**Proposals for Harmonizing the EDXL Family – 02-12-2018**:

1. **ADD “CommentsType” to new wd: edxl-ct-v1.0-wd08.xsd from EDXL-HAVE-v2.0 to complement “RemarksType” in edxl-ct-v1.0-wd06.xsd from. EDXL-SitRep-v1.0**: to support the elements ‘comment’ and ‘remarks’ in EDXL specifications. We have a possible conflict that the EM RIM SC should resolve before EDXL-HAVE-v2.0 or EDXL-SitRep-v1.0 proceed further. This issue is that we have two differently named XML elements which serve nearly the same purpose. Based on our last discussion Jeff suggested that the following was what I was actually proposing. I’m showing how it would look in a new wd for edxl-ct-v1.0-wd08 following ‘RemarksType’ which is already included:

<xs:simpleType name="RemarksType">

 <xs:restriction base="xs:string"/>

</xs:simpleType>

<xs:simpleType name="CommentsType">

 <xs:restriction base="xs:string">

 <xs:whiteSpace value="preserve"/>

 <xs:maxLength value="1023"/>

 <xs:minLength value="1"/>

 </xs:restriction>

</xs:simpleType>

However, EDXL-HAVE-v2.0 uses a combination of specially defined strings to achieve the same result for the element <comment> in context as a subclass of other complexTypes to which these numerous individual instances of <comment> belong:

<xs:element name="comment" type="FreeTextType" minOccurs="0" maxOccurs="1">

 <xs:annotation>

 <xs:documentation>General comment/summary on the trauma center status</xs:documentation>

 </xs:annotation>

</xs:element>

If we add ‘CommentsType’ then future use would be as follows,

<xs:element name="comment" type="ct:CommentsType" minOccurs="0" maxOccurs="1">

**but** the question is, should we do this for EDXL-HAVE-v2.0 at the same time as we introduce a new Committee Specification for EDXL-CT-v1.0?

 In EDXL-HAVE-v2.0.this element is constrained by limitations of the type="FreeTextType" (see below) and we should give some thought to including FreeTextType, AlternateTextType and LimitedString (as a Type) from HAVE 2.0 in edxl-ct-v1.0-wd08 to cover multiple languages and retain the structure.

<xs:complexType name="FreeTextType">

 <xs:sequence>

 <xs:element name="defaultText" type="LimitedString">

 <xs:annotation>

 <xs:documentation>The text value that uses the message default language (defined at in the HAVE message defaultLanguage attribute).</xs:documentation>

 </xs:annotation>

 </xs:element>

 <xs:element name="alternateText" type="AlternateTextType" minOccurs="0" maxOccurs="unbounded">

 <xs:annotation>

 <xs:documentation>Alternate language representation.</xs:documentation>

 </xs:annotation>

 </xs:element>

 </xs:sequence>

</xs:complexType>

<xs:complexType name="AlternateTextType">

 <xs:simpleContent>

 <xs:extension base="LimitedString">

 <xs:attribute name="language" type="xs:string" use="required">

 <xs:annotation>

 <xs:documentation>Language code for the text in this element. Code MUST comply with RFC3066. </xs:documentation>

 </xs:annotation>

 </xs:attribute>

 </xs:extension>

 </xs:simpleContent>

</xs:complexType>

This type, in turn, is constrained by the restriction of “Limited String”:

<xs:simpleType name="LimitedString">

 <xs:annotation>

 <xs:documentation>Text block for preserving whitespace but limiting length to 1024 characters.</xs:documentation>

 </xs:annotation>

 <xs:restriction base="xs:string">

 <xs:whiteSpace value="preserve"/>

 <xs:maxLength value="1024"/>

 </xs:restriction>

</xs:simpleType>

1. **Delete ‘xlink-2003-12-31.xsd’ in favor of ‘xlink.xsd’ in all instances.**:
2. **Replace edxl\_xPIL.xsd with edxl-xPIL.xsd in all imports and replace edxl\_xAL.xsd with edxl-xAL.xsd in all imports**