INTERNATIONAL STANDARD ISO/IEC 19757-8:2008
TECHNICAL CORRIGENDUM 1

Published 2011-12-01

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Document Schema Definition Languages (DSDL) —

Part 8:
Document Semantics Renaming Language (DSRL)

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Langages de définition de schéma de documents (DSDL) —

Partie 8: Langage pour renommer une sémantique de documents (DSRL)

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 19757-8:2008 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

_____

Page 1, Clause 2

In the final normative reference, replace the unique identifier "XSLT 2.0" with "XSLT".

Page 2, Clause 4.1, paragraph 2

Delete the Note following paragraph 2.

ICS 35.240.30

Ref. No. ISO/IEC 19757-8:2008/Cor.1:2011(E)

Published in Switzerland

Page 3, Clause 5, paragraph 3

At the end replace:

"has a target namespace"

with:

"conforms to XML Schema and has a target namespace".

Page 3, Clause 6.1, paragraph 1

In the Example following paragraph 1, replace:

"`added-attribute = attribute additional { xsd:boolean }`"

with:

"`added-attribute = attribute additional { ( "true" | "false" ) }`"

Pages 3–4, Clause 6.1, paragraph 3

Replace the entire text:

"The name of the element to be mapped is recorded in the content of the `dsrl:from` element. This content shall be a valid XML name, which can include a prefix that identifies the appropriate namespace. If the content is a prefixed name, it is recommended that the prefix used be declared in a namespace declaration that is declared as an attribute of the `dsrl:from` element."

with:

"The name of the element to be mapped is recorded in the content of the `dsrl:from` element. This content shall be a valid XML qualified name (QName) in accordance with XML-Names. If the content is a prefixed name, the namespace prefix must fall within the scope of a corresponding namespace declaration in accordance with XML-Names, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI. If the content is an unprefixed name and it falls within the scope of a corresponding default namespace declaration, the name shall be interpreted as the expanded name after prepending the default namespace IRI. If the content is an unprefixed name and it does not fall within the scope of a corresponding default namespace declaration, the name shall be interpreted as being in no namespace."

Page 4, Clause 6.1, paragraph 3

Delete the Note following paragraph 3.

Page 4, Clause 6.1, paragraph 4

Replace the entire text:

"If the named element needs to be replaced in different ways in different contexts the optional `dsrl:parent` element can be used to record XML Stylesheet Langauge Transformation (XSLT) patterns that distinguish between the different contexts in which replacement of the name is to be applied. No two `dsrl:element-map` elements shall have identical contents for both their `dsrl:parent` element and their `dsrl:from` element. If two or more maps with different contents match the same result path the last of matches shall be applied."

with:

> "If the named element needs to be replaced in different ways in different contexts the optional `dsrl:parent` element can be used to record patterns conforming to XSLT that distinguish between the different contexts in which replacement of the name is to be applied. No two `dsrl:element-map` elements shall have identical contents for both their `dsrl:parent` element and their `dsrl:from` element (interpreted as an expanded name, if it either contains a qualifying prefix or is unqualified and falls within the scope of a corresponding default namespace declaration). If two or more maps with different contents match the same result path the last of matches shall be applied."

Page 4, Clause 6.1, paragraph 4

Replace the entire text of the Note following paragraph 4:

> "NOTE   This precedence rule allows the XSLT error recovery rule for matching paths using the last template defined to be applied when processing DSRL using XSLT."

with:

> "NOTE   This precedence rule avoids a DSRL application having to treat a multiple map match as an error, as this could easily occur when merging multiple DSRL schemas."

