Styling Specification for ncWMS (Environmental Data Abstraction Library (edal-java) mercurial tag “SLD/SE Parser v1.0” changeset f42190a32c19)

27 March 2014

C Roberts
Document Control

Contributors

<table>
<thead>
<tr>
<th>Person</th>
<th>Role</th>
<th>Organisation</th>
<th>Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>C Roberts</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Document Approval

<table>
<thead>
<tr>
<th>Person</th>
<th>Role</th>
<th>Organisation</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Distribution

<table>
<thead>
<tr>
<th>Person</th>
<th>Role</th>
<th>Organisation</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

References

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Document</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Revision History

<table>
<thead>
<tr>
<th>Issue</th>
<th>Author</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
# Table of Contents

1 Introduction.............................................................................................................................................. 5

2 Description of XML tags.......................................................................................................................... 5

2.1 General principles.................................................................................................................................. 5

2.2 Functions.............................................................................................................................................. 7

2.2.1 Categorize....................................................................................................................................... 8

2.2.2 Categorize2D.................................................................................................................................. 8

2.2.3 Interpolate...................................................................................................................................... 9

2.2.4 Segment....................................................................................................................................... 10

2.3 Colour maps........................................................................................................................................ 12

2.3.1 Common elements............................................................................................................................ 12

2.3.2 Thresholds (SE compliant)............................................................................................................ 12

2.3.3 Interpolation (SE compliant)......................................................................................................... 13

2.3.4 Segment and named palettes......................................................................................................... 13

2.4 Bivariate colour maps......................................................................................................................... 14

2.4.1 Common elements............................................................................................................................ 14

2.5 Glyphs............................................................................................................................................... 15

2.5.1 In situ coloured glyphs.................................................................................................................. 15

2.6 Arrows............................................................................................................................................... 16

2.7 Confidence interval triangles............................................................................................................. 17

2.8 Contours............................................................................................................................................ 18

2.9 Patterns............................................................................................................................................. 19

2.9.1 Stippling....................................................................................................................................... 19

2.10 Opacity transforms.............................................................................................................................. 20

2.10.1 Flat............................................................................................................................................. 20

2.10.2 Opacity map................................................................................................................................ 20

3 Examples of usage of XML tags.............................................................................................................. 20

3.1 Thresholds and interpolation.............................................................................................................. 20

3.2 Palette............................................................................................................................................... 23

3.3 Bivariate colour map........................................................................................................................... 25

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
3.4 In situ coloured glyphs ................................................................................................................. 29
3.5 Arrows ............................................................................................................................................. 30
3.6 Confidence interval triangles ........................................................................................................... 32
3.7 Contours .......................................................................................................................................... 36
3.8 Stippling .......................................................................................................................................... 38
3.9 Linear opacity transform .................................................................................................................. 41
1 Introduction

Information for specifying the style of image layers generated using a web map service (WMS) may be specified as an XML document using the existing standards of styled layer descriptor (SLD) and symbology encoding (SE) defined by the Open Geospatial Consortium (OGC). Those parts of the standard specific to a WMS are contained in SLD and those parts more generally applicable to specifying symbology within SE. The latter only supports representing raster data by colour maps defined by the following methods:

1. Thresholds – a list of thresholds and values representing colours.
2. Interpolation – a list of interpolation points.

We wish to stay within the SLD framework as it is the existing standard, but need to support more styling methods in order to represent quantitative uncertainty and other kinds of data e.g. vectors.

2 Description of XML tags

2.1 General principles

Each SLD document contains a number of layers, which correspond to a layer in the output image that results when the document is parsed. Each layer has an associated style. The latter defines how the data is plotted in the image layer. Each layer can either be named or a user layer. In the former case the name of the layer is the unique, machine-readable Name of the layer in the WMS server (which usually maps to a variable within an underlying NetCDF file in ncWMS). More than one variable can be plotted within the same layer, for example as a bivariate colour map, by specifying coverage constraints within a user layer, which may be named arbitrarily.

An outline for the general structure of an SLD document containing named layers is as follows:

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:se="http://www.opengis.net/se"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:resc="http://www.resc.reading.ac.uk">
  <NamedLayer>
    <se:Name>OSTIA/analysed_sst</se:Name>
    <UserStyle>
      <se:Name>sea_surface_temperature_style</se:Name>
      <se:CoverageStyle>
        <se:Rule>
          <se:RasterSymbolizer>
            <se:Opacity>1.0</se:Opacity>
          </se:RasterSymbolizer>
        </se:Rule>
      </se:CoverageStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
The root element of the document is an <sld:StyledLayerDescriptor> tag. Within this are contained one or more <sld:NamedLayer> elements, which specify how to plot the layers of the image. Our convention is that the layers are plotted in order from the top to the bottom of the document. In the outline above the sea surface temperature would be plotted first and then the error would be plotted on top of it.

