



PLASMAR+

www.plasmar.eu



PROYECTO COFINANCIADO
POR LA UNIÓN EUROPEA
Investigación
e Innovación



Interreg 
Fondo Europeo de Desarrollo Regional

SIGMAR SLD: STYLING GEOGRAPHIC DATA ON THE SIGMAR GEOPORTAL

Paulo Miranda, Emanuel Silveira & Gilberto Carreira
Regional Directorate for Maritime Policies

www.plasmar.eu
f www.facebook.com/ProyectoPlasmar



Secretaria Regional de Mar e Pescas
Direção Regional do Mar



To cite this report:

Miranda, P.; Silveira, E.; Carreira, G. 2023. SIGMAR SLD: Styling geographic data on the SIGMAR geoportal. Regional Directorate for Maritime Policies. Report prepared as part of PLASMAR+ Project (co-financed by ERDF as part of POMAC 2014-2020). 17 pp.

Reproduction is authorised provided the source is acknowledged.

LEGAL NOTICE

This document has been prepared as part of PLASMAR+ Project (MAC2/1.1a/347), however it reflects the views only of the authors, and the Project partners or POMAC 2014-2010 programme cannot be held responsible for any use which may be made of the information contained therein.

TABLE OF CONTENTS

1 Introduction	2
1.1 SLD and SE Specifications	2
2 Symbolizers: Points, Polygons, and Lines	6
2.1 PointSymbolizer	6
2.2 LineSymbolizer	8
2.3 PolygonSymbolizer	9
2.4 Additional Styling Properties	12
2.4.1 Line Break in Legend	12
3 Filters	12
4 Adaptative Scale / Zoom	14
5 Conclusion and References	15
5.1.1 References	16



1. Introduction

This document provides an overview of the Styled Layer Descriptor (SLD) and Symbology Encoding (SE) specifications used for styling geographic data in the SIGMAR GeoServer.

SIGMAR is the Spatial Data Infrastructure (SDI) of the Regional Directorate for Maritime Policies (DRPM), where geospatial information related to the Azorean Sea is made available, namely geographical information regarding the maritime spatial planning process.

This infrastructure comprises a PostgreSQL database, a geospatial service provider (geoserver), and a geoportal based on the Mapstore2 framework. These components are interconnected, enabling the geographic information, stored in the PostgreSQL database, have associated geospatial data services (Web Map Services (WMS) and Web Feature Services (WFS)), which feeds the Mapstore2 viewers.

The configuration of the Symbologies through the geoserver allows that WMS and WFS services could carry these styles and make them be rendered in any compatible viewer framework. This is the only method that SIGMAR Geoportal has for now available to control the features representation and labelling withing the mapstore2 viewers.

This document was developed as part of action 2.3.2 “INFRAESTRUCTURAS DE DATOS Y PLATAFORMAS” of the PLASMAR+ project. It aims to be a relevant documentation and a guide for the symbolization of the remaining layers in SIGMAR geoportal.

1.1. SLD and SE Specifications

SLD is a specification in a XML-based markup document format established by the Open Geospatial Consortium (OGC), that defines the visualization properties for geospatial data layers in services such as Web Map Service (WMS) and Web Feature Service (WFS). It allows the customization of the appearance of map features, such as colour, size, opacity, and apply filters or control the styling based on the scale of visualization.

The SLD file is structured with a XML declaration and the StyledLayerDescriptor root element as follow:

```
<?xml version="1.0" encoding="UTF-8"?>
```

The first line describes the XML document version and the encoding used. The encoding must be set to UTF-8 to ensure proper handling of characters from various alphabets, such as ç, ã, à, and others.

