETF BCP 47, the value of *@language* shall be a language code. Serialization for the XMP property that also specifies language shall be done as a JSON object having two JSON-LD keywords, *@language* and *@value* as keys. The actual value of XMP property shall be given as a value of the key *@value*.

To specify the default language for a subtree, *@language* can be used in local context. Local context can be used to specify language of array and structure property. Also, local context can be used in places where simple properties have qualifiers on them, because *@language* cannot be used with *rdf:value*.

EXAMPLE Serialized XMP Packet in RDF/XML with xml:lang qualifier.

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:xmp="http://ns.adobe.com/xap/1.0/">
    <rdf:Description rdf:about="">
```

```
        <dc:source xml:lang="en-us">
            Adobe XMP Specification, April 2010
        </dc:source>
        <xmp:BaseURL rdf:resource="https://www.adobe.com/" xml:lang="en"/>
        <dc:subject xml:lang="en">
            <rdf:Bag>
                    <rdf:li>XMP</rdf:li>
                    <rdf:li>metadata</rdf:li>
                    <rdf:li>ISO standard</rdf:li>
                    <rdf:li xml:lang="fr">Norme internationale de l'ISO</rdf:li>
            </rdf:Bag>
        </dc:subject>
    </rdf:Description>
</rdf:RDF>
```

Serialized XMP Packet in JSON-LD with “@language” JSON-LD keyword

```
{
    "@context": {
        "dc": "http://purl.org/dc/elements/1.1/",
        "xmp": "http://ns.adobe.com/xap/1.0/"
    },
    "@id": "",
    "dc:source": {
        "@value": "Adobe XMP Specification, April 2010",
        "@language": "en-us"
    },
    "dc:subject": {
        "@set": [
            "XMP","metadata","ISO standard",
            {
                "@value": "Norme internationale de l'ISO",
                "@language": "fr"
            }
        ],
        "@context": {
            "@language": "en-us"
        }
    },
    "xmp:BaseURL": {
        "@id": "https://www.adobe.com/"
    }
}
```

Language associations shall only be applied to plain strings. Typed values or values that are subject to type coercion shall not be language tagged.

4.5.5.5 Language alternative array value

Language alternatives facilitate the selection of a simple text item based on a desired language. Each array item shall have an *xml:lang* qualifier(in RDF/XML serialization).

In JSON-LD, serialization of language alternative is the same as alternative array valued XMP forms with qualifiers; the only difference being that each item of the array has the JSON-LD keyword *@language* as its key. In the XML serialization, the default language is selected via *x-default*, but in JSON-LD, the default is *i-default*.

EXAMPLE Serialized XMP and its corresponding serialized JSON-LD for language alternative XMP property.

Serialized XMP Packet in RDF/XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
    <rdf:Description rdf:about="">
        <dc:title>
```