Each layer element contains an <se:Name> tag specifying the variable to be plotted and a <se:UserStyle> tag, which specifies the styling information for the layer. The style can also have a name, which may be human readable and is ignored by the parser. In the case of a two or more variables being plotted in the same layer a user layer can be specified, which does not require a name. The variables to be plotted are then determined by coverage constraints. The general structure is as in the following example:

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
Ref: MonWMS_Styling_Specification_1.0
```
Each style element must contain one <se:CoverageStyle> element containing one <se:Rule> element. The latter must contain one symbolizer element. SE provides the <se:RasterSymbolizer> element for specifying the style of raster data. It contains a function, which in SE can either be the categorize or interpolate function. We support all the functionality of <se:RasterSymbolizer> and additional functionality. Most of this is provided by additional types of symbolizer element. There is also an extension to the <se:RasterSymbolizer> tag with the addition of the segment function.

The next section describes the types of functions that we support. The following sections describe the different types of symbolizers. In the tables of XML tags if an element has a default value it is optional otherwise it must be specified. Colours are encoded by 3 or 4 byte hexadecimal strings. The last 3 bytes specify the red, green and blue values in that order. In the case of a 4 byte string, the first byte specifies the opacity. For example #000000 and #FF000000 both specify opaque black. Finally, each symbolizer may contain one opacity transform, either flat or an opacity map based on a function. Opacity transforms are discussed in their own section at the end. Currently opacity transforms can only be applied to a layer not the whole image.

2.2 Functions

SE defines two functions that are used in the <se:RasterSymbolizer> tag: <se:Categorize> and <se:Interpolate>. We have created additional functions. So far these are <resc:Categorize2D> for bivariate colour maps and <resc:Segment>, which creates equally spaced bands, which are either
interpolated or subsampled as necessary. All functions must have an attribute named fallbackValue, which describes the value to use if there is no data present. Where a 1D function is expected any of the tree functions Categorize, Interpolate or Segment may be used interchangeably. This currently applies to colour maps, stippling and opacity transforms.

2.2.1 Categorize
The categorize function divides the data for the variable being plotted up into sections based on numerical thresholds, above or below which a different value of either the colour, the density of a pattern or the opacity is plotted. The \(<\text{se:Categorize}>\) function can contain the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;\text{se:LookupValue}&gt;)</td>
<td>String</td>
<td>When used within (&lt;\text{se:RasterSymbolizer}&gt;) it must contain the text “Rasterdata” for SE compliance. Must be specified within opacity maps.</td>
<td>Specifies the raster data source to use within an opacity map, otherwise ignored.</td>
</tr>
<tr>
<td>(&lt;\text{se:Threshold}&gt;)</td>
<td>Floating point number</td>
<td>One or more must be specified</td>
<td>The thresholds within the data</td>
</tr>
<tr>
<td>(&lt;\text{se:Value}&gt;)</td>
<td>String specifying a colour or a floating point number specifying a pattern density or opacity.</td>
<td>One more than the number of thresholds must be specified</td>
<td>The value to plot before or after the corresponding threshold(s) within the data.</td>
</tr>
</tbody>
</table>

This list of thresholds and values must start and end with a value and alternates between a threshold and a value to be compliant with SE. For an example see the section on the use of thresholds with colour maps.

2.2.2 Categorize2D
The categorize2D function is similar to the categorize function, but takes two variables as input and is applicable to bivariate colour maps. The \(<\text{se:Categorize2D}>\) function can contain the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;\text{resc:XThreshold}&gt;)</td>
<td>Floating point number</td>
<td>1..n must be specified</td>
<td>List of thresholds applicable to x variable</td>
</tr>
</tbody>
</table>

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
For an example see the section on bivariate colour maps.

### 2.2.3 Interpolate
The interpolate function interpolates the value to be plotted linearly between a series of points in the range of the variable where the value to be plotted is specified explicitly. Within the `<se:Interpolate>` function the following tags can be used:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;se:LookupValue&gt;</code></td>
<td>String</td>
<td>When used within <code>&lt;se:RasterSymbolizer&gt;</code> it must contain the text “Rasterdata” for SE compliance. Must be specified within opacity maps.</td>
<td>Within an opacity map specifies the raster data source(s) to use, otherwise ignored.</td>
</tr>
<tr>
<td><code>&lt;se:InterpolationPoint&gt;</code></td>
<td>XML element</td>
<td>At list of at least two must be specified</td>
<td>Contains the value and data point of an interpolation point</td>
</tr>
</tbody>
</table>

The `<se:InterpolationPoint>` must contain the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;se:Data&gt;</code></td>
<td>Floating point value</td>
<td>Must be specified</td>
<td>The data point of the interpolation point</td>
</tr>
<tr>
<td>Tag name</td>
<td>Type of contents</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>&lt;se:Value&gt;</code></td>
<td>String specifying a colour or a floating point number specifying a pattern density or opacity.</td>
<td>Must be specified</td>
<td>The value to be plotted at the interpolation point</td>
</tr>
</tbody>
</table>

For an example see the section on colour maps.

### 2.2.4 Segment

The segment function divides the data for the variable to be plotted into uniform bands of different value given either a list of values or the name of a colour palette, the number of bands and a range for the data. If the number of bands is greater than or less than the number of values then these are either interpolated uniformly or subsampled to give the specified number of bands. Up to 250 bands can be specified, which has an effect near to interpolation. The `<resc:Interpolate>` tag contains the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;se:LookupValue&gt;</code></td>
<td>String</td>
<td>Must be used within opacity maps, otherwise ignored.</td>
<td>Within an opacity map specifies the raster data source(s) to use, otherwise ignored.</td>
</tr>
<tr>
<td><code>&lt;resc:ValueList&gt;</code></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Specifies the values to interpolate between or subsample.</td>
</tr>
<tr>
<td><code>&lt;resc:BelowMinValue&gt;</code></td>
<td>String specifying a colour or a floating point number specifying a pattern density or opacity.</td>
<td>If omitted the same value is used as at the minimum in the range.</td>
<td>Specifies the value to plot below the minimum point in the range.</td>
</tr>
<tr>
<td><code>&lt;resc:AboveMaxValue&gt;</code></td>
<td>String specifying a colour or a floating point number specifying a pattern density or opacity.</td>
<td>If omitted the same value as at the maximum of the range is used.</td>
<td>Specifies the value to plot above the maximum point in the range.</td>
</tr>
<tr>
<td>Tag name</td>
<td>Type of contents</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td><a href="">resc:NumberOfSegments</a></td>
<td>Integer</td>
<td>Must be specified</td>
<td>The number of uniformly spaced bands to create, with a maximum or 250.</td>
</tr>
<tr>
<td><a href="">resc:Range</a></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Specifies the range of the data</td>
</tr>
</tbody>
</table>

The `<resc:ValueList>` element contains the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="">se:Value</a></td>
<td>String specifying a colour or a floating point number specifying a pattern density or opacity.</td>
<td>At least one must be specified if <code>&lt;se:Name&gt;</code> is omitted, or if patterns or opacities are being plotted.</td>
<td>The values to be interpolated between or subsampled.</td>
</tr>
<tr>
<td><a href="">se:Name</a></td>
<td>String</td>
<td>Must be specified if <code>&lt;se:Value&gt;</code> is omitted.</td>
<td>The name of a predefined palette, which contains a list of values.</td>
</tr>
</tbody>
</table>

The `<resc:Range>` element contains the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="">resc:Minimum</a></td>
<td>Floating point number</td>
<td>Must be specified</td>
<td>The minimum of the range.</td>
</tr>
<tr>
<td><a href="">resc:Maximum</a></td>
<td>Floating point number</td>
<td>Must be specified</td>
<td>The maximum of the range.</td>
</tr>
<tr>
<td><a href="">resc:Spacing</a></td>
<td>String</td>
<td>linear</td>
<td>Whether the bands will be spaced linearly (“linear”) or logarithmically (“logarithmic”)</td>
</tr>
</tbody>
</table>
For an example see the next section on colour maps.

## 2.3 Colour maps

### 2.3.1 Common elements

Colour maps are specified using the `<se:RasterSymbolizer>` tag. The latter must contain a `<se:ColorMap>` tag as well as an optional opacity transform. Within the colour map there must be one of three functions defining it either by a list of thresholds, a list of interpolation points, or a palette. Each colour map requires one variable.

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;se:ColorMap&gt;</code></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Must contain the function defining the colour map</td>
</tr>
</tbody>
</table>

### 2.3.2 Thresholds (SE compliant)

A colour map can be defined by a list of thresholds within the `<se:Categorize>` function (see the section on functions). The lists of thresholds and values should be in interleaved to be compliant with SE as in the following example:

```xml
<se:RasterSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <se:ColorMap>
    <se:Categorize fallbackValue="#00000000">
      <se:LookupValue>Rasterdata</se:LookupValue>
      <se:Value>#FF0000FF</se:Value>
      <se:Threshold>275.0</se:Threshold>
      <se:Value>#FF00FFFF</se:Value>
      <se:Threshold>280.0</se:Threshold>
      <se:Value>#FF00FF00</se:Value>
      <se:Threshold>285.0</se:Threshold>
      <se:Value>#FFFFFF00</se:Value>
      <se:Threshold>290.0</se:Threshold>
      <se:Value>#FFFFC800</se:Value>
      <se:Threshold>295.0</se:Threshold>
      <se:Value>#FFFFAFAF</se:Value>
      <se:Threshold>300.0</se:Threshold>
      <se:Value>#FFFF0000</se:Value>
    </se:Categorize>
  </se:ColorMap>
</se:RasterSymbolizer>
```

The above example is compliant with SE. Note that the opacity transform must be flat for this to be the case.

Ref: 03_ncWMS_Styling_Specification_1.0
2.3.3 Interpolation (SE compliant)
A colour map can be defined by a list of interpolation points within the <se:Interpolate> function (see the section on functions). The colour values will be interpolated linearly between the data points. For example:

```xml
<se:RasterSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <se:ColorMap>
    <se:Interpolate fallbackValue="#FF006400">
      <se:LookupValue>Rasterdata</se:LookupValue>
      <se:InterpolationPoint>
        <se:Data>265.0</se:Data>
        <se:Value>#FF0000FF</se:Value>
      </se:InterpolationPoint>
      <se:InterpolationPoint>
        <se:Data>285.0</se:Data>
        <se:Value>#FFFFFFFF</se:Value>
      </se:InterpolationPoint>
      <se:InterpolationPoint>
        <se:Data>305.0</se:Data>
        <se:Value>#FFFF0000</se:Value>
      </se:InterpolationPoint>
    </se:Interpolate>
  </se:ColorMap>
</se:RasterSymbolizer>
```

The above example is compliant with SE. Note that the opacity transform must be flat for this to be the case.

2.3.4 Segment and named palettes
A colour map can be defined by a named palette of list of colours comprising a palette within the <resc:Segment> function (see the section on functions). For example:

```xml
<se:RasterSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <se:ColorMap>
    <resc:Segment fallbackValue="#FF006400">
      <se:LookupValue>Rasterdata</se:LookupValue>
      <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
      <resc:ValueList>
        <se:Name>redblue</se:Name>
      </resc:ValueList>
      <resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
      <resc:Range>
        <resc:Minimum>270.0</resc:Minimum>
        <resc:Maximum>310.0</resc:Maximum>
        <resc:Spacing>linear</resc:Spacing>
      </resc:Range>
      <resc:NumberOfSegments>250</resc:NumberOfSegments>
    </resc:Segment>
  </se:ColorMap>
</se:RasterSymbolizer>
```
2.4  Bivariate colour maps

2.4.1  Common elements

A bivariate colour map may be specified using the `<resc:Raster2DSymbolizer>` tag. Currently only the `<resc:Categorize2D>` function is supported. It is intended to support perceptually linear colour maps in the future. They can be supported now by entering the corresponding sRGB values into the `<resc:Categorize2D>` function. Two variables are required, which must be specified using coverage constraints in a user layer. The following tags can be specified within a `<resc:Raster2DSymbolizer>` tag:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;resc:ColourMap2D&gt;</code></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Must contain the function defining the colour map</td>
</tr>
</tbody>
</table>

For example:

```xml
<resc:Raster2DSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <resc:ColorMap2D>
    <resc:Categorize2D fallbackValue="#FF006400">
      <se:LookupValue>Rasterdata</se:LookupValue>
      <!-- x thresholds define columns -->
      <resc:XThreshold>276.7</resc:XThreshold>
      <resc:XThreshold>283.3</resc:XThreshold>
      <resc:XThreshold>290.0</resc:XThreshold>
      <resc:XThreshold>296.7</resc:XThreshold>
      <resc:XThreshold>303.3</resc:XThreshold>
      <!-- y thresholds define rows -->
      <resc:YThreshold>0.5</resc:YThreshold>
      <resc:YThreshold>1.0</resc:YThreshold>
      <resc:YThreshold>1.5</resc:YThreshold>
      <resc:YThreshold>2.0</resc:YThreshold>
      <resc:YThreshold>2.5</resc:YThreshold>
      <!-- row #01 -->
      <se:Value>#FF0000FF</se:Value>
      <se:Value>#FF00FFFF</se:Value>
      <se:Value>#FF00FF00</se:Value>
      <se:Value>#FFFFFF00</se:Value>
      <se:Value>#FFFFC800</se:Value>
      <se:Value>#FFFF0000</se:Value>
      <!-- row #02 -->
      <se:Value>#CC0000FF</se:Value>
    </resc:Categorize2D>
  </resc:ColorMap2D>
</resc:Raster2DSymbolizer>
```

Ref: 03_ncWMS_Styling_Specification_1.0
2.5 Glyphs

2.5.1 In situ coloured glyphs

Currently only in situ coloured glyphs are supported using the `<resc:ColoredGlyphSymbolizer>` tag. It requires one variable. The following XML tags must be specified:
<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="">resc:IconName</a></td>
<td>String</td>
<td>“circle”</td>
<td>Name of the symbol to use for glyphs (currently either “circle” or “square”)</td>
</tr>
<tr>
<td><a href="">se:ColorMap</a></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Colour map for colouring the glyphs according to the value of the variable, as for colour maps above</td>
</tr>
</tbody>
</table>

For example:

```xml
<resc:ColoredGlyphSymbolizer>
    <se:Opacity>1.0</se:Opacity>
    <resc:IconName>circle</resc:IconName>
    <se:ColorMap>
        <resc:Segment fallbackValue="#FF006400">
            <se:LookupValue>Rasterdata</se:LookupValue>
            <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
            <resc:ValueList>
                <se:Name>redblue</se:Name>
            </resc:ValueList>
            <resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
            <se:Range>
                <resc:Minimum>-5.0</resc:Minimum>
                <resc:Maximum>40.0</resc:Maximum>
                <resc:Spacing>linear</resc:Spacing>
            </se:Range>
            <resc:NumberOfSegments>250</resc:NumberOfSegments>
        </resc:Segment>
    </se:ColorMap>
</resc:ColoredGlyphSymbolizer>
```

2.6 Arrows

Arrows are designed to represent the direction of vectors. The <resc:ArrowSymbolizer> element is used to specify the styling information for a layer of arrows. It requires one variable. This data should be a set of directions given as an angular compass bearing (degrees clockwise from North). The following tags can be used within this element:
<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="">resc:ArrowSize</a></td>
<td>Integer</td>
<td>8</td>
<td>Size of arrows</td>
</tr>
<tr>
<td><a href="">resc:ArrowColour</a></td>
<td>String</td>
<td>#FF000000 (Black)</td>
<td>Colour of arrows</td>
</tr>
</tbody>
</table>

For example:

```xml
<resc:ArrowSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <resc:ArrowSize>8</resc:ArrowSize>
  <resc:ArrowColour>#FF000000</resc:ArrowColour>
</resc:ArrowSymbolizer>
```

### 2.7 Confidence interval triangles

Confidence interval triangles are designed to show variables with an upper and lower bound. It requires two machine readable variable names that represent the upper and lower bound. These must be specified within coverage constraints in a user layer. The data is subsampled into squares, which are bisected into two triangles. The upper bound is shown in an upper triangle and the lower bound in the lower triangle. The contrast between the upper and lower triangles should indicate the uncertainty in the variable. Where there is no contrast the appearance is similar to a single variable colour map. They are specified by the XML element `<resc:ConfidenceIntervalSymbolizer>`. This can contain the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="">resc:GlyphSize</a></td>
<td>Integer</td>
<td>10</td>
<td>The size of the glyphs</td>
</tr>
<tr>
<td><a href="">se:ColorMap</a></td>
<td>XML element</td>
<td>Must be specified</td>
<td>Colour map for colouring the glyphs according to the value of the variable, as for colour maps above</td>
</tr>
</tbody>
</table>

For example:

```xml
<resc:ConfidenceIntervalSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <resc:GlyphSize>10</resc:GlyphSize>
  <se:ColorMap>
    <resc:Palette fallbackValue="#FF006400">
      <se:LookupValue>Rasterdata</se:LookupValue>
      <resc:PaletteDefinition>redblue</resc:PaletteDefinition>
    </resc:Palette>
  </se:ColorMap>
</resc:ConfidenceIntervalSymbolizer>
```
A variable can be plotted as contours by specifying the `<resc:ContourSymbolizer>` XML element. This requires one variable. It can contain the following tags:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;resc:NumberOfContours&gt;</code></td>
<td>Integer</td>
<td>10</td>
<td>The number of contour levels</td>
</tr>
<tr>
<td><code>&lt;resc:ContourLineColour&gt;</code></td>
<td>String specifying colour</td>
<td>#FF000000 (Black)</td>
<td>The colour of the contours</td>
</tr>
<tr>
<td><code>&lt;resc:Range&gt;</code></td>
<td>XML element</td>
<td></td>
<td>The scale range for the contours</td>
</tr>
<tr>
<td><code>&lt;resc:AutoRangeEnabled&gt;</code></td>
<td>Boolean</td>
<td>false</td>
<td>Whether to automatically fit the range of the contours to the range of the data</td>
</tr>
</tbody>
</table>

The `<resc:Range>` element contains the following tags similar to the `<resc:Range>` tag in the Segment function:

<table>
<thead>
<tr>
<th>Tag name</th>
<th>Type of contents</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;resc:Minimum&gt;</code></td>
<td>Floating point number</td>
<td>Must be specified</td>
<td>The minimum of the range.</td>
</tr>
<tr>
<td><code>&lt;resc:Maximum&gt;</code></td>
<td>Floating point number</td>
<td>Must be specified</td>
<td>The maximum of the range.</td>
</tr>
<tr>
<td>Tag name</td>
<td>Type of contents</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------</td>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><a href="">resc:Spacing</a></td>
<td>String</td>
<td>linear</td>
<td>Whether the contours will be spaced linearly (&quot;linear&quot;) or logarithmically (&quot;logarithmic&quot;)</td>
</tr>
</tbody>
</table>

It is possible to specify XML tags containing further styling information, but these are not currently supported and have no effect. An example of a specification for contours is:

```xml
<resc:ContourSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <resc:NumberOfContours>10</resc:NumberOfContours>
  <resc:ContourLineColour>#FF000000</resc:ContourLineColour>
  <resc:Range>
    <resc:Minimum>0.5</resc:Minimum>
    <resc:Maximum>2.5</resc:Maximum>
    <resc:Spacing>linear</resc:Spacing>
  </resc:Range>
  <resc:AutoRangeEnabled>false</resc:AutoRangeEnabled>
</resc:ContourSymbolizer>
```

### 2.9 Patterns

#### 2.9.1 Stippling

Stippling is designed to represent uncertainty by varying degrees of black stippling. The `<resc:StippleSymbolizer>` element is used to specify the styling information for a layer of stippling. It requires one variable. It must contain one function in which the density of stippling is represented by a floating point number between 0.0 for no stippling and 1.0 for solid black. Any function may be used. In the case where the interpolate function is used the stippling is varied uniformly between 65 possible different levels of stippling and only two interpolation points may be specified. Using the segment function for example:

```xml
<resc:StippleSymbolizer>
  <se:Opacity>1.0</se:Opacity>
  <resc:Segment fallbackValue="0.0">
    <se:LookupValue>Rasterdata</se:LookupValue>
    <resc:ValueList>
      <se:Value>0.0</se:Value>
      <se:Value>1.0</se:Value>
    </resc:ValueList>
    <resc:Range>
      <resc:Minimum>0.0</resc:Minimum>
      <resc:Maximum>2.5</resc:Maximum>
    </resc:Range>
  </resc:Segment>
</resc:StippleSymbolizer>
```

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading

Page 19 of 43
2.10 Opacity transforms

2.10.1 Flat
A flat opacity transform can be specified in any symbolizer with the <se:Opacity> tag. By default there is 100% opacity. The opacity tag can contain a floating point number between 0.0 for transparent and 1.0 for opaque. See the above sections for examples.

2.10.2 Opacity map
Instead of a flat opacity transform an opacity map can be specified using the XML element <resc:OpacityMap>. Within the latter is a function in which opacity is represented by a floating point number between 0.0 for transparent and 1.0 for opaque. This requires one variable, which is specified within the <se:LookupValue> tag in the function. The transform will vary spatially between entirely opaque and transparent based on the value of the variable at a particular point in space. Any function may be used. In the case of the Interpolate function only two interpolation points may be specified. For example:

```xml
<resc:OpacityTransform>
  <se:Interpolate fallbackValue="1.0">
    <se:LookupValue>OSTIA/analysis_error</se:LookupValue>
    <se:InterpolationPoint>
      <se:Data>0.0</se:Data>
      <se:Value>1.0</se:Value>
    </se:InterpolationPoint>
    <se:InterpolationPoint>
      <se:Data>2.5</se:Data>
      <se:Value>0.0</se:Value>
    </se:InterpolationPoint>
  </se:Interpolate>
</resc:OpacityTransform>
```

3 Examples of usage of XML tags

3.1 Thresholds and interpolation
This example XML file is SE compliant. It uses interpolation to plot a reanalysis of sea surface temperature as a colour map and thresholds to plot the analysis error as degrees of blackening, also as a colour map.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
  xsi:schemaLocation="http://www.opengis.net/sld
StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"

Ref: 03_ncWMS_Styling_Specification_1.0
```

Copyright © 2012 University of Reading
The resulting output image is shown below:

The legend for the map is below:
In this and all the legends that follow the topography is there to indicate regions where there is a degree of opacity. In this case it is where there is no data present for the SST over land.