The first line of the root element has the SLD version and the schema declarations. In the SIGMAR symbology's were also used other schemas as follow:

```
<StyledLayerDescriptor xmlns:XMLSchema-instance="http://www.w3.org/2001/XMLSchema-instance" XMLSchema-
instance:schemaLocation="http://www.opengis.net/sld
http://schemas.opengis.net/sld/1.1.0/StyledLayerDescriptor.xsd" version="1.1.0"
xmlns="http://www.opengis.net/sld" xmlns:se="http://www.opengis.net/se"
xmlns:ogc="http://www.opengis.net/ogc">

</StyledLayerDescriptor>
```

In the StyledLayerDescriptor element, the SE (Symbology Encoding) and OGC (Open Geospatial Consortium) schemas are also declared. These schemas are defined by schema locations, namely <http://www.opengis.net/se> and <http://www.opengis.net/ogc>.

The declaration of these schemas in the root element allows them to be referenced throughout the document without explicitly specifying their namespaces for each element. For instance, the declaration `xmlns:se="http://www.opengis.net/se"` enables the usage of the "se:" prefix instead of explicitly specifying the namespace for each element.

```
<PointSymbolizer xmlns="http://www.opengis.net/se">
  <Graphic xmlns="http://www.opengis.net/se">
    <Mark xmlns="http://www.opengis.net/se">
      <WellKnownName xmlns="http://www.opengis.net/se">circle</WellKnownName>
      <Fill xmlns="http://www.opengis.net/se">
        <SvgParameter xmlns="http://www.opengis.net/se" name="fill">#e52121</SvgParameter>
      </Fill>
      <Stroke xmlns="http://www.opengis.net/se">
        <SvgParameter xmlns="http://www.opengis.net/se" name="stroke">#000000</SvgParameter>
        <SvgParameter xmlns="http://www.opengis.net/se" name="stroke-width">0.4</SvgParameter>
      </Stroke>
    </Mark>
    <Size xmlns="http://www.opengis.net/se">12</Size>
  </Graphic>
</PointSymbolizer>

<se:PointSymbolizer>
  <se:Graphic>
    <se:Mark>
      <se:WellKnownName>circle</se:WellKnownName>
      <se:Fill>
        <se:SvgParameter name="fill">#e52121</se:SvgParameter>
      </se:Fill>
      <se:Stroke>
        <se:SvgParameter name="stroke">#000000</se:SvgParameter>
        <se:SvgParameter name="stroke-width">0.4</se:SvgParameter>
      </se:Stroke>
    </se:Mark>
    <se:Size>12</se:Size>
  </se:Graphic>
</se:PointSymbolizer>
```

The Symbology Encoding (SE) defines an XML Schema that can be used for styling feature and coverage data. It provides a standard way to describe and define visual properties for geographic features such as points, lines, and polygons. It specifies the elements and attributes that can be used to define symbolizers, colour ramps,

label placements, and other styling aspects. This schema is primarily focused on the representation of cartographic styles.

The Open Geospatial Consortium (OGC) schema allows the usage of elements related with data access and filters.

After the schema declaration line, a basic structure is developed, consisting of the elements NamedLayer, UserStyle, and FeatureTypeStyle, within which the rules and symbolizers are defined. Note that this is the main elements structure used in SIGMAR symbologies, but other could also be used.

```
<NamedLayer>
  <se:Name>Tipologia_NL</se:Name>
  <UserStyle>
    <se:Name>Tipologia_US</se:Name>
    <se:FeatureTypeStyle>

      <!-- Rules and Symbolizers -->

    </se:FeatureTypeStyle>
  </UserStyle>
</NamedLayer>
```

Within the <FeatureTypeStyle> element, rules are created and identified by the <Rule> element, allowing for individualizing different visualization styles.

```
<se:FeatureTypeStyle>
  <se:Rule>
    <se:Name>Rule_1</se:Name>
    <se:PointSymbolizer>
      <!-- represents a point -->
    </se:PointSymbolizer>
  </se:Rule>

  <se:Rule>
    <se:Name>Rule_2</se:Name>
    <se:PolygonSymbolizer>
      <!-- represents a polygon -->
    </se:PolygonSymbolizer>
  </se:Rule>
</se:FeatureTypeStyle>
```

Within each <rule>, there are a set of other elements that allow describing, filtering, conditionally visualizing, and defining the characteristics of graphic elements.

Table 1 presents the definition of the main elements inside rules in the SIGMAR geoserver. They are listed by usage order inside each rule.