Page 4, Clause 6.1, paragraphs 5 and 6

Replace the entire text:

> "The name to be applied to the mapped element when it is validated is recorded as the content of the `dsrl:to` element. This content shall be a valid XML name, which can include a prefix that identifies the appropriate namespace. If the content is a prefixed name, it is recommended that the prefix used be declared in a namespace declaration that is declared as an attribute of the `dsrl:to` element.
>
> If the element name is to stay the same, but one or more attributes of the element is to have its name or values mapped, the `dsrl:name` element can be used in place of the `dsrl:from` and `dsrl:to` pair. The content of a `dsrl:name` element shall be a valid XML name, which can include a prefix that identifies the appropriate namespace."

with:

> "The name to be applied to the mapped element when it is validated is recorded as the content of the `dsrl:to` element. This content shall be a valid XML name. If the content is a qualified name, the namespace prefix must fall within the scope of a corresponding namespace declaration in accordance with XML-Names, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI.
>
> If the element name is to stay the same, but one or more attributes of the element is to have its name or values mapped, the `dsrl:name` element can be used in place of the `dsrl:from` and `dsrl:to` pair. The content of a `dsrl:name` element shall be a valid XML qualified name (QName) in accordance with XML-Names. If the content is a qualified name, the namespace prefix must fall within the scope of a corresponding namespace declaration in accordance with XML-Names, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI. If the content is an unprefixed name and it falls within the scope of a corresponding default namespace declaration, the name shall be interpreted as the expanded name after prepending the default namespace IRI. If the content is an unprefixed name and it does not fall within the scope of a corresponding default namespace declaration, the name shall be interpreted as being in no namespace."

Page 4, Clause 6.1, paragraph 7

Replace the entire text:

"Names can be qualified providing the relevant namespace prefixes have been declared within the map. They cannot contain spaces, or any other character that is not a valid name character as defined in the W3C XML specification."

with the following Note:

"NOTE [USAGE] If the content of `dsrl:from`, `dsrl:to` or `dsrl:name` is a qualified name, it is recommended that the namespace declaration for any namespace qualifying prefix be included on the element in question. This is to reduce the risk of drafting errors, for example when cutting and pasting element maps using non-XML-aware text editors.

Page 4, Clause 6.1, Note following paragraph 9

Replace the text following the first example:

"If namespaces are used for the source or result element they must be declared as part of the definition, giving the declaration the form:"

with:

"If the element name map is mapping from or to a qualified name, it is recommended that the namespace declarations be included on the appropriate element(s). A typical example would be:"

Pages 4–5, Clause 6.1, paragraphs 11 and 12

Replace the entire text:

"Each `dsrl:attribute-map` replaces a single attribute. The name of the attribute to be mapped is recorded as the content of the `dsrl:from` element. No two `dsrl:attribute-map` elements within a given `dsrl:element-map` shall have the same value for their `dsrl:from` element.

If the attribute is to be directly mapped to an attribute in the result document, the name to be applied to the mapped attribute when it is validated is recorded as the contents of the immediately following `dsrl:to` element. If the `dsrl:to` element is empty the attribute named in the `dsrl:from` element is to be removed prior to validation."

with:

"Each `dsrl:attribute-map` replaces a single attribute. The name of the attribute to be mapped is recorded as the content of the `dsrl:from` element. The content of the `dsrl:from` element shall be a valid XML name. If the content is a qualified name, the namespace prefix must fall within the scope of a corresponding namespace declaration in accordance with XML-Names, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI. No two `dsrl:attribute-map` elements within a given `dsrl:elementmap` shall have the same value for their `dsrl:from` element (after expansion of any qualified names).

If the attribute is to be directly mapped to an attribute in the result document, the name to be applied to the mapped attribute when it is validated is recorded as the contents of the immediately following `dsrl:to` element.

If the `dsrl:to` element is non-empty, its content shall be a valid XML name. If the content is a qualified name, the namespace prefix must fall within the scope of a corresponding namespace

declaration in accordance with XMLNames, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI.

If the `dsrl:to` element is empty, the attribute named in the `dsrl:from` element is to be removed prior to validation."

Page 5, Clause 6.1, paragraph 14

At the end of the paragraph, insert the following two sentences:

"The content of a `dsrl:name` element shall be a valid XML name. If the content is a qualified name, the namespace prefix must fall within the scope of a corresponding namespace declaration in accordance with XML-Names, and the content shall be interpreted as the expanded name after replacement of the qualifying prefix by the namespace IRI."