### 3.2 Palette

This example XML file displays a reanalysis of sea surface temperature as a named red-blue palette and the analysis error as bands of blackening specified using thresholds.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
 xsi:schemaLocation="http://www.opengis.net/sld
 StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
 xmlns:ogc="http://www.opengis.net/ogc"
 xmlns:se="http://www.opengis.net/se"
 xmlns:xlink="http://www.w3.org/1999/xlink"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:resc="http://www.resc.reading.ac.uk">
 <NamedLayer>
  <se:Name>OSTIA/analysed_sst</se:Name>
  <UserStyle>
   <se:Name>Red-blue palette showing SST</se:Name>
   <se:CoverageStyle>
    <se:Rule>
     <se:RasterSymbolizer>
      <se:Opacity>1.0</se:Opacity>
      <se:ColorMap>
       <resc:Segment fallbackValue="#FF006400">
        <se:LookupValue>Rasterdata</se:LookupValue>
        <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
        <resc:ValueList>
         Ref: 03_ncWMS_Styling_Specification_1.0
        </resc:ValueList>
     </resc:Segment>
    </se:ColorMap>
   </se:RasterSymbolizer>
  </se:Cov
```
<se:Name>redblue</se:Name>
</resc:ValueList>
<resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
<resc:Range>
<resc:Minimum>270.0</resc:Minimum>
<resc:Maximum>310.0</resc:Maximum>
<resc:Spacing>linear</resc:Spacing>
</resc:Range>
<resc:NumberOfSegments>250</resc:NumberOfSegments>
</resc:Segment>
</se:ColorMap>
</se:RasterSymbolizer>
</se:Rule>
</se:CoverageStyle>
</UserStyle>
</NamedLayer>
<NamedLayer>
<se:Name>OSTIA/analysis_error</se:Name>
<UserStyle>
<se:Name>Show uncertainty with thresholded blackening</se:Name>
<se:CoverageStyle>
<se:Rule>
<se:RasterSymbolizer>
<se:Opacity>1.0</se:Opacity>
<se:ColorMap>
<se:Categorize fallbackValue="#00000000">
<se:LookupValue>Rasterdata</se:LookupValue>
<se:Value>#00000000</se:Value>
<se:Threshold>0.5</se:Threshold>
<se:Value>#33000000</se:Value>
<se:Threshold>1.0</se:Threshold>
<se:Value>#66000000</se:Value>
<se:Threshold>1.5</se:Threshold>
<se:Value>#99000000</se:Value>
<se:Threshold>2.0</se:Threshold>
<se:Value>#CC000000</se:Value>
<se:Threshold>2.5</se:Threshold>
<se:Value>#FF000000</se:Value>
</se:Categorize>
</se:ColorMap>
</se:RasterSymbolizer>
</se:Rule>
</se:CoverageStyle>
</UserStyle>
</NamedLayer>
</StyledLayerDescriptor>

The resulting output image is shown below:

Ref: 03_ncWMS_Styling_Specification_1.0
The legend for the map is below:

![Legend Image]

### 3.3 Bivariate colour map

This example XML file displays a reanalysis of sea surface temperature and the analysis error as a perceptually uniform bivariate colour map (CIECAM02). The lightness varies with temperature and colorfulness with the error. An external program was used to generate the RGB values that are closest to the CIECAM02 values. There is an error in this encoding, which is the distance between the RGB and
CIECAM02 values in the colour space. If this is less than 1 then the error is small, which it is not for all of the values in the XML below. The name of the user layer is ignored by the parser.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
  xsi:schemaLocation="http://www.opengis.net/sld
  StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:se="http://www.opengis.net/se"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:resc="http://www.resc.reading.ac.uk">
  <UserLayer>
    <se:Name>OSTIA/analysed_sst_bivariate_colourmap</se:Name>
    <LayerCoverageConstraints>
      <CoverageConstraint>
        <se:CoverageName>OSTIA/analysis_error</se:CoverageName>
      </CoverageConstraint>
      <CoverageConstraint>
        <se:CoverageName>OSTIA/analysed_sst</se:CoverageName>
      </CoverageConstraint>
    </LayerCoverageConstraints>
    <UserStyle>
      <se:Name>2D thresholded colour scheme</se:Name>
      <se:CoverageStyle>
        <se:Rule>
          <resc:Raster2DSymbolizer>
            <se:Opacity>1.0</se:Opacity>
            <resc:ColorMap2D>
              <resc:Categorize2D fallbackValue="#00000000">
                <se:LookupValue>Rasterdata</se:LookupValue>
                <!-- x thresholds define columns -->
                <resc:XThreshold>0.6</resc:XThreshold>
                <resc:XThreshold>1.2</resc:XThreshold>
                <resc:XThreshold>1.8</resc:XThreshold>
                <resc:XThreshold>2.4</resc:XThreshold>
                <!-- y thresholds define rows -->
                <resc:YThreshold>275.0</resc:YThreshold>
                <resc:YThreshold>278.5</resc:YThreshold>
                <resc:YThreshold>282.0</resc:YThreshold>
                <resc:YThreshold>285.5</resc:YThreshold>
                <resc:YThreshold>289.0</resc:YThreshold>
                <resc:YThreshold>292.5</resc:YThreshold>
                <resc:YThreshold>296.0</resc:YThreshold>
                <resc:YThreshold>299.5</resc:YThreshold>
                <resc:YThreshold>303.0</resc:YThreshold>
                <!-- Colors of equal Hue (h = 65.0) -->
                <!-- Colorfulness (C) and Lightness (J) varied according to
                CIECAM02. -->
              </resc:Categorize2D>
            </resc:ColorMap2D>
          </resc:Raster2DSymbolizer>
        </se:Rule>
      </se:CoverageStyle>
    </UserStyle>
  </UserLayer>
</StyledLayerDescriptor>
```