Note that the structure of an SLD document follows a specific order to ensure the correct definition of styles and rules. It is important to observe this order to avoid formatting errors and ensure that the document is correctly interpreted by geoprocessing applications.

Table 1 – Main elements used within <Rule> element

Element	Description
<Name> <i>SE schema</i>	Unique element for each rule. This element is used as the legend text when the <Description> element is not present. In symbolizations where different rules have the same legend text, it is mandatory to differentiate the <Name> element and use the <Description> element for the legend. <pre><se:Name>Rule_1</se:Name></pre>
<Description> <i>SE schema</i>	This element, when present, replaces the <Name> element to represent the legend text. It can be considered as an alias of the <Name> element. This element is optional but becomes mandatory when labelling two different rules with the same text. Within this element is used the <Title> sub-element that will carry the text to be used in the legend. <pre><se:Rule> <se:Name>Rule_1</se:Name> <se:Description> <se:Title>Legend_Text</se:Title> </se:Description> ...</pre>
<Filter> <i>OGC schema</i>	Optional element. Allows filtering items in a layer based on the value of one or more fields in the corresponding attribute table. Used in SIGMAR to classify different elements of the same layer with differentiated symbology. It has several sub-elements that will be described in Section 3 of this document.
<MinScaleDenominator> <MaxScaleDenominator> <i>SE schema</i>	Optional elements. They allow conditioning the representation of the layer according to the visualization scale. The use of these elements can be found in Section 4 of this document.
<PointSymbolizer> <LineSymbolizer> <PolygonSymbolizer> <i>SE schema</i>	Symbolizers. At least one of these elements is required. They can be used in different types of combinations. For more details, refer to Section 2 of this document, where examples of these combinations as well as several of their sub-elements can be viewed.

Here is an example that shows the general structure of a rule with several sub-elements in their correct order:


```
<se:Rule>
  <se:Name>Rule_1</se:Name>
  <se:Description>
    <se:Title>Legend_Text</se:Title>
  </se:Description>

  <ogc:Filter>
    ...
  </ogc:Filter>

  <se:MinScaleDenominator>10000</se:MinScaleDenominator>

  <se:PointSymbolizer>
    ...
  </se:PointSymbolizer>
</se:Rule>
```

2. Symbolizers: Points, Polygons, and Lines

Symbolizers define the visual representation of features on a map supporting styling of points, polygons, lines, text, or raster. They specify the shape of the feature and the graphic properties like colour and opacity. Each Symbolizer is associated with a specific type, and its default behaviour can be overridden by providing additional parameters.

This chapter focuses on the elements of `<PointSymbolizer>`, `<LineSymbolizer>` and `<PolygonSymbolizer>` and presents the ones used in the SIGMAR SLDs.

TextSymbolizers and RasterSymbolizers were not used in the SIGMAR SLDs.

2.1 PointSymbolizer

The `<PointSymbolizer>` element is used to represent a graphic or symbol at a specific point location, defining a point feature on a map. In the SIGMAR symbologies it was used to represent point features or to represent the centroid of a polygon feature.

The main element within `<PointSymbolizer>` is the `<Graphic>` element. Within it it's possible to define two elements as `<mark>` or `<ExternalGraphic>` that defines the type of symbol (if it's a shape or an external image) and three elements as `<Opacity>`, `<Size>` and `<Rotation>` that defines general properties common to both symbol types.

In the next table is possible to find the description of these elements that could be found inside the `<Graphic>` element.

Table 2 – Elements used within <Graphic> element

Element	Description
<Mark> <i>SE schema</i>	Mandatory if there are no <ExternalGraphic> element. Allows to use a known shape as the symbol. Inside of it, it's possible to define properties as the <WellKnownName>, <Fill> and <Stroke>.
<ExternalGraphic> <i>SE schema</i>	Mandatory if there are no <Mark> element. This element allows the usage of image files in formats as GIF, PNG or SVG. This element has two sub-elements <OnlineResource> and <Format>. <pre><se:ExternalGraphic> <se:OnlineResource xlink:type="simple" xlink:href="filepath.svg"></se:OnlineResource> <se:Format>image/svg+xml</se:Format> </se:ExternalGraphic></pre>
<Opacity> <i>SE schema</i>	Optional. Specifies the transparency of the symbol. Values can range from 0 to 1, where 0 represents complete transparency and 1 represents full opacity. If this element is not specified, the default value of 1 will be assumed.
<Size> <i>SE schema</i>	Optional. Specifies the size of the symbol. The value represents the height of the symbol in pixels. If this element is not specified, the default size will be assumed. <pre><se:Size>10</se:Size></pre>
<Rotation> <i>SE schema</i>	Optional. Specifies the rotation of the symbol in decimal degrees, based on its center point. If this element is not specified, the default value of 0 (zero) will be assumed. <pre><se:Rotation>45</se:Rotation></pre>

Next lines show two different examples using PointSymbolizers, the first one using the <Mark> element and the second one using the <ExternalGraphic> element. The sub-elements that could be used within the <Mark> element are shown in Table 3.

```
<se:PointSymbolizer>
  <se:Graphic>
    <se:Mark>
      <se:WellKnownName>square</se:WellKnownName>
      <se:Fill>
        <se:SvgParameter name="fill">#00FFFF</se:SvgParameter>
      </se:Fill>
      <se:Stroke>
        <se:SvgParameter name="stroke">#000000</se:SvgParameter>
        <se:SvgParameter name="stroke-width">0.4</se:SvgParameter>
      </se:Stroke>
    </se:Mark>
    <se:Size>10</se:Size>
    <se:Rotation>45</se:Rotation>
  </se:Graphic>
</se:PointSymbolizer>

<se:PointSymbolizer>
  <se:Graphic>
    <se:ExternalGraphic>
      <se:OnlineResource xlink:type="simple" xlink:href="filepath.svg"></se:OnlineResource>
      <se:Format>image/svg+xml</se:Format>
    </se:ExternalGraphic>
    <se:Size>25</se:Size>
  </se:Graphic>
</se:PointSymbolizer>
```

Table 3 – Elements used within <Mark> element

Element	Description
<WellKnownName> <i>SE schema</i>	Optional. Identifies the pre-defined shapes or symbols that can be used plot points as circle, square, triangle, cross, X and star. If this element is not specified, the default value square will be assumed. <code><se:WellKnownName>star</se:WellKnownName></code>
<Fill> <i>SE schema</i>	Optional. Specifies a solid fill colour using the <SvgParameter> element, the parameter name="fill" and a colour in hexadecimal value. <code><se:Fill> <se:SvgParameter name="fill">#00FFFF</se:SvgParameter> </se:Fill></code>
<Stroke> <i>SE schema</i>	Optional. If present, this element allows to define an outline to the symbol. With the parameters from the <SvgParameter> element, it's possible to specifies for example the line colour, width, and opacity. <code><se:Stroke> <se:SvgParameter name="stroke">#000000</se:SvgParameter> <se:SvgParameter name="stroke-width">0.4</se:SvgParameter> </se:Stroke></code>

2.2 LineSymbolizer

The <LineSymbolizer> element is used to style lines. Its main sub-element is the <Stroke> element that with parameters from the <SvgParameter> element, may specify for example the line colour, width, and opacity.

Next, it's possible to find one example of the code associated to the <LineSymbolizer> element and its corresponding visual representation on a map.

The "stroke" parameter is used to define the hexadecimal value for the line colour.

The "stroke-width" is used to define the line with.

The "stroke-linejoin" is used to represent a smooth join at angles formed between different line segments.

code

```
<LineSymbolizer>
  <se:Stroke>
    <se:SvgParameter name="stroke">#FF8C00</se:SvgParameter>
    <se:SvgParameter name="stroke-width">1</se:SvgParameter>
    <se:SvgParameter name="stroke-linejoin">bevel</se:SvgParameter>
    <se:SvgParameter name="stroke-linecap">square</se:SvgParameter>
    <se:SvgParameter name="stroke-dasharray">8 6 2 6</se:SvgParameter>
  </se:Stroke>
</LineSymbolizer>
```

Visual representation



The "stroke-linecap" is used to represent the shape used in the end of a line.