Page 5, Clause 6.1, paragraph 17

Delete the paragraph:

"Attribute names can be qualified providing the relevant namespace prefixes have been declared within the map. They cannot contain spaces, or any other character that is not a valid name character as defined in the W3C XML specification."

Page 6, Clause 6.5, paragraph 1

Replace "can used" with "can be used" in the first line.

Pages 6–7, Clause 6.5, paragraphs 2 and 3

Replace the entire text:

"The default content for the element is recorded as the content of the `dsrl:default-content` element. Any contents assigned to the element within the document instance will be used during validation of the document, but if the element is empty, or not present, the default content will be applied during validation.

The `after` attribute of the `dsrl:default-content` element records the name of the target element after which the content has to be inserted if not found in the source instance. The element the default content is to be placed after shall be declared within the same DSRL map, even if it is not altered in any way. To ensure that the `after` attribute can be applied in the appropriate context the containing `dsrl:element-map` element shall start with a `dsrl:parent` statement that contains the name of the element within which the element identified by the `after` attribute occurs."

with:

"The default content for the element is recorded as the content of the `dsrl:default-content` element.

All `dsrl:element-map` elements that contain a `dsrl:default-content` element shall also contain a `dsrl:parent` element.

Each `dsrl:default-content` element shall contain an `after` attribute. The value of the `after` attribute of the `dsrl:default-content` element shall either be empty or specify the name of an element in the result document.

If the source instance contains no occurrence of the element with a parent element that matches the pattern specified by the corresponding `dsrl:parent` element, the `dsrl:default-content` element is ignored.

If the source instance contains no occurrences of an element for which default content is defined but contains at least one element that matches the pattern specified by the corresponding `dsrl:parent` element, the element with default content shall be inserted in the result instance. The precise point of insertion is determined by the value of the corresponding `after` attribute:

— If, for a given element map A, the value of the `after` attribute is empty and the first element that matches the pattern specified by the corresponding `dsrl:parent` element is the subject of its own element map B (i.e. its name corresponds to the content of a `dsrl:from` element in a distinct element map B), the element with default content shall be inserted as the first child of the element in the result instance that is the result of applying this element map B to the first element that matches the pattern specified by the corresponding `dsrl:parent` element;

— If, for a given element map, the value of the `after` attribute is empty and the first element that matches the pattern specified by the corresponding `dsrl:parent` element is *not* the subject of its own element map (i.e. its name does *not* correspond to the content of any `dsrl:from` element in any other element map), the element with default content shall be inserted as the first child of the element in the result instance that corresponds to the first element in the source instance that matches the pattern specified in the `dsrl:parent` element;

— If, for a given element map, the value of the `after` attribute specifies the name of an element in the result document that is a child of an element matching the pattern specified by the corresponding `dsrl:parent` element, the element with default content shall be inserted immediately following the first occurrence of a matching element that contains the element specified by the `after` attribute.

If the source instance contains an element for which default content is defined, and this element is empty, the default content shall be inserted into this element in the result document.

If the source instance contains an element for which default content is defined, and this element already has content, the default content shall be ignored.

It is an error for the `after` attribute to specify the name of an element in the result document that does not occur as a child of an element in the result document that matches the pattern specified by the corresponding `dsrl:parent` element.

It is an error for the name specified by the `after` attribute to be the same as the name specified by the corresponding `dsrl:to` element (after expansion in each case of any qualifying prefix) in the same element map."

Page 7, Clause 6.6, paragraph 1

Replace the text:

"users of ISO/IEC 19757-2 can create a mapping rule"

with:

"a mapping can be declared".

Page 8, Clause 6.7, paragraph 2

Delete the Note following paragraph 2.