Ref: 03_ncWMS_Styling_Specification_1.0
The resulting output image is shown below:

![Map Output Image](image-url)

The legend for the map is below:
3.4 In situ coloured glyphs

This example XML file displays in situ measurements of sea surface temperature a coloured glyphs.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
xsi:schemaLocation="http://www.opengis.net/sld
StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:se="http://www.opengis.net/se"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:resc="http://www.resc.reading.ac.uk">
  <NamedLayer>
    <se:Name>EN3/TEMP</se:Name>
    <UserStyle>
      <se:Name>Red-blue palette showing SST with basic glyphs</se:Name>
      <se:CoverageStyle>
        <resc:ColoredGlyphSymbolizer>
          <se:Opacity>1.0</se:Opacity>
          <resc:IconName>circle</resc:IconName>
          <resc:ColorMap>
            <resc:Segment fallbackValue="#FF006400">
              <se:LookupValue>Rasterdata</se:LookupValue>
              <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
              <resc:ValueList>
                <se:Name>redblue</se:Name>
              </resc:ValueList>
              <resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
            </resc:Segment>
          </resc:ColorMap>
          <resc:Segment fallbackValue="#FF006400">
            <se:LookupValue>Rasterdata</se:LookupValue>
            <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
            <resc:ValueList>
              <se:Name>redblue</se:Name>
            </resc:ValueList>
            <resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
          </resc:Segment>
        </resc:ColoredGlyphSymbolizer>
      </se:CoverageStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```

Ref: 03_ncWMS_Styling_Specification_1.0
The resulting output is shown below:

![Image of sea water velocity direction as arrows on top of sea surface temperature plotted as a colour map]

### 3.5 Arrows
This example XML file displays sea water velocity direction as arrows on top of sea surface temperature plotted as a colour map.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
  xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:se="http://www.opengis.net/se"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <NamedLayer>
    <UserStyle>
      <CoverageStyle>
        <ColorMap>
          <Range>
            <Minimum>-5.0</Minimum>
            <Maximum>40.0</Maximum>
            <Spacing>linear</Spacing>
          </Range>
          <NumberOfSegments>250</NumberOfSegments>
        </ColorMap>
        <ColoredGlyphSymbolizer>
          <Rule>
            <CoverageStyle>
              <NamedLayer>
                </NamedLayer>
              </UserStyle>
            </UserStyle>
          </CoverageStyle>
        </ColoredGlyphSymbolizer>
      </CoverageStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```
The resulting output image is shown below:
The legend for the map is below:

3.6 Confidence interval triangles

This example XML file plots sea surface temperature using confidence interval triangles to show the uncertainty with a perceptually uniform colour map (CIECAM02). An external program was used to encode the CIECAM02 values into sRGB. There is an error in doing this as explained in bivariate colour maps above. The colours were chosen in this case to be ones with small errors (less than 1). The

