

# Custom XML Defect Reports

---

## **DR 09-0191 — Custom XML Schema: Allowed attribute values**

Agreed.

The following changes will be made to both transitional and strict:

- Create a new simple type, ST\_Uri, with restriction set to xs:anyURI.
- The attributes schemaLocation (schema @schemaLocation (23.2.1)) and manifestLocation (schema @manifestLocation (23.2.1)) will be restricted to ST\_Uri instead of ST\_String

In summary, this change affects the following elements

- schema @schemaLocation (23.2.1)
- schema @manifestLocation (23.2.1)

## **DR 09-0207 – WML: Custom XML and Smart Tags**

See attached document for revisions made to section 17.5 and 17.5.1.

## **DR 09-0208 – WML: Custom XML and Smart Tags**

See attached document for revisions made to section 17.5 and 17.5.1.

## **DR 09-0209 – WML: Custom XML and Smart Tags**

No change is needed in the text to describe behavior of missing uri attribute. The description of how the SmartTag element behaves in the cases where the attributes are missing is described properly in Part 1, 17.5.1.9, “SmartTag (Inline-Level Smart Tag).” The standard states that if uri is omitted then “uri shall be assumed to be null (no associated URI)”

For rewording of 17.5.1 see attached document for revisions made to section 17.5 and 17.5.1.

## **DR 09-0210 – WML: Custom XML and Smart Tags**

See attached document for revisions made to section 17.5 and 17.5.1.

## **DR 09-0211 – WML: Custom XML and Smart Tags**

See attached document for revisions made to section 17.5 and 17.5.1.

## DR 09-0212 – WML: Custom XML and Smart Tags

Agreed.

The following changes will be made to both transitional and strict:

- Create a new simple type, ST\_XmlName, with restriction set to xs:NCName
- The attributes name (attr @name (17.5.1.1) and attr @name (17.5.1.2)) and element (customXml @element (17.5.1.3), customXml @element (17.5.1.4), customXml @element (17.5.1.5), customXml @element (17.5.1.6), and smartTag @element(17.5.1.9)) will be restricted to ST\_XmlName instead of ST\_String

In summary, this change affects the following elements:

- attr @name (17.5.1.1)
- attr @name (17.5.1.2)
- customXml @element (17.5.1.3)
- customXml @element (17.5.1.4)
- customXml @element (17.5.1.5)
- customXml @element (17.5.1.6)
- smartTag @element(17.5.1.9)

The following changes will be made to both transitional and strict:

- Create a new simple type, ST\_Uri, with restriction set to xs:anyURI.
- The attribute uri (attr @uri (17.5.1.1), attr @uri (17.5.1.2), customXml @uri (17.5.1.3), customXml @uri (17.5.1.4), customXml @uri (17.5.1.5), customXml @uri (17.5.1.6), smartTag @uri(17.5.1.9) , and schema @uri (23.2.1)) will be restricted to ST\_Uri instead of ST\_String

In summary, this change affects the following elements:

- attr @uri (17.5.1.1)
- attr @uri (17.5.1.2)
- customXml @uri (17.5.1.3)
- customXml @uri (17.5.1.4)
- customXml @uri (17.5.1.5)
- customXml @uri (17.5.1.6)
- smartTag @uri(17.5.1.9)
- schema @uri (23.2.1)

## DR 09-0213 – WML: Custom XML and Smart Tags

See attached document for revisions made to section 17.5 and 17.5.1.

## DR 09-0028 – WML, Custom XML and Smart Tags: Tighten data types

Same proposal as DR 09-0212.

Agreed.

The following changes will be made to both transitional and strict:

- Create a new simple type, ST\_XmlName, with restriction set to xs:NCName
- The attributes name (attr @name (17.5.1.1) and attr @name (17.5.1.2)) and element (customXml @element (17.5.1.3), customXml @element (17.5.1.4), customXml @element (17.5.1.5), customXml @element (17.5.1.6), and smartTag @element(17.5.1.9)) will be restricted to ST\_XmlName instead of ST\_String

In summary, this change affects the following elements:

- attr @name (17.5.1.1)
- attr @name (17.5.1.2)
- customXml @element (17.5.1.3)
- customXml @element (17.5.1.4)
- customXml @element (17.5.1.5)
- customXml @element (17.5.1.6)
- smartTag @element(17.5.1.9)

The following changes will be made to both transitional and strict:

- Create a new simple type, ST\_Uri, with restriction set to xs:anyURI.
- The attribute uri (attr @uri (17.5.1.1), attr @uri (17.5.1.2), customXml @uri (17.5.1.3), customXml @uri (17.5.1.4), customXml @uri (17.5.1.5), customXml @uri (17.5.1.6), smartTag @uri(17.5.1.9) , and schema @uri (23.2.1)) will be restricted to ST\_Uri instead of ST\_String

In summary, this change affects the following elements:

- attr @uri (17.5.1.1)
- attr @uri (17.5.1.2)
- customXml @uri (17.5.1.3)
- customXml @uri (17.5.1.4)
- customXml @uri (17.5.1.5)
- customXml @uri (17.5.1.6)
- smartTag @uri(17.5.1.9)
- schema @uri (23.2.1)

## DR 09-0221 – WNL: Custom XML Attribute

Section 17.5.1.1 shall be updated as follows:

### 17.5.1.1 attr (Custom XML Attribute)

This element specifies a custom XML attribute which shall be located on the parent custom XML element specified via the customXml element (§17.5.1.4;§17.5.1.5;§17.5.1.3; §17.5.1.6). The attributes on this element shall be used to specify the contents of the custom XML attribute.

A document that contains two or more attr elements with identical name and uri attribute values, including attr elements with identical name attribute values and an empty or omitted uri attribute, shall be considered non-conformant.

[*Example:* Consider a custom XML element with the following properties:

```
<w:customXmlPr>  
<w:attr w:name="companyName" ... />  
<w:attr w:name="companySymbol" ... />  
</w:customXmlPr>
```

This property bag specifies that the parent custom XML element must have two attributes associated with it, the first with a name of companyName, and the second with a name of companySymbol. *end example*]

## DR 09-0249 – Custom XML markup: well-formedness and validation

See attached document for revisions made to section 17.5 and 17.5.1. A new paragraph has been added to 17.5.1 regarding this issue.

It is a consumer defined operation to extract XML documents from custom XML markup. The result of extracting XML from custom XML markup shall always return well-formed XML. In addition, custom XML markup with Open XML, if added according to the specification, shall never create a non-conformant Open XML document. However, the result of extracting XML from custom XML markup may return an invalid document according to the custom XML schema.

[*Guidance:* The uri attribute on custom XML elements or attributes can be used to specify XML prefixes when extracting XML documents from custom XML markup. If the uri attribute is specified for custom XML elements or attributes then a consumer can choose to add a prefix to the output XML elements or attributes based on the value of the uri attribute. If the uri attribute is omitted for custom XML elements or attributes then a consumer can choose to not write a prefix to the output XML elements or attributes. *End guidance*]