Page 8, Clause 6.7, paragraph 5

Insert after the Note following paragraph 5:

> "If a processor uses the entity name mapping feature specified in this Standard, depending upon its processing model it may not be able to map names found in the replacement text of entities whose names are mapped. However, such behaviour is implementation-dependent and it is strongly recommended that processors should always find a way of mapping names found in entity replacement text."

Page 8, Clause 7, paragraph 1

Delete the Note following paragraph 1.

Page 8, Clause 7, paragraph 2

Replace the second line of the schema fragment:

> "`define replacement-text = element dsrl:replacement-text {any-content}`"

with:

> "`define replacement-text = element dsrl:replacement-text {text}`"

Page 8, Clause 7, paragraph 4

Replace the entire text:

> "The replacement text for any references made to the named entity is recorded as the contents of the immediately following `dsrl:replacement-text` element. If the replacement text contains markup it shall be defined as a CDATA marked section. The replacement text shall be well-formed XML. The replacement text shall not include entity references to entities other than the five pre-defined attributes recognized by XML, but can contain character references."

with:

> "The replacement text for any references made to the named entity is specified by the contents of the immediately following `dsrl:replacement-text` element. The replacement text shall not include any text recognised as markup other than the five pre-defined entities recognized by XML and character references."

Page 9, Clause 7, Note following paragraph 4

Replace the entire example:

```
" <dsrl:define-entity>
  <dsrl:from>e</dsrl:from>
  <dsrl:replacement-text>&#233;</dsrl:replacement-text>
  <dsrl:from>ISO</dsrl:from>
  <dsrl:replacement-text>
   International Organization for Standardization
  </dsrl:replacement-text>
 </dsrl:define-entity>
 <dsrl:define-entity>
  <dsrl:from>XML</dsrl:from>
  <dsrl:replacement-text>
```

```
   <![CDATA[the W3C Extensible Markup Language (<acronym>XML</acronym>)]]>
  </dsrl:replacement-text>
 </dsrl:define-entity>"
```

with:

```
" <dsrl:define-entity>
  <dsrl:from>e</dsrl:from>
  <dsrl:replacement-text>&#233;</dsrl:replacement-text>
 </dsrl:define-entity>
 <dsrl:define-entity>
  <dsrl:from>ISO</dsrl:from>
  <dsrl:replacement-text>
   International Organization for Standardization
  </dsrl:replacement-text>
 </dsrl:define-entity>"
```

Page 9, Clause 8, paragraph 1

Replace "two" with "three" in the first line and insert the following item at the end of the list:

> "3.   Conformance without entity-mapping or entity-definition"

Page 9, Clause 8, paragraph 3

Replace the entire text:

> "Applications that do not support entity-mapping conformance shall support entity-definition conformance. This means that as well as being able to support all the compulsory components of a `dsrl:maps` element they shall also support the optional `dsrl:define-entity` methodology for defining entity declarations to be used in the result document of a DSRL map."

with:

> "Applications that support entity-definition conformance shall support all features of ISO/IEC 19757-8 except entity-mapping. This means that, as well as being able to support all the compulsory components of a `dsrl:maps` element, they shall also support the optional `dsrl:define-entity` methodology for defining entity declarations to be used in the result document of a DSRL map.
>
> Applications that support conformance without entity-mapping or entity-definition shall support all features of ISO/IEC 19757-8 except entity-mapping and entity-definition, as defined in Clause 6.1 and Clause 7.
>
> Applications shall ignore the features that are not included in their conformance class."