Finally, The “Stroke-dasharray” is used to define a dashed line. It is formed by a sequence of numbers separated by spaces. The odd number positions represents the length in pixels of the draw line segments, on the other hand, the even number positions represents the length in pixels of the spaces between line segments.

For the given example, the dashed line will have a sequence of dashes, the first one with 8 pixels of length, and the second one with 2 pixels of length, both separated by spaces of 6 pixels of length.

In the next example, the dashed line will have a sequence of dashes with 5 pixels of length intercalated by spaces of 10 pixels of length.

```
code
    <se:LineSymbolizer>
      <se:Stroke>
        <se:SvgParameter name="stroke" style="color:blue">#FF0000</se:SvgParameter>
        <se:SvgParameter name="stroke-width" style="color:blue">1</se:SvgParameter>
        <se:SvgParameter name="stroke-linejoin" style="color:blue">bevel</se:SvgParameter>
        <se:SvgParameter name="stroke-linecap" style="color:blue">square</se:SvgParameter>
        <se:SvgParameter name="stroke-dasharray" style="color:red">5 10</se:SvgParameter>
      </se:Stroke>
    </se:LineSymbolizer>
```

Visual representation



2.3 PolygonSymbolizer

The <PolygonSymbolizer> element is used to represents a polygon with an interior <fill> and an outline <stroke>.

The syntax of the <stroke> element is similar with the one in the 2.2 section.

The <fill> element has the sub-elements: One, used to represent repeated patterns named <GraphicFill>, used to represent repeated patterns, and one or several <SvgParameter> used to configure a solid colour to the fill.

The <GraphicFill> element contains a <Graphic> element which syntax is described is similar to the one described in the 2.1 section.

In the following example, the polygon symbology is demonstrated using a repeated pattern represented by an X shape given by the value "shape://times". This X shape is repeated without any spacing throughout the polygon. The symbology also includes an outline that surrounds the entire polygon.

code

```
<se:Rule>
  <se:Name>Geossítios Marinhos</se:Name>
  <se:PolygonSymbolizer>
    <se:Fill>
      <se:GraphicFill>
        <se:Graphic>
          <se:Mark>
            <se:WellKnownName>shape://times</se:WellKnownName>
            <se:Stroke>
              <se:SvgParameter name="stroke">#b5b5ae</se:SvgParameter>
              <se:SvgParameter name="stroke-width">1</se:SvgParameter>
            </se:Stroke>
          </se:Mark>
          <se:Size>10</se:Size>
        </se:Graphic>
      </se:GraphicFill>
    </se:Fill>
    <se:Stroke>
      <se:SvgParameter name="stroke">#b5b5ae</se:SvgParameter>
      <se:SvgParameter name="stroke-width">1</se:SvgParameter>
      <se:SvgParameter name="stroke-linejoin">bevel</se:SvgParameter>
    </se:Stroke>
  </se:PolygonSymbolizer>
</se:Rule>
```

Visual representation



Note that inside each PolygonSymbolizer it's only possible to have one <Fill> element. If would be necessary to have two different <Fill> it's mandatory to encapsule them within two different <PolygonSymbolizer> elements, as follows:

code

```
<se:Rule>
  <se:Name>Parque Marinho dos Açores</se:Name>
  <se:PolygonSymbolizer>
    <se:Fill>
      <se:GraphicFill>
        <se:Graphic>
          <se:Mark>
            <se:WellKnownName>shape://slash</se:WellKnownName>
            <se:Stroke>
              <se:SvgParameter name="stroke">#17e800</se:SvgParameter>
              <se:SvgParameter name="stroke-width">1</se:SvgParameter>
            </se:Stroke>
          </se:Mark>
          <se:Size>30</se:Size>
        </se:Graphic>
      </se:GraphicFill>
    </se:Fill>
  </se:PolygonSymbolizer>
  <se:PolygonSymbolizer>
    <se:Fill>
      <se:GraphicFill>
        <se:Graphic>
          <se:Mark>
            <se:WellKnownName>shape://backslash</se:WellKnownName>
            <se:Stroke>
              <se:SvgParameter name="stroke">#17e800</se:SvgParameter>
              <se:SvgParameter name="stroke-width">1</se:SvgParameter>
            </se:Stroke>
          </se:Mark>
          <se:Size>30</se:Size>
        </se:Graphic>
      </se:GraphicFill>
    </se:Fill>
  </se:PolygonSymbolizer>
  <se:PolygonSymbolizer>
    <se:Stroke>
      <se:SvgParameter name="stroke">#17e800</se:SvgParameter>
      <se:SvgParameter name="stroke-width">2</se:SvgParameter>
      <se:SvgParameter name="stroke-linejoin">bevel</se:SvgParameter>
    </se:Stroke>
  </se:PolygonSymbolizer>
</se:Rule>
```

Visual representation



As it's possible to read, there were used also another type of <Mark> values for the <wellknownName> element. There was used a new set of symbols given by the prefix "shape://". This set could have the values of: verline, horline, slash, backslash, dot, plus, times, oarrow, and carrow.

2.4 Additional Styling Properties

2.4.1 Line Break in Legend

To create a line break in the legend, this sequence of characters "**1
**" can be used.

```
code
<Rule>
  <Name>Situação Potencial:&#10;Afundamento de Navios</Name>
  <MaxScaleDenominator>25000</MaxScaleDenominator>
  <PolygonSymbolizer>
```

Legend representation

Preview legend

Situação Potencial:
Afundamento de Navios

3. Filters

The <Filter> element and its sub-elements enable conditional styling based on attribute values or geometric attributes.

Since this element is not described under the SE and SLD schemas, it is necessary to add the OGC schema to the code schema declarations, or alternatively, it can be named in each element namespace.

In SIGMAR symbologies, the filter element was only used to perform binary comparisons considering the values in the attribute table.

In the following code example, the <Filter> element is used with the <PropertyIsEqualTo> element, which encapsulates two mandatory sub-elements: <PropertyName> and <Literal>. The rule containing this <Filter> element will only render symbols for features that have the literal value "Marina dentro da Área de Intervenção do PSOEMA" in the "tipologia" field of their attribute table.

```
<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>tipologia</ogc:PropertyName>
    <ogc:Literal>Marina dentro da Área de Intervenção do PSOEMA</ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>
```

Note that in Geoserver SLD symbologies, only lowercase letters or numbers can be used. Even if the field contains uppercase letters, they must be written in lowercase in the code.

By combining different rules with different filter elements, it is possible to classify different elements of the same layer with distinct symbology.

Next, it's possible to find two <rule> elements each one used to symbolize different features based in the "classport" field from the same attribute table.




code

```
<se:Rule>
  <se:Name>Porto_Classe_A</se:Name>
  <se:Description>
    <se:Title>Porto de Classe A</se:Title>
  </se:Description>
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>classport</ogc:PropertyName>
      <ogc:Literal>A</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
  <se:PointSymbolizer>
    <se:Graphic>
      <se:Mark>
        <se:WellKnownName>circle</se:WellKnownName>
        <se:Fill>
          <se:SvgParameter name="fill">#e52121</se:SvgParameter>
        </se:Fill>
        <se:Stroke>
          <se:SvgParameter name="stroke">#000000</se:SvgParameter>
          <se:SvgParameter name="stroke-width">0.4</se:SvgParameter>
        </se:Stroke>
      </se:Mark>
      <se:Size>12</se:Size>
    </se:Graphic>
  </se:PointSymbolizer>
</se:Rule>

<se:Rule>
  <se:Name>Porto_Classe_B</se:Name>
  <se:Description>
    <se:Title>Porto de Classe B</se:Title>
  </se:Description>
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>classport</ogc:PropertyName>
      <ogc:Literal>B</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
  <se:PointSymbolizer>
    <se:Graphic>
      <se:Mark>
        <se:WellKnownName>circle</se:WellKnownName>
        <se:Fill>
          <se:SvgParameter name="fill">#e58a21</se:SvgParameter>
        </se:Fill>
        <se:Stroke>
          <se:SvgParameter name="stroke">#000000</se:SvgParameter>
          <se:SvgParameter name="stroke-width">0.4</se:SvgParameter>
        </se:Stroke>
      </se:Mark>
      <se:Size>12</se:Size>
    </se:Graphic>
  </se:PointSymbolizer>
</se:Rule>
```


Legend representation

Preview legend

-  Porto de Classe A
-  Porto de Classe B
-  Porto de Classe C

4. Adaptative Scale / Zoom

In addition of using filters to conditional represent elements of a feature, it is also possible to conditionally control their representation based on the scale of the map viewer.

The <MinScaleDenominator> element defines the denominator value of the scale below which the symbology will not be displayed.

The <MaxScaleDenominator> element defines the denominator value of the scale above which the symbology will not be displayed.

Both elements can be used individually or in combination, allowing the definition of one or more scale ranges.

Here is an example of their combined usage, where points are represented for smaller scales, and polygons are represented for larger scales. There is also a scale interval where both are represented on the map.

```

<se:Rule>
  <se:Name>Sit_Potencial_Afundamento_Navios_poligono</se:Name>
  <se:Description>
    <se:Title>Situação Potencial:&#10;Afundamento de Navios</se:Title>
  </se:Description>
  <se:MaxScaleDenominator>25000</se:MaxScaleDenominator>
  <se:PolygonSymbolizer>
    <se:Fill>
      <se:SvgParameter name="fill">#58cc7b</se:SvgParameter>
      <se:SvgParameter name="fill-opacity">0.35</se:SvgParameter>
    </se:Fill>
    <se:Stroke>
      <se:SvgParameter name="stroke">#00FF00</se:SvgParameter>
      <se:SvgParameter name="stroke-width">1</se:SvgParameter>
      <se:SvgParameter name="stroke-linejoin">bevel</se:SvgParameter>
    </se:Stroke>
  </se:PolygonSymbolizer>
</se:Rule>
<se:Rule>
  <se:Name>Sit_Potencial_Afundamento_Navios_ponto</se:Name>
  <se:Description>
    <se:Title>Situação Potencial:&#10;Afundamento de Navios</se:Title>
  </se:Description>
  <se:MinScaleDenominator>10000</se:MinScaleDenominator>
  <se:PointSymbolizer>
    <se:Graphic>
      <se:Mark>
        <se:WellKnownName>square</se:WellKnownName>
        <se:Fill>
          <se:SvgParameter name="fill">#58cc7b</se:SvgParameter>
        </se:Fill>
        <se:Stroke>
          <se:SvgParameter name="stroke">#00FF00</se:SvgParameter>
          <se:SvgParameter name="stroke-width">2</se:SvgParameter>
        </se:Stroke>
      </se:Mark>
      <se:Size>10</se:Size>
    </se:Graphic>
  </se:PointSymbolizer>
</se:Rule>

```

This configuration ensures that when zoomed out to a maximum scale denominator of 25,000, polygons are represented by point symbols at their centroids. As the zoom level increases to a minimum scale denominator of 10,000, the polygon representation becomes visible. It's important to note that the legend of the layer should reflect this behavior.

5. Conclusion and References

This documentation provided an overview of the SLD styling using SE encoding and some OGC filtering properties used in the SIGMAR geoserver geographic data services. It covered the common properties for points, polygons, and lines, as well as additional styling properties, filters, adaptive scale/zoom, and other general guidelines. By leveraging the SLD language, users can create visually appealing and informative maps tailored to their specific requirements.

5.1 References

<https://docs.geoserver.org/stable/en/user/styling/sld/reference/index.html>

<http://opengeospatial.github.io/e-learning/visualization-standards/basic-index.html>

<https://www.ogc.org/standard/sld/>

<https://www.ogc.org/standard/se/>

<https://www.ogc.org/standard/filter/>