

GeOxygene - 'Semiology' Plugin

Vo.4

Base model for works on map legends in GeOxygene

Abstract	<p>This document describes the module 'semiology' of GeOxygene dedicated to the works on map legends.</p> <p>A base model is proposed to represent the legend of a map as a tree. This model describes symbolization and allows manipulating map features symbolised with those symbols. More originally, this model describes metadata on the legend, such as semantic relationships between map layers (order, association or difference) related to Bertin's work [B67].</p> <p>A first application based on this model is provided. This application modifies a map legend in order to improve visual contrasts between symbolised features on a map.</p> <p>Extensions of this module are under development at COGIT lab.</p>
Authors	Sébastien Mustière and Charlotte Hoarau
Reviewers	Sidonie Christophe, Mickaël Brasebin, Elodie Buard, Laurence Jolivet

Evolution of the document

Version	Date	Modifications
V0.1	15/04/10	Description of the model for representing legends Description of the process of color contrasts improvement
V0.2	01/11	Redaction of the How-To use the contrast improvement application
V0.3	03/08/11	Adding a table of contents. Expanding the chapter “2. A model for representing legends” Redaction of chapter 4
V0.4	08/11	Re-organization of the document Corrections, Finalizations.

Table of contents

Evolution of the document	2
Foreword and References	4
Introduction	5
1. A model of legends	5
1.1. Overview	5
1.2. Package legendContent	8
1.3. Package mapContent	9
1.4. Package symbol	11
1.5. Package metadata	12
1.6. Foreseen extensions	12
2. Chromatic wheels and Color Contrasts	13
2.1. The COGIT Reference Color System	13
2.2. The package Color	14
3. Application: Contrast improvement	15
3.1. Description of the process	15
3.1.1. Preliminary remarks	15
3.1.2. Step 1: Evaluation of contrasts	16
3.1.3. Step 2: Change of color for all 'order' relationships	16
3.1.4. Step 3: Change of color of the least contrasted feature collection	16
3.2. Implementation: Package improvement	16
4. How to?	17
4.1. Semiology plugin GUI	17
4.1.1. Package gui	17
4.1.2. Loading the Semiology plugin	18
4.1.3. Menus and Toolbars	19
4.2. Loading data	20
4.3. How to visualize the chromatic wheels?	21
4.4. How to explore color contrasts?	21
4.5. How to Build a Legend tree?	22
4.5.1. Adding a theme to the legend tree	23
4.5.2. Removing an element of the legend tree	23
4.5.3. Adding all the layers to the legend tree	24
4.5.4. Validating the legend tree	24
4.6. Specifying the semantic relations	24
4.6.1. Creating semantic relationships	25
4.6.2. Removing a semantic relation	26
4.6.3. Validating semantic relations	26
How to	26
4.7. launch the contrast improvement process?	26
Annexes	27
Annex 1: Legend tree xml format	28
Annex 2: Semantic relations xml format	30
Bibliography	33
Figures Table	33

Foreword and References

The 'semiology' module originates from various works at COGIT lab. Most of the module originates from the works of Elodie Buard and Elisabeth Chesneau on the improvement of contrasts between symbolised features in maps. Other works on map legends have influenced the module, even if the related codes are not yet included in this version of the module.

Main publications directly linked to the module:

- Buard E., Ruas A., 2009, *Processes for improving the colours of topographic maps in the context of Map-on-Demand*, 24th International Cartographic Conference (ICC'09), 15-21 november, Santiago (Chile)
- Buard E., Ruas A., 2007, *Evaluation of colour contrasts by means of expert knowledge for on-demand mapping*, 23rd International Cartographic Conference (ICC'07), 4-10 August, Moscow (Russia)
- Chesneau E., 2007, *Improvement of Colour Contrasts in Maps : Application to Risk Maps*, 10th International Conference on Geographic Information Science (AGILE'07), 8-11 may, Aalborg (Denmark)
- Chesneau E., 2007, *Etude des contrastes de couleur pour améliorer la lisibilité des cartes*, 8èmes rencontres ThéoQuant, 10-12 janvier, Besançon (France)
- Chesneau E. 2006. *Modèle d'amélioration automatique des contrastes de couleur en cartographie, Application aux cartes de risques*. PhD, Univ. Marne-La-Vallée.

Other related works on legends in COGIT Lab:

- Jolivet L., 2009, *Characterizing maps to improve on-demand cartography - the example of European topographic maps*, 17th Conference on GIScience and Research in UK (GISRUK'09), 1-3 April, Durham (UK)
- Christophe S. 2009, *Aide à la conception de légendes personnalisées et originales: Proposition d'une méthode coopérative pour le choix des couleurs*. PhD, Univ. Paris-Est.
- Christophe S., 2009, Making legends by means of painters' palettes, in Cartwright W., Gartner G., Lehn A., Cartography and Art. Lecture Notes in Geoinformation and Cartography series, Springer
- Dominguès C., Christophe S., Jolivet L., 2009, *Connaissances opérationnelles pour la conception automatique de légendes de cartes*, 20èmes Journées Francophones d'Ingénierie des Connaissances (IC2009), 25-29 mai, Hammamet (Tunisie)
- Christophe S., Ruas A., 2009, *A process to design creative legend on-demand*, 24th International Cartographic Conference (ICC'09), 15-21 november, Santiago (Chile)
- Christophe S., Bucher B., 2009, *Un modèle de dialogue pour la conception de légendes cartographiques*, 27ème Congrès INFORSID, 26-29 Mai, Toulouse (France)
- Dominguès C., 2008, *Description de cartes géographiques*, 8èmes journées francophones Extraction et gestion des connaissances (EGC2008) - Atelier sur la modélisation utilisateur et la personnalisation d'interfaces web, 29 janvier - 1er février, Sophia-Antipolis (France)
- Jolivet L., 2008, *On-demand map design based on user-oriented specifications*, 17th AutoCarto conference (AutoCarto'08), 8-11 september, Shepherdstown (USA)
- Bucher B., Jolivet L., Buard E., Ruas A., 2007, *The need for Web legend services*, in J.M.Ware and G.E.Taylor (Eds), 7th International Symposium on Web and Wireless GIS (W2GIS), Springer Lecture Note in Computer Science, 28-29 November, Cardiff (UK)

Introduction

This document describes the module ‘semiology’ of GeOxygene dedicated to the works on map legends.

A base model is proposed to represent the legend of a map as a tree. This model describes symbolization and allows manipulating map features symbolised with those symbols. More originally, this model describes metadata on the legend, such as semantic relationships between map layers (order, association or difference) related to Bertin’s work [B67].

A first application based on this model is provided. This application modifies a map legend in order to improve visual contrasts between symbolised features on a map.

Chapter 2 describes the model of legends, chapter 3 describes the COGIT representation of colors, chapter 4 describes the application of contrasts improvement, and finally chapter 5 provides a user guide.

1. A model of legends

1.1. Overview

In the proposed framework (see Figure 1), the legend is thought of as a mediator between what has to be represented (which features?), how they are represented (which signs?), and what is the intended/perceived relations between represented features (which message?).

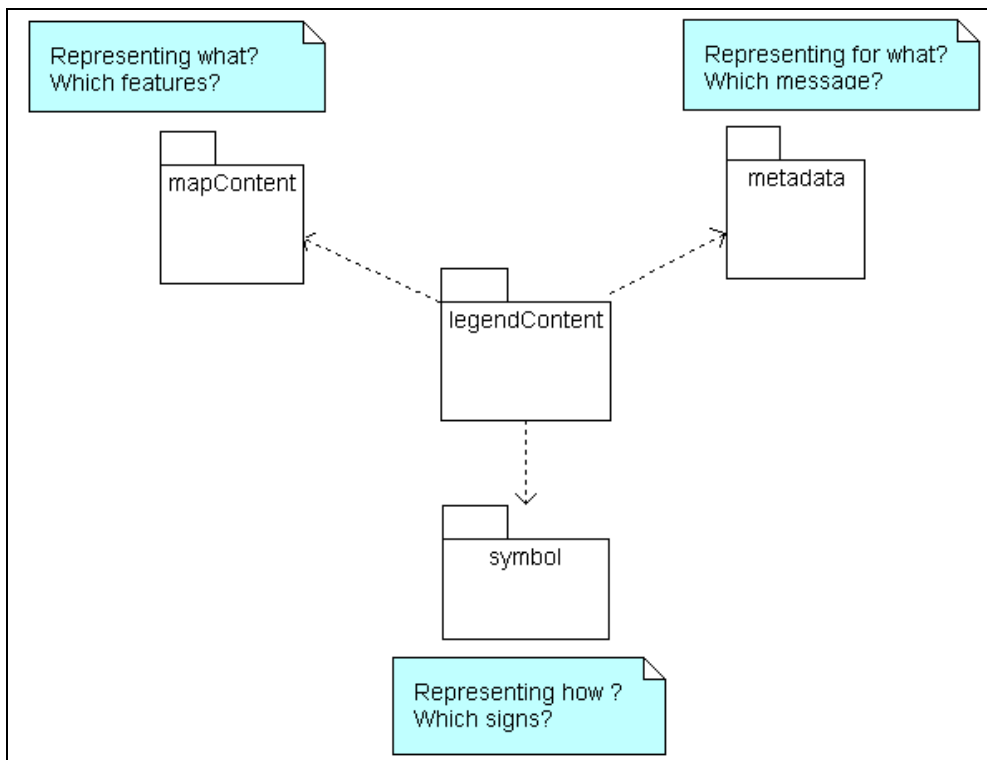


Figure 1: Overview of the model

This theoretical framework is organized in four main packages:

The main package is `legendContent`: it describes which components are contained in the map legend of the map (names and symbol choices of each line of the key), and how they are organized (which lines are gathered in a theme).

The package `mapContent` describes, for a given map, the set of features that are to be represented (`features` organized in `feature collections`).

The package `symbol` describes the symbols used to draw features on the map (color, size...).

The package `metadata` describes messages to be conveyed by the map (concretely restricted to relations between set of symbolised features for the time being, like 'association', 'difference' or 'order'). It could be extended after time (Cf. paragraph 1.6 Foreseen extensions).

Main elements of the model are described in Figure 2 and detailed hereafter.

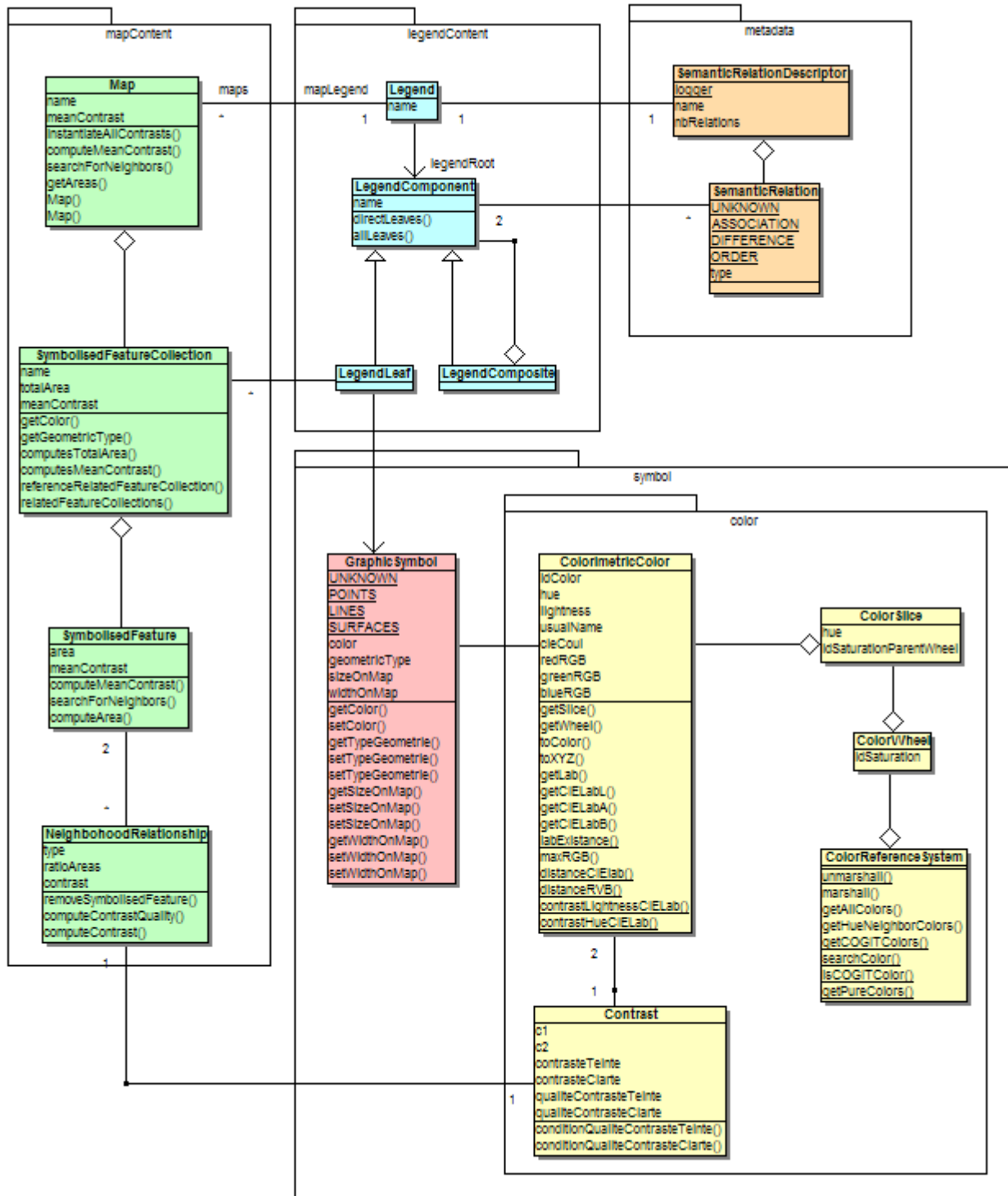


Figure 2. Main classes of the model

1.2. Package legendContent

This package models the legend as a tree of components, it will be called “legend tree” in this document. The pattern ‘recursive aggregation’ (Cf. [GHJV95]) is used to model it as shown in Figure 3: Package legendContentFigure 3:

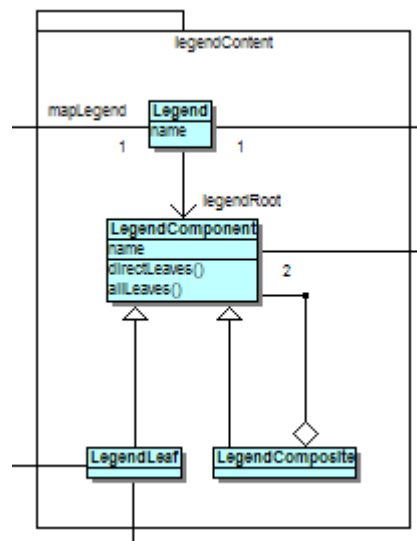


Figure 3: Package legendContent

A **LegendComponent** can be a **LegendComposite** or a **LegendLeaf**.

A **LegendComposite** can contain **LegendComponents**, e.g. sub-**LegendComposites** or **LegendLeafs**. It has a name (e.g. ‘Roads’); it groups thematically **LegendComponents**: in cartography and in this document, we call it a theme.

A **LegendLeaf** contains the same information as the one found on a line of the key of the map. It has a name (e.g. ‘Highways’ or ‘Main Roads’). It is related to a set of objects to display on the map with the same symbol choices (e.g. the set of features that are ‘B roads’ on the mapped area); and is related to a description about the symbol itself (e.g. ‘orange line’).

The class **Legend** provides the methods to store/load legend trees in xml files (Cf. Annex 1: Legend tree xml format).

1.3. Package mapContent

This package models the features to be represented on a given map. A `Map` is a set of `SymbolisedFeatureCollections` (e.g. all buildings), which are sets of `SymbolisedFeatures` (e.g. one particular building).

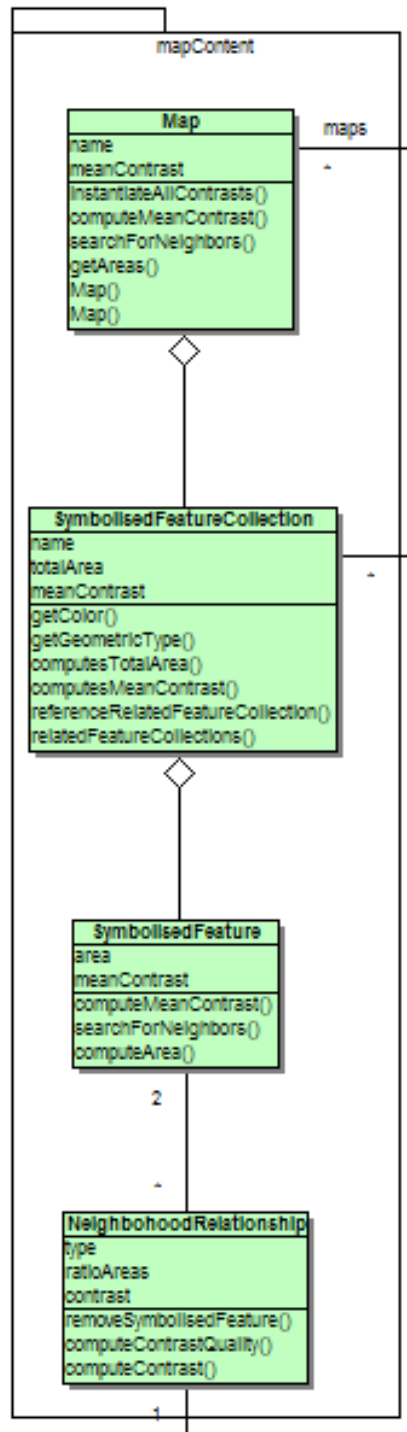


Figure 4: Package `mapLegend`

`SymbolisedFeatures` are geographic features depicted on the map: their area (computed by the specific method `computeArea()` and available by the method `getArea()`) does not refer to the geographic area but rather to the covered area on the map.

The `SymbolisedFeatureCollection` class has also a `totalArea` attribute which refers to the sum of the covered area by all the `SymbolisedFeature` of the collection. This attribute can be computed by the method `ComputeTotalArea()`. A specific method is accessible for the linear `SymbolisedFeatureCollection`, `computeLinesArea()`, which first aggregates all the objects of the collection and then calculates the covered area of this aggregation in order to avoid superposed sections.

`NeighborhoodRelationship` is a class that explicitly represents spatial relations between two features close enough to be considered as neighbors on the map. This class is important for evaluating the quality of contrasts between close objects on the map. `NeighborhoodRelationship` can be computed by a method of the class `Map`, `searchForNeighbors()`.

All these classes contain specific methods to calculate color contrasts between features. Contrasts are modeled in the package `color` (Cf. paragraph 2.2 The package `color`). They are first calculated for each `NeighborhoodRelationship` and then aggregated for each `SymbolisedFeature`, each `SymbolisedFeatureCollection` and finally for the entire map. This global mechanism is available by the method `instanciateAllContrasts()` of the class `Map`. It is possible to weigh these contrasts aggregation using the method `instanciateAllWeightedContrasts()`:

- For each `SymbolisedFeature`, contrasts will be weighted by the covered area of its neighbors.
- For each `SymbolisedFeatureCollection`, contrasts will be weighted by the covered area of the `SymbolisedFeature` of the collection.

1.4. Package `symbol`

This package models the graphical symbols that can be used in the map. A `GraphicSymbol` describes a style object with a cartographic point of view. Its attributes match with the visual variables described in [B67]: size, shape, color, etc. The class provides some methods to instantiate these attributes from an SLD layer.

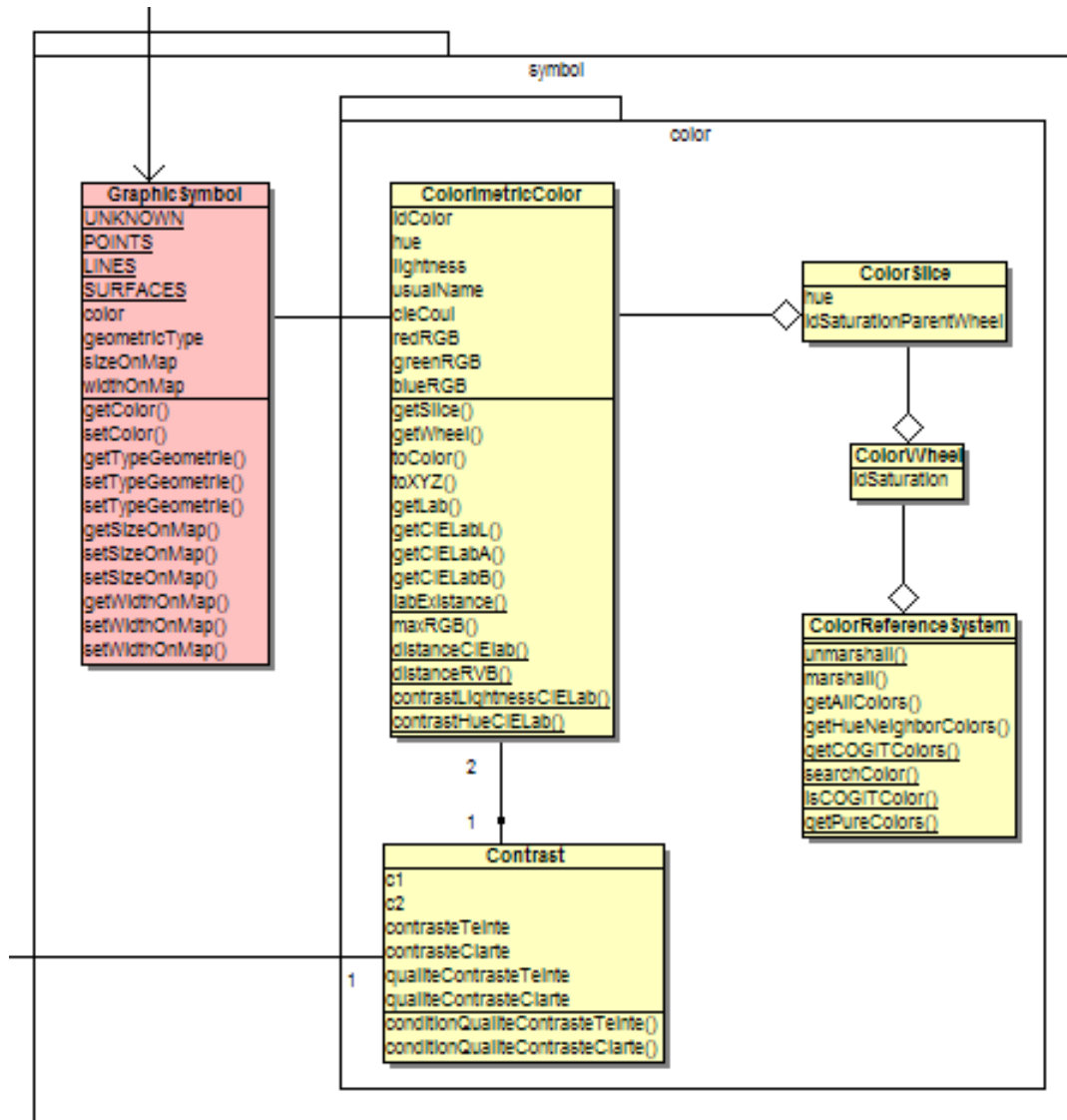


Figure 5: Package `symbol` and sub-package `color`

1.5. Package metadata

This package describes `SemanticRelations` between legend components that the map is supposed to convey. `SemanticRelations` of a legend are grouped in a `SemanticRelationDescriptor`, it can be used to store and/or load `SemanticRelations` in xml files (Cf. Annex 2: Semantic relations xml format).

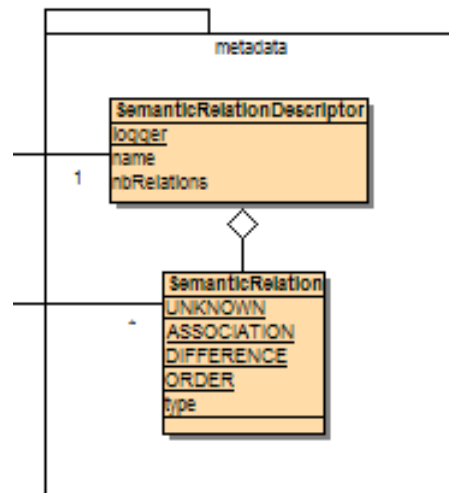


Figure 6: Package metadata

[B67] described four kinds of semantic relationships: association, differentiation, order and quantity. In the model, we consider association, differentiation and order relationships; the kind of relationship is stored in the `type` attribute. For example, it may be used to express that “‘Highways’ and ‘Main Roads’ are linked by an ‘order’ relationship”.

1.6. Foreseen extensions

The package `metadata` is supposed to contain other information describing the message to be conveyed by the map like the importance of elements (e.g. ‘this legend element should appear as a background or foreground element on the map’) or semantics of element (e.g. ‘this element represents hydrographical features’) for example.

A package is planned to model general cartographic knowledge, such as typical colors for some thematic phenomena (‘the water is usually displayed in blue’) or cartographic rules (‘hue is not an efficient variable to represent order between phenomena’). For the time being, such knowledge is used in the process that handles legends. Work is undergoing to externalize it.

Other processes for legend manipulation are already developed and may be added to the module.

2. Chromatic wheels and Color Contrasts

2.1. The COGIT Reference Color System

Navigation in the color space and evaluation of contrasts between colors rely on the chromatic wheels shown in Figure 7. For more details about the wheels and evaluation of contrasts in this wheel see:

Buard E., Ruas A., 2007, *Evaluation of colour contrasts by means of expert knowledge for on-demand mapping*, 23rd International Cartographic Conference (ICC'07), 4-10 August, Moscow (Russia)

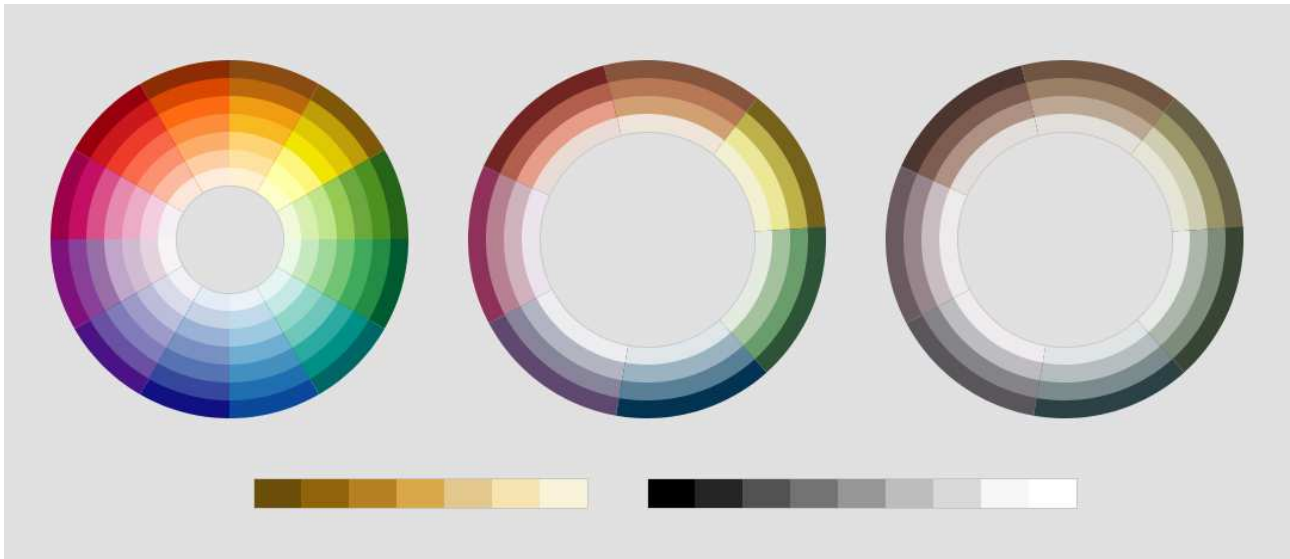


Figure 7 : COGIT chromatic wheels

2.2. The package `color`

The package `symbol` contains the package `color` which groups all the classes dedicated to the description of color and color contrasts.

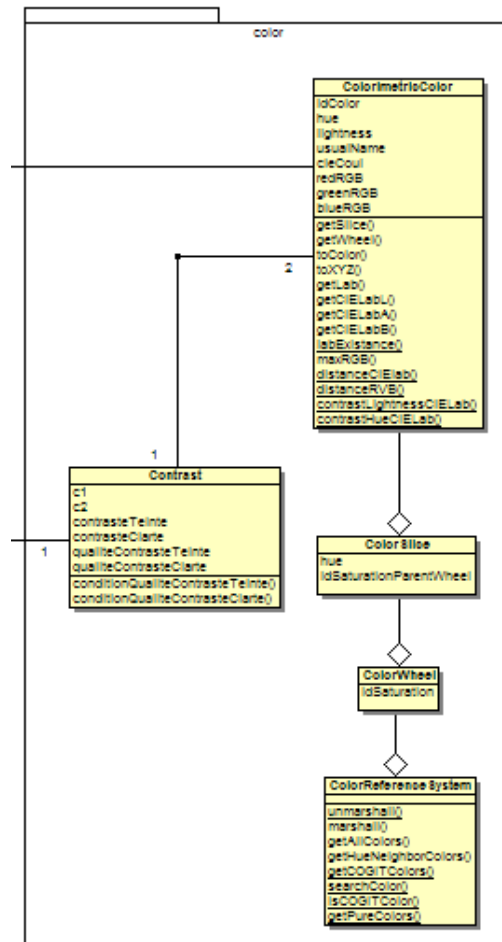


Figure 8 : Package `color`

A `ColorReferenceSystem` is a set of `ColorWheels` which are sets of `ColorSlices`, which are sets of `ColorimetricColors`. The `ColorimetricColor` contains methods to manipulate RVB, CIElab and COGIT color coordinates. The `ColorReferenceSystem` also provides methods to store and/or load `ColorimetricColors` organized in `ColorSlices` and `ColorWheels` in xml files. `ColorimetricColors` of the COGIT `ColorReferenceSystem` are stored as resources in the `ColorReferenceSystem.xml` file.

The class `Contrast` models a color contrast between two colors. Its attributes store the values of lightness contrast, hue contrast and the quality of these values regarding the semantic relationship between corresponding features. `Contrasts` are grouped in a `ContrastCollection`, it can be used to store and/or load `Contrasts` in xml files. `Contrasts` evaluated by COGIT works are stored in the `Contrasts.xml` files.

For more details on colors and contrasts, see:

Buard E., Ruas A., 2007, *Evaluation of colour contrasts by means of expert knowledge for on-demand mapping*, 23rd International Cartographic Conference (ICC'07), 4-10 August, Moscow (Russia)

3. Application: Contrast improvement

The goal of the contrast improvement process is to modify colors of a given legend in order to improve contrasts between objects on a given map. The global idea is that two features close enough on the map should have sufficiently different colors to be distinguished. Another idea guiding the process is that relations between colors ('same hue but darker', 'same lightness but different hue'...) should be consistent with relations between legend leaves that the map may convey (e.g. : colors of phenomena 'low risk', 'medium risk' and 'high risk' should have similar hue but different lightness to express an order relationship).

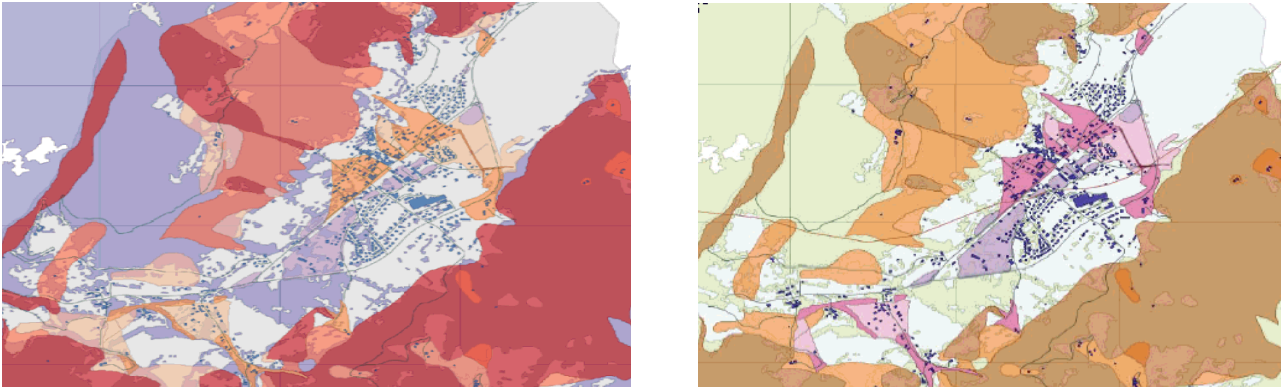


Figure 9: Typical result: map before (left) and after (right) contrast improvement

For more details about this functionality, see:

Buard E., Ruas A., 2009, *Processes for improving the colours of topographic maps in the context of Map-on-Demand*, 24th International Cartographic Conference (ICC'09), 15-21 november, Santiago (Chile)

Chesneau E., 2007, *Improvement of Colour Contrasts in Maps : Application to Risk Maps*, 10th International Conference on Geographic Information Science (AGILE'07), 8-11 may, Aalborg (Denmark)

3.1. Description of the process

3.1.1. Preliminary remarks

The process described here is supposed to deal with a legend associated with some relations (order, association, difference). However, some restrictions are applied to those relations in this process by comparison with the general model described in "chapter 1 A model of legends" :

- Relations are only expressed between legend leaves (not between legend composites).
- Each legend leaf participates only to one relation. Work is in progress to modify this restriction.

The process is iterative: it modifies some of the colors of the legend, with the goal to improve the contrast of features having worst contrasts with their neighbors on the map. This process may be repeated several times in order to reach good results.

3.1.2. Step 1: Evaluation of contrasts

All contrasts between close objects are computed. In order to evaluate contrasts between two close objects, a matrix of predefined contrast values between colors on a chromatic wheel is used (Cf. chapter 2 Chromatic wheels and Color Contrasts). More precisely, colors of the features are match to the closest color on the COGIT chromatic wheels. Hue and lightness contrasts are then determined between those colors of the wheels. The quality of hue and lightness contrasts is then evaluated regarding the type of relation between the related legend leaves. For example, if the lightness contrast is low but the relation is an 'order', then the quality of lightness contrast will be considered as bad. Similarly, if the lightness contrast is low but the relation is an 'association', then the quality of lightness contrast will be considered as good.

Means of contrasts for each feature, and for each feature collections, and for the entire map are then computed. These means can be weighted or not, by the cartographic area of the neighbors for each feature and by the cartographic area of the features for each feature collection.

For more details, on contrasts, see:

Buard E., Ruas A., 2007, *Evaluation of colour contrasts by means of expert knowledge for on-demand mapping*, 23rd International Cartographic Conference (ICC'07), 4-10 August, Moscow (Russia)

3.1.3. Step 2: Change of color for all 'order' relationships.

For each legend leaf participating to an order relationship, the hue of the leaves is changed in such a way that all leaves of a given relationship have the same hue (the hue of one of the leaves, randomly selected for the time being).

3.1.4. Step 3: Change of color of the least contrasted feature collection

The application finds the feature collection with the worst qualities of hue and lightness contrasts. And then, colors of this feature collection is changed (priority is given to changes of hues than changes of lightness).

3.2. Implementation: Package `improvement`

This package contains an interface and one implementation for contrasts analysis processes. Main methods for those contrasts analysis improvement are an initialization method `initialize()`, a method to improve the contrast of a map `improvementStep()` and a global method to manage the whole process `run()`.

This package also contains some `stopCriteria`, which are different strategies to stop the contrasts improvement process.

4. How to?

4.1. Semiology plugin GUI

4.1.1. Package *gui*

The package `gui` groups all the graphical user interface components needed to call semiotics tools and to launch the application of contrasts improvement:

- The class `SemioPlugin` is the main one. It has to be added to the `plugins.xml` file in order to make it available on the main Geoxylene application.
- The `SemioToolBar` dedicated to semiotics tools. It contains tree buttons: one to open the `SpecificationToolBar`, one to visualize the colors of the legend on the Chromatic Wheels, and one to open the `PaletteToolBar` and visualize a palette of the colors of the legend.
- The `SpecificationToolBar` contains tree tabs: one to specify a legend tree from the loaded data, one to specify semantic relationships, one to give information help concerning specifications.
- The `ContrastAnalysisToolBar` is dedicated to the application of contrasts improvement.
- The `ContrastFrame` allows visualizing the Chromatic Wheels and exploring contrasts of hue and lightness between colors. It uses the `ColorRenderer` class to visualize color in the results table.
- The `PaletteToolBar` allows the visualization of color palettes below the map.

4.1.2. Loading the Semiology plugin

The semiology plugin is available as a GeOxygene plugin from the Geoxygene main interface. In order to load it, this line code must be added to the file `plugins.xml` (at the root of the GeOxygene project):

```
<plugin>gui.SemioPlugin</plugin>
```

Then, a specific menu will be available on the Geoxygene main frame:

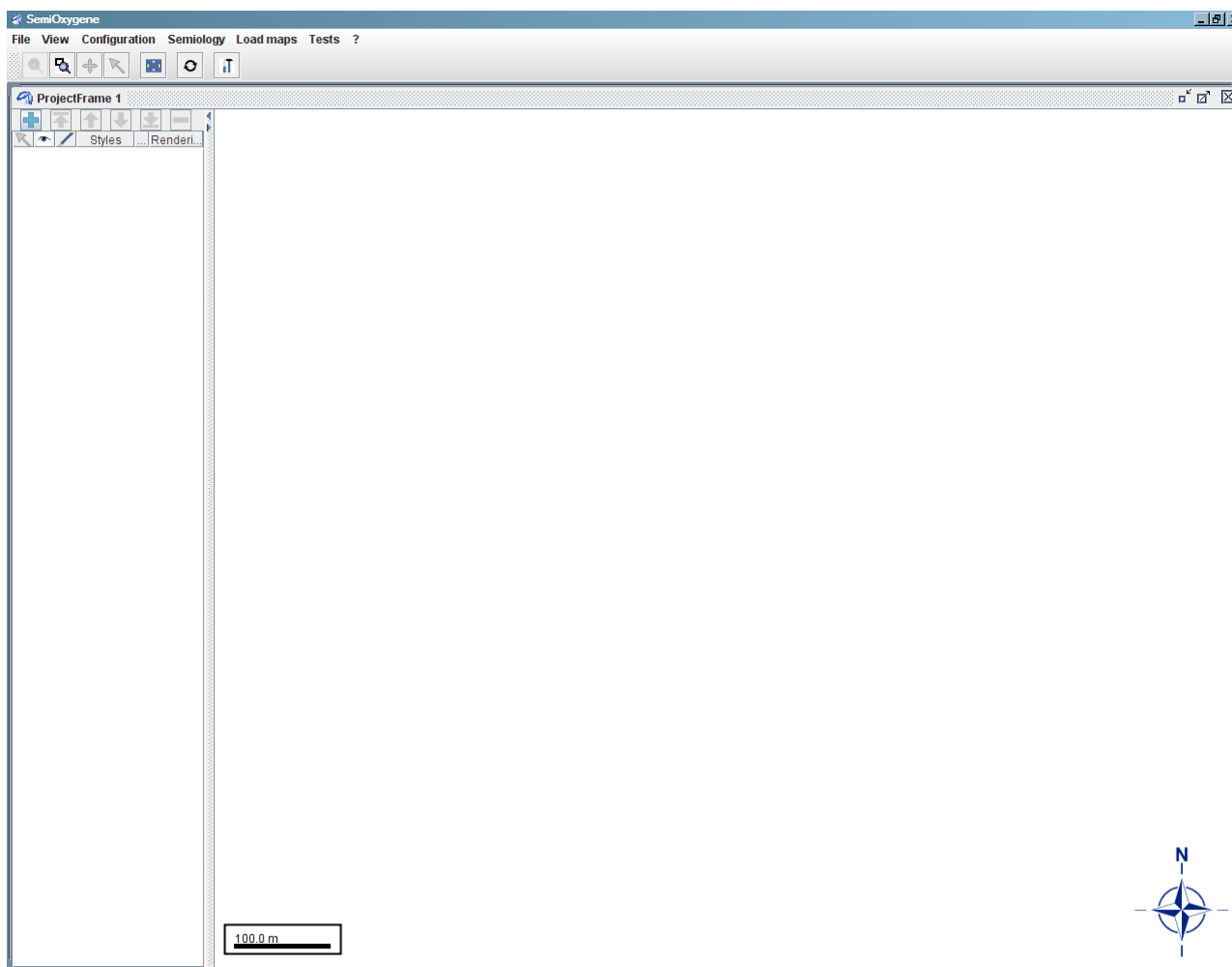


Figure 10 : Semiology menu in the GeOxygene main frame

4.1.3. Menus and Toolbars

The semiology menu provides items to open toolbars and applications:

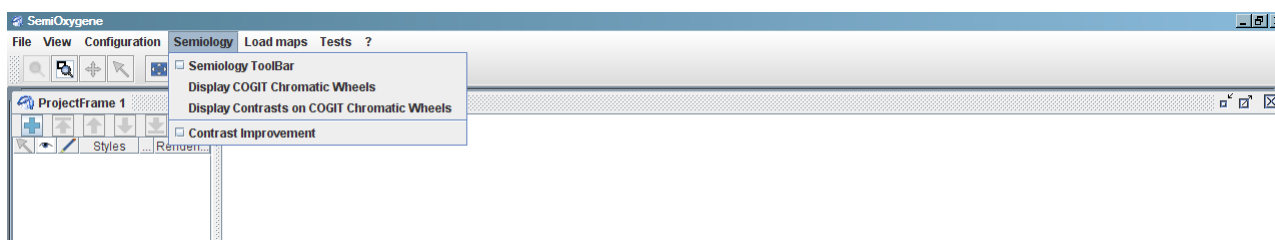


Figure 11 : Items of the semiology menu

- The **Semiology ToolBar** item opens a toolbar with tree tools:

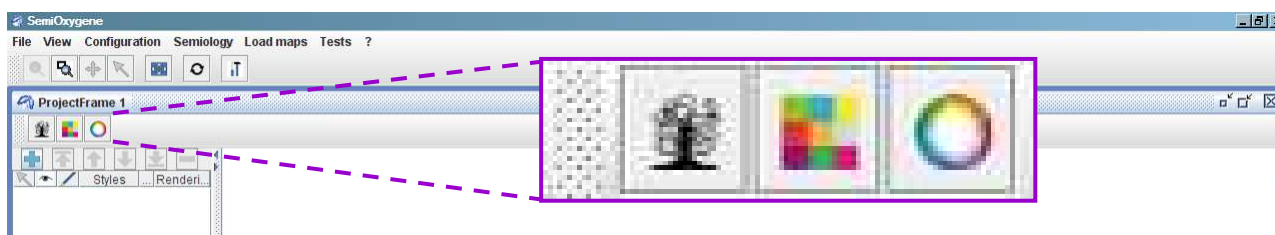


Figure 12 : Semiology toolbar in the GeOxygene main frame

- The first one opens the specification toolbar for the creation of a legend tree from loaded layers (Cf. paragraph 4.5 How to Build a Legend tree?)
 - The second one, for the visualization of colors palette from loaded layers
 - The last one for the visualization of the colors of loaded layers on the chromatic wheels.
- The **Display COGIT Chromatic Wheels** item opens a window showing the chromatic wheels (Cf. paragraph 4.3 How to visualize the chromatic wheels?).
 - The **Display Contrasts on COGIT Chromatic Wheels** item opens a window to explore the contrasts between colors of the chromatic wheels (Cf. paragraph 4.4 How to explore color contrasts?)
 - The **Contrast Improvement** item opens a toolbar dedicated to the contrast improvement application (Cf. paragraph 4.7 How to launch the contrast improvement process?) :

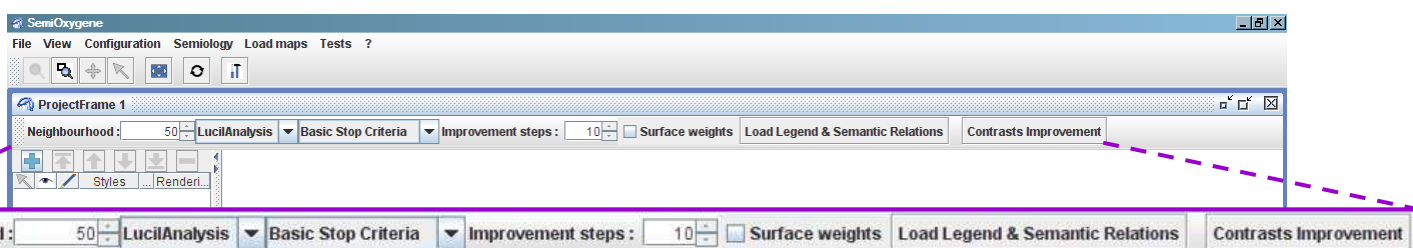



Figure 13 : Contrast improvement toolbar in the GeOxygene main frame

4.2. Loading data

Data must first be loaded and symbolised in the main frame. Data can be loaded with the button  in the main frame or with some adapted programming. They can be symbolised in the main frame using the Style Edition Frame (right click on the layer and choose the “Edit style” button) or loading an SLD file with some programming.

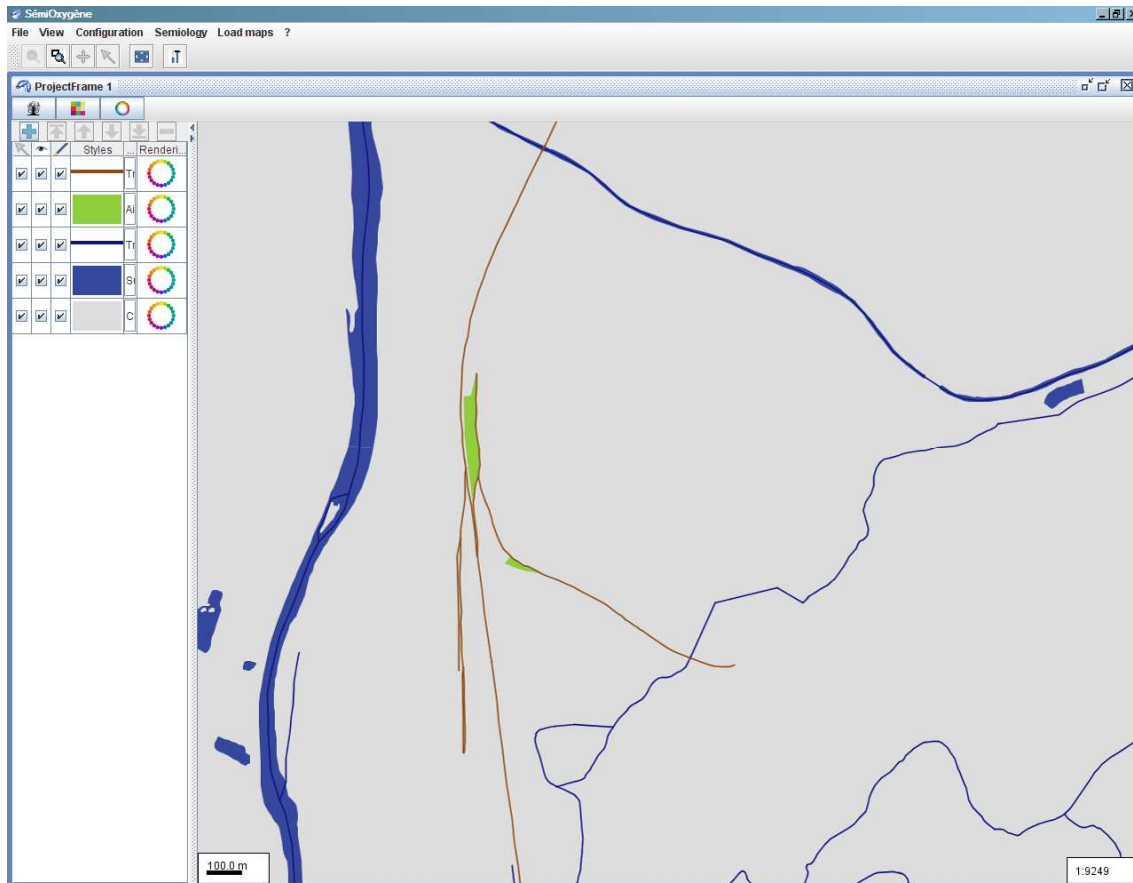


Figure 14 : Data loaded in the GeOxygene main frame

For further information about loading and symbolising data, see the general documentation of GeOxygene.

4.3. How to visualize the chromatic wheels?

To visualize the chromatic wheels in the GeOxygene application, you need to load the Semiology plugin (Cf. paragraph 4.1.2 Loading data) and click on the menu “Semiology/Display COGIT Chromatic wheels”. The following frame will appear:



Figure 15 : Chromatic wheels, the COGIT color reference system

By clicking on a color, the result table provides information about the selected color concerning the COGIT color reference system (the name of the color and its color key) and other color coordinates systems (RGV, sRGB, CIE Lab and hexa-code).

4.4. How to explore color contrasts?

Such as previously, to explore COGIT hue and lightness contrasts, you need to load the Semiology plugin (Cf. paragraph 4.1.2 Package gui

The package `gui` groups all the graphical user interface components needed to call semiotics tools and to launch the application of contrasts improvement:

- The class `SemioPlugin` is the main one. It has to be added to the `plugins.xml` file in order to make it available on the main Geoxygene application.
- The `SemioToolBar` dedicated to semiotics tools. It contains tree buttons: one to open the `SpecificationToolBar`, one to visualize the colors of the legend on the Chromatic Wheels, and one to open the `PaletteToolBar` and visualize a palette of the colors of the legend.
- The `SpecificationToolBar` contains tree tabs: one to specify a legend tree from the loaded data, one to specify semantic relationships, one to give information help concerning specifications.
- The `ContrastAnalysisToolBar` is dedicated to the application of contrasts improvement.

- The `ContrastFrame` allows visualizing the Chromatic Wheels and exploring contrasts of hue and lightness between colors. It uses the `ColorRenderer` class to visualize color in the results table.
- The `PaletteToolBar` allows the visualization of color palettes below the map.

Loading the Semiology plugin) and click on the menu “Semiology/Display Contrasts on COGIT Chromatic wheels”. The following frame will appear:

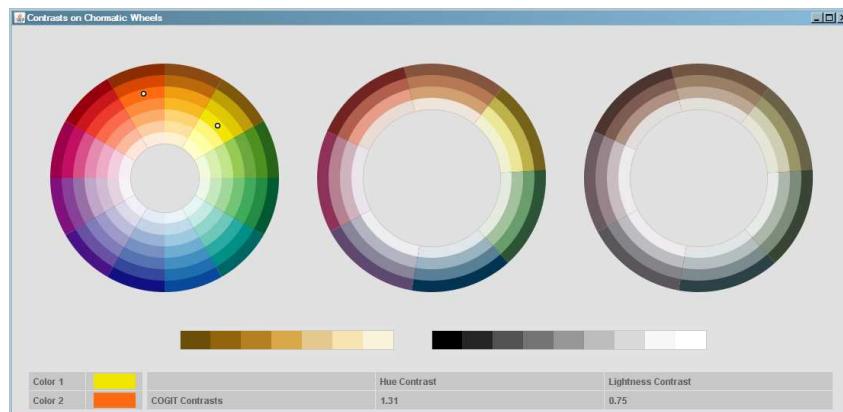



Figure 16 : Exploring color contrasts on the COGIT chromatic wheels

If two colors are selected (click on the chromatic wheel to select them), the result table provides the values of the corresponding hue and lightness contrasts. Then you can unselect the colors and go on in the exploration of other color contrasts.

4.5. How to Build a Legend tree?

As detailed in paragraph 1.2, the legend tree can be described with Java objects of the package `legendContent`. A user interface is available to build such a tree from loaded data in GeOxygene application.

Click on the button  of the semiology toolbar to open this interface. A toolbar will appear on the right side of the main application interface, it has three tabs:

- The first one is dedicated to the build of the legend tree,
- The second one is dedicated to the specification of semantic relationships (Cf. paragraph 4.6 How to specify the semantic relations?),
- The last one gives some help and concept explanations.

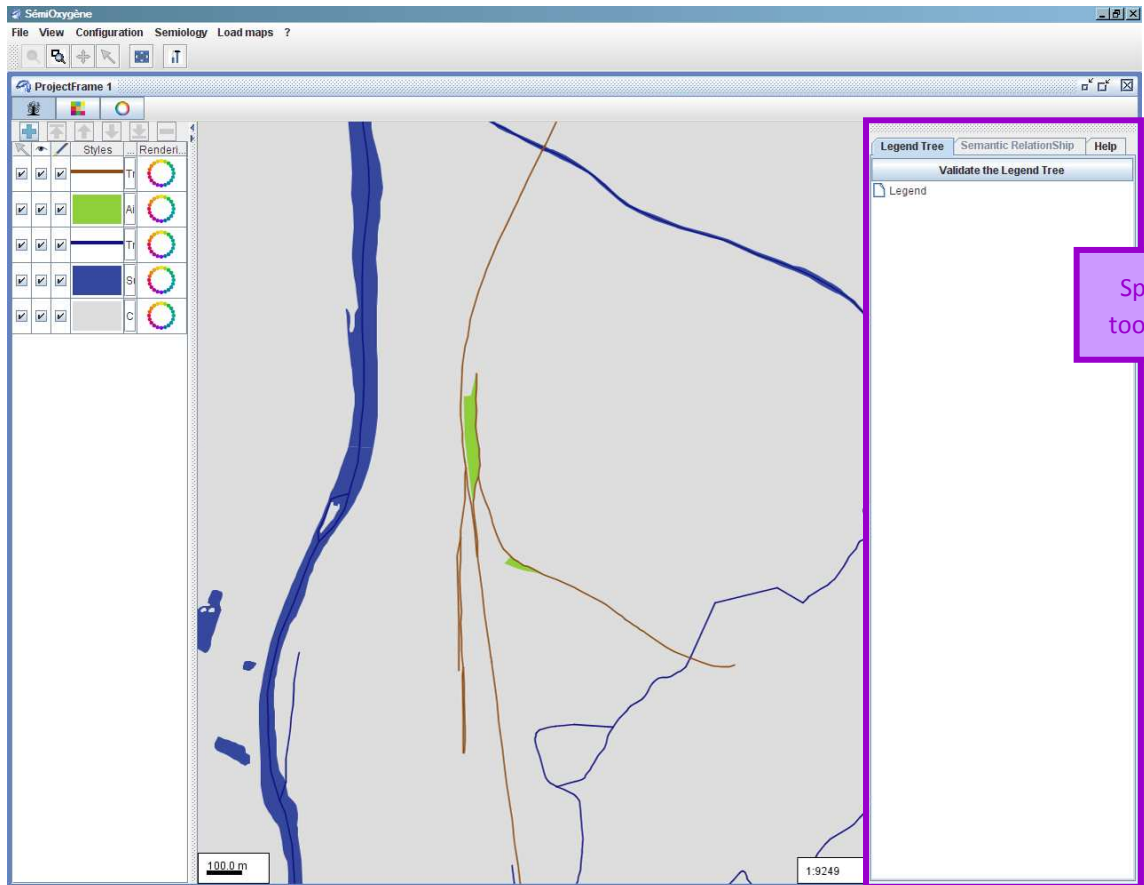


Figure 17 : Specification toolbar loaded in the main frame.

4.5.1. Adding a theme to the legend tree

To create a new theme in the legend tree, right click on the parent of the future theme and choose the **Add a theme** item on the context menu. Then, type the name of your new theme, click on OK to validate the new theme. It has been added on the legend tree:



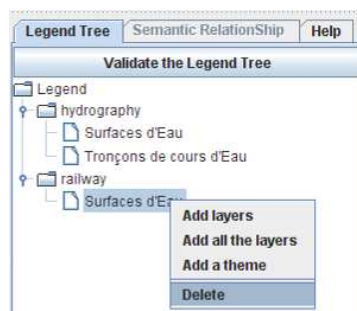
a) Adding layers to the legend tree

Once all the themes have been created, layers can be added to the different themes. Right click on the corresponding theme and choose the **Add layers** item on the context menu. A list of all the layers loaded on the GeOxygene application is available. Select the layers to be added to the theme. “Ctrl” and/or “Shift” keys can be used to select several layers at the same time. Then validate with the button OK. Selected layers have been added to the theme:



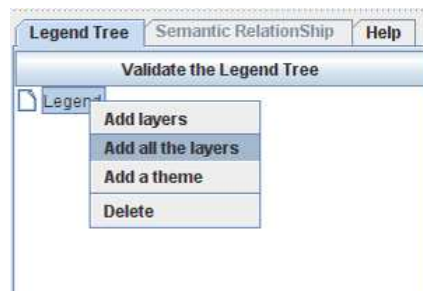
4.5.2. Removing an element of the legend tree

If there is a mistake in the build legend tree, layers or themes can be removed from the tree: right click on the incorrect element and choose the **Delete** item on the context menu.



4.5.3. Adding all the layers to the legend