Clause A.1, Page 11, line 26

Replace the text:

> " `<data type="boolean"/>` "

with:

```
" <choice>
   <value>true</value>
   <value>false</value>
 </choice>"
```

Clause A.1, Page 12, lines 21 to 23

Delete these lines:

```
"<optional>
  <ref name="force-default"/>
 </optional>"
```

Clause A.1, Page 12, lines 27 to 31

Delete these lines:

```
"<define name="force-default">
  <attribute name="force-default">
   <data type="boolean"/>
  </attribute>
 </define>"
```

Clause A.1, Page 12, lines 52 to 56

Replace the text:

```
"<define name="replacement-text">
  <element name="dsrl:replacement-text">
   <ref name="any-content"/>
  </element>
 </define>"
```

with:

```
"<define name="replacement-text">
  <element name="dsrl:replacement-text">
   <text/>
  </element>
 </define>"
```

Clause A.1, Page 13, lines 20 to 22

Delete these lines:

```
"<optional>
  <ref name="force-default"/>
 </optional>"
```

Page 15, Clause A.2, line 19

Replace the text:

```
"added-attribute = attribute additional {xsd:boolean}"
```

with:

```
"added-attribute = attribute additional { ( "true" | "false" ) }"
```

Page 15, Clause A.2, line 28

Replace the text:

```
"default-value= element dsrl:default-value {force-default?, text }"
```

with:

```
"default-value= element dsrl:default-value { text }"
```

Page 15, Clause A.2, line 29

Delete this line:

```
"force-default = attribute force-default {xsd:boolean}"
```

Page 15, Clause A.2, line 33

Replace the text:

```
"replacement-text = element dsrl:replacement-text {any-content}"
```

with:

```
"replacement-text = element dsrl:replacement-text {text}"
```

Page 15, Clause A.2, line 37

Replace the text:

```
"default-content = element dsrl:default-content {force-default?, after,
any-content}"
```

with

```
"default-content = element dsrl:default-content {after, any-content}"
```

Page 19, Annex B, lines 13 to 17

Delete these lines:

```
" <dsrl:define-entity>
  <dsrl:from>xml</dsrl:from>
  <dsrl:replacement-text><![CDATA[Extensible Markup Language
  (<acronym>XML</acronym>) ]]></dsrl:replacement-text>
 </dsrl:define-entity>"
```

Page 19, Annex B, paragraph 2 (following the first Note)

Replace "XSLT 2.0" with "XSLT" and delete the final sentence:

"A tutorial explaining this process, together with the XSLT code required, is available online at http://www.dsdl.org."

Page 19, Annex B, paragraph 3

Replace the list:

1. Parse the DSRL map to see if it contains any of the optional `dsrl:entity-name-map` or `dsrl:define-entity` element. If it does:

   — generate an XML Document Type Definition containing the revised definitions of the entities to be mapped

   — add a reference to the revised definition of the entities to the document instance to be processed

2. Generate an XSLT transformation that changes the names and, where appropriate, namespace prefixes, assigned to elements, attributes, entities or processing instructions. This transformation shall include templates that:

   — map any attribute or element content that is declared in a value map

   — assign default content to any missing attributes or elements

   — convert attribute values to element content where this has been requested

   — serialize the amended information set as an XML document.

3. Parse the transformed document against the target schema/DTD.

with:

1. Parse the DSRL map to see if it contains any of the optional `dsrl:entity-name-map` or `dsrl:define-entity` element. If it does:

   — generate an XML Document Type Definition containing the revised definitions of the entities to be mapped

   — add a reference to the revised definition of the entities to the document instance to be processed

   — replace each entity reference, whose name corresponds to the content of a `dsrl:from` element in a `dsrl:entity-name-map` element, to a processing instruction whose name and property don't coincide with the name and property of any other processing instruction in the document and which unambiguously represent the original entity reference.

2. Generate an XSLT transformation that changes the names and, where appropriate, namespace prefixes, assigned to elements, attributes or processing instructions. This transformation shall include templates that:

   — map any attribute or element content that is declared in a value map

   — assign default content to any missing attributes or elements

   — convert attribute values to element content where this has been requested

   — serialize the amended information set as an XML document.

3. Replace each of the processing instructions, which represent those original entity references that are to have their names transformed, with entity references using the names in the corresponding `dsrl:to` elements in the `dsrl:entity-name-map` element.

4. Parse the transformed document against the target schema/DTD.

Page 20, Bibliography

Delete reference [1] and re-number the remaining reference.