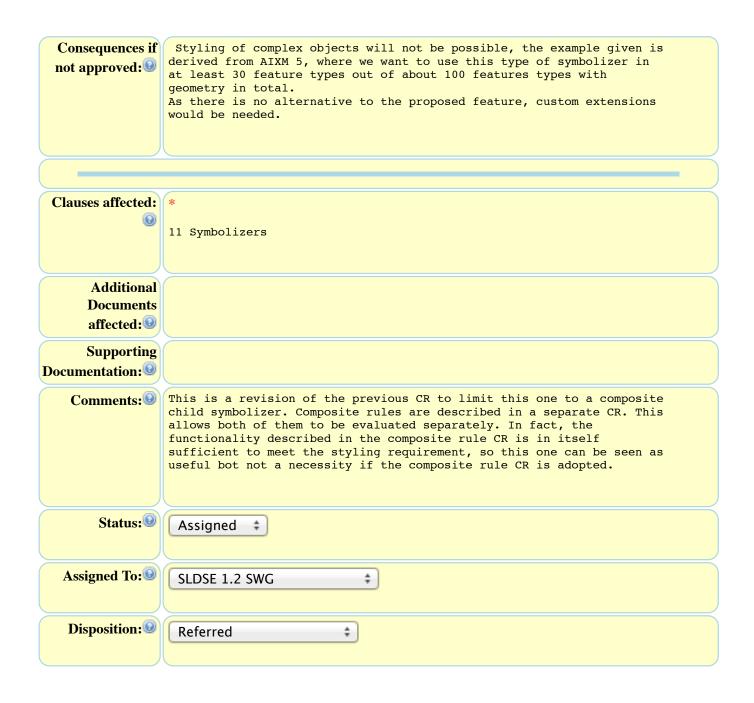
All Fields marked with * are mandatory.

Change Request #:	176
Assigned OGC Document #:	11-148
Name:	*Jeroen Dries
Organization:	*Luciad NV
Email:	*jeroen.dries@luciad.com
Document Name/Version:	*Symbology Encoding Implementation Specification / 1.1.0
OGC Project Document:	*05-077r4
If this is a revision of a previous submission and you have a Change Request Number, then check here: Enter the CR number here: 153 Enter the Revsion Number that you are revising here:	
Title:	*Symbolizer for styling of nested child objects
Source:	*OWS-8 Aviation
Work item code:	
Category:	* B (Addition of feature) ‡
Reason for change:	Complex data models like AIXM 5 have feature types that are a composition of nested objects. The current SE 1.1 only allows limited styling of these child objects, such as using the same style for all objects. This change request defines a new type of symbolizer that makes this possible.
Summary of change:	* Add a new symbolizer: 'CompositeChildSymbolizer'. It has a mandatory

1 of 3

```
element that points to a list of nested child objects, and a list of
symbolizers that will be applied to each object in the list. For
instance, for a
GML feature that looks like:
<Parent>
<ParentColor>Black</ParentColor>
<Color>Blue</Color>
<ChildGeometry> ...</ChildGeometry>
</Child>
<Child>
<Color>Red</Color>
<ChildGeometry> ...</ChildGeometry>
<ChildGeometry> ...</ChildGeometry>
</Child>
<Child>
<ChildGeometry> ...</ChildGeometry>
</Child>
<Parent>
A composite child symbolizer would look like this:
<se:CompositeChildSymbolizer>
<se:childProperty>Child</se:childProperty>
<se:PolygonSymbolizer>
<se:Geometry><oqc:PropertyName>ChildGeometry</oqc:PropertyName></se:Geometry>
<se:Stroke>
<se:SvqParameter
name="stroke"><ogc:PropertyName>Color</ogc:PropertyName></se:SvgParameter>
<se:SvqParameter
name="stroke-width">4</se:SvgParameter>
</se:Stroke>
</se:PolygonSymbolizer>
</se:CompositeChildSymbolizer>
Applications that support this symbolizer will also need to support
the XPath parent axes, or '../' in short to reference the parent of
a child node from within the composite child rule or symbolizer.
The XML schema for these types would look like:
<xsd:element name="CompositeChildSymbolizer"</pre>
type="se:CompositeChildSymbolizerType"
substitutionGroup="se:Symbolizer">
<xsd:annotation>
<xsd:documentation>A
"CompositeChildSymbolizer" comprises a group of
Symbolizers that are to be applied to a list of
objects</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:complexType</pre>
name="CompositeChildSymbolizerType">
<xsd:complexContent>
<xsd:extension base="se:SymbolizerType">
<xsd:sequence>
<xsd:element ref="fes:ValueReference"/>
<xsd:element ref="se:Symbolizer"</pre>
maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

2 of 3



3 of 3