Ref: 03_ncWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
The magnitude of the SST is represented by lightness and the degree of uncertainty is shown by the contrast between the upper and lower confidence bounds.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
  xsi:schemaLocation="http://www.opengis.net/sld
  StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:se="http://www.opengis.net/se"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:resc="http://www.resc.reading.ac.uk">
  <UserLayer>
    <se:Name>OSTIA/analysed_sst_confidence_intervals</se:Name>
    <LayerCoverageConstraints>
      <CoverageConstraint>
        <se:CoverageName>OSTIA/sst_lb</se:CoverageName>
      </CoverageConstraint>
      <CoverageConstraint>
        <se:CoverageName>OSTIA/sst_ub</se:CoverageName>
      </CoverageConstraint>
    </LayerCoverageConstraints>
    <UserStyle>
      <se:Name>Split triangles with perceptual color map</se:Name>
      <se:CoverageStyle>
        <se:Rule>
          <resc:ConfidenceIntervalSymbolizer>
            <se:Opacity>1.0</se:Opacity>
            <resc:GlyphSize>10</resc:GlyphSize>
            <se:ColorMap>
              <se:Categorize fallbackValue="#00000000">
                <se:LookupValue>Rasterdata</se:LookupValue>
                <!-- Colors of equal Hue (h = 65.0) -->
                <!-- and Colorfulness (C = 25.0). -->
                <!-- Lightness (J) varied between 15.0 and 85.0 according to
                CIECAM02. -->
                <se:Value>#FFFFD5A8</se:Value>
                <se:Threshold>274.0</se:Threshold>
                <se:Value>#FFFDD1A4</se:Value>
                <se:Threshold>275.0</se:Threshold>
                <se:Value>#FFF9CDA0</se:Value>
                <se:Threshold>276.0</se:Threshold>
                <se:Value>#FFF4C99C</se:Value>
                <se:Threshold>277.0</se:Threshold>
                <se:Value>#FFEFC498</se:Value>
                <se:Threshold>278.0</se:Threshold>
                <se:Value>#FFEBC094</se:Value>
                <se:Threshold>279.0</se:Threshold>
                <se:Value>#FFE6BC90</se:Value>
              </se:Categorize>
            </se:ColorMap>
          </resc:ConfidenceIntervalSymbolizer>
        </se:Rule>
      </se:CoverageStyle>
    </UserStyle>
  </UserLayer>
</StyledLayerDescriptor>
```

Ref: 03_nWMS_Styling_Specification_1.0

Copyright © 2012 University of Reading
The resulting output image is shown below:

The legend for the map is below:
### 3.7 Contours

The following example XML file shows a reanalysis of sea surface temperature as a colour map, with the analysis error as contours.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:se="http://www.opengis.net/se"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmns:resc="http://www.resc.reading.ac.uk">
  <NamedLayer>
    <se:Name>OSTIA/analysed_sst</se:Name>
    <UserStyle>
      <se:Name>Red-blue palette showing SST</se:Name>
      <se:CoverageStyle>
        <se:Rule>
          <se:RasterSymbolizer>
            <se:Opacity>1.0</se:Opacity>
            <se:ColorMap>
              <resc:Segment fallbackValue="#FF006400">
                <se:LookupValue>Rasterdata</se:LookupValue>
                <resc:BelowMinValue>#FF0000FF</resc:BelowMinValue>
                <resc:ValueList>
                  <se:Name>redblue</se:Name>
                </resc:ValueList>
                <resc:AboveMaxValue>#FFFF0000</resc:AboveMaxValue>
              </resc:Segment>
            </se:ColorMap>
          </se:RasterSymbolizer>
        </se:Rule>
      </se:CoverageStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```

Ref: [03_ncWMS_Styling_Specification_1.0](#)
<resc:Range>
  <resc:Minimum>270.0</resc:Minimum>
  <resc:Maximum>310.0</resc:Maximum>
  <resc:Spacing>linear</resc:Spacing>
</resc:Range>
<resc:NumberOfSegments>250</resc:NumberOfSegments>
</resc:Segment>
</se:ColorMap>
</se:RasterSymbolizer>
</se:Rule>
</UserStyle>
</NamedLayer>
<NamedLayer>
  <se:Name>OSTIA/analysis_error</se:Name>
  <UserStyle>
    <se:Name>Contours showing error</se:Name>
    <se:CoverageStyle>
      <se:Rule>
        <resc:ContourSymbolizer>
          <se:Opacity>1.0</se:Opacity>
          <resc:NumberOfContours>10</resc:NumberOfContours>
          <resc:ContourLineColour>#FF000000</resc:ContourLineColour>
          <resc:ContourLineWidth>1</resc:ContourLineWidth>
          <resc:ContourLineStyle>DASHED</resc:ContourLineStyle>
          <resc:LabelEnabled>true</resc:LabelEnabled>
          <resc:Range>
            <resc:Minimum>0.5</resc:Minimum>
            <resc:Maximum>2.5</resc:Maximum>
            <resc:Spacing>linear</resc:Spacing>
          </resc:Range>
          <resc:AutoRangeEnabled>false</resc:AutoRangeEnabled>
        </resc:ContourSymbolizer>
      </se:Rule>
    </se:CoverageStyle>
  </UserStyle>
</NamedLayer>
</StyledLayerDescriptor>

The resulting output image is shown below:
The legend for the map is below:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
 xsi:schemaLocation="http://www.opengis.net/sld
 StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
 Ref: 03_ncWMS_Styling_Specification_1.0
```

### 3.8 Stippling

The following example XML file shows the analysis error as stippling on top of a reanalysis of sea surface temperature.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
 xsi:schemaLocation="http://www.opengis.net/sld
 StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
 Ref: 03_ncWMS_Styling_Specification_1.0
```
The resulting output image is shown below:

The legend for the map is below:
3.9 Linear opacity transform

The following example XML shows a reanalysis of sea surface temperature as a colour map. The analysis error is represented by a linear opacity transform. The more uncertain an area is the more transparent it appears. The opacity transform only applies to the layer.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<StyledLayerDescriptor version="1.1.0"
 xsi:schemaLocation="http://www.opengis.net/sld
 StyledLayerDescriptor.xsd" xmlns="http://www.opengis.net/sld"
 xmlns:ogc="http://www.opengis.net/ogc"
 xmlns:se="http://www.opengis.net/se"
 xmlns:xlink="http://www.w3.org/1999/xlink"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:resc="http://www.resc.reading.ac.uk">
 <NamedLayer>
  <se:Name>OSTIA/analysed_sst</se:Name>
  <UserStyle>
   <se:Name>Red-blue palette showing SST</se:Name>
   <se:CoverageStyle>
    <se:Rule>
     <se:RasterSymbolizer>
      <resc:OpacityTransform>
       <se:Interpolate fallbackValue="1.0">
        <se:LookupValue>OSTIA/analysis_error</se:LookupValue>
        <se:InterpolationPoint>
         <se:Data>0.0</se:Data>
         <se:Value>1.0</se:Value>
        </se:InterpolationPoint>
        <se:InterpolationPoint>
         <se:Data>310.0</se:Data>
         <se:Value>0.0</se:Value>
        </se:InterpolationPoint>
      </se:Interpolate>
     </resc:OpacityTransform>
    </se:RasterSymbolizer>
   </se:Rule>
  </se:CoverageStyle>
 </UserStyle>
</NamedLayer>
</StyledLayerDescriptor>
```

Ref: 03_ncWMS_Styling_Specification_1.0
The resulting output image is shown below:
The legend for the map is below: