About the ESE v3.2 XML Schema

07/08/2009

Purpose

The ESE v3.2 XML Schema (http://www.europeana.eu/schemas/ese/ESE-V3.2.xsd) is the XML representation of the Europeana Semantic Elements (ESE) specifications v3.2 (http://version1.europeana.eu/c/document_library/get_file?uuid=c56f82a4-8191-42fa-9379-4d5ff8c4ff75&groupId=10602).
This schema can be used to validate XML instances of Data Sets to be submitted to Europeana.
The ESE v3.2 XML Schema extends the DC XML Schema[1] with the addition of elements belonging to the Europeana namespace.

Use

A typical XML instance file containing ESE metadata records has the following structure:

```
<metadata xmlns="http://www.europeana.eu/schemas/ese/"
          xmlns:europeana="http://www.europeana.eu/schemas/ese/"
          xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:dcterms="http://purl.org/dc/terms/">

    <record>

    <europeana:isShownBy>http://www.xx.yy/yy</europeana:isShownBy>
        <dc:subject>archéologie ; Grec ; Céramique</dc:subject>
        ...
    </record>
    <record>
        ...
    </record>
    ...
</metadata>
```

---

[1] The Dublin Core organisation provides XML Schemas definition to be used in applications based on Dublin Core (DC) elements (see: http://dublincore.org/schemas/xmls/). All DC elements in the provided XML Schema are based on the DC defined *SimpleLiteral* data type (which is based on the XML Schema *anyType* primitive data type, i.e. it can accept any values type). Moreover the DC XML Schema defines for all elements the optional attribute *xml:lang*. To be valid the DC elements in the XML instance documents must have a data type based on the *SimpleLiteral* DC data type (this occurs for instance in the dcterms XML Schema definition [http://dublincore.org/schemas/xmls/qdc/2008/02/11/dcterms.xsd])

Characteristics

Namespace URI: http://www.europeana.eu/schemas/ese/

In the XML schema and in XML instances this URI represents the namespace from which the elements defined by the schema come from.

Example of use in *XML schema*:

```
<xs:schema  xmlns:xs="http://www.w3.org/2001/XMLSchema"
            xmlns:dcterms="http://purl.org/dc/terms/"
            xmlns:dc="http://purl.org/dc/elements/1.1/"
            xmlns="http://www.europeana.eu/schemas/ese/"
            targetNamespace="http://www.europeana.eu/schemas/ese/"
            version="3.2"
            elementFormDefault="qualified"
            attributeFormDefault="unqualified">
```

Example of use in *XML instance*:

```
<metadata xmlns="http://www.europeana.eu/schemas/ese/"
          xmlns:europeana="http://www.europeana.eu/schemas/ese/"
          xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:dcterms="http://purl.org/dc/terms/">
```

Location URL: http://www.europeana.eu/schemas/ese/ESE-V3.2.xsd

URL of the location where the XML Schema file can be found.
In XML instance documents the *targetLocation* attribute provides hints to a processor regarding the location of schema documents.

Example of use in *XML instance*:

```
<metadata xmlns="http://www.europeana.eu/schemas/ese/"
          xmlns:europeana="http://www.europeana.eu/schemas/ese/"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://www.europeana.eu/schemas/ese/
http://www.europeana.eu/schemas/ese/ESE-V3.2.xsd"
          xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:dcterms="http://purl.org/dc/terms/">
```

Constraints

Specific constraints

The ESE v3.2 Schema doesn't accept the following elements (Europeana office is responsible for providing all these elements):

```
europeana:country
europeana:hasObject
europeana:language
europeana:uri
europeana:usertag
europeana:year
```

The values for the following elements must be valid URI:

```
europeana:isShownBy
europeana:isShownAt
europeana:object
```

The values for `europeana:type` element must be one of the following:

```
TEXT
IMAGE
SOUND
VIDEO
```

While DC elements can appear in a user preferred order, in the instances of documents, the Europena elements **must** appear after all DC elements and in the following order:

```
europeana:unstored
europeana:object
europeana:provider
europeana:type
europeana:isShownBy
europeana:isShownAt
```

The `europeana:isShownBy` and `europeana:isShownAt` elements may appear in any order (one of the two is mandatory) after the `europeana:type` element.

Constraints on Attributes

The ESE XML Schema extends the DC XML Schema by adding elements from the Europeana namespace. Currently the Europeana namespace does not define any attribute. Therefore the only attributes allowed by the XML Schema within the *<record>* element of the XML instance documents are the attributes *xml:lang* (defined by DC XML Schema) and *xsi:type*[2] of the DC elements. Moreover DC element including the *xsi:type* attribute must obey the DC type definition rules (i.e. must be derived by the DC SimpleLiteral data type).

Examples which are valid:

```
<dc:title xml:lang="en">My title</dc:title>
```
This is a valid combination of a DC element with the *xml:lang* attribute.

```
<dc:language xsi:type="dcterms:ISO639-2">en</dc:language>
```
This is a valid combination of a DC element with the *xsi:type* attribute. The value of the attribute is also valid because the dcterms vocabulary is based on the DC SimpleLiteral data type.

```
<dcterms:spatial xml:lang="en"
xsi:type="dcterms:TGN">Paris</dcterms:spatial>
```
This is a valid combination of a DC element with the *xml:lang* and the *xsi:type* attributes. The value of *xsi:type* is also valid because the dcterms vocabulary is based on the DC SimpleLiteral data type.

Examples which are not valid:

```
<europeana:unstored xml:lang="en">This is a rare object</
europeana:unstored>
```
This is *invalid* because a Europeana element cannot have any *xml:lang* attribute.

```
<dc:subject xsi:type="MyVoc">Paris</dc:subject>
```
This is *invalid* because the value of the attribute xsi:type is not based on DC SimpleLiteral data type.

```
<dcterms:temporal abc="MyVoc">Paris</dcterms:temporal>
```
This is *invalid* because "abc" is not an attribute allowed in a DC element.

---

[2] **xsi:type**. An element information item in an instance may explicitly assert its type using the attribute xsi:type. The value of this attribute is a *QName*. Where the type of an attribute information item in a document involved in validation is identified as QName, its actual value is composed of a [Definition:] *local name* and a [Definition:] *namespace name*. (e.g. dcterms: ISO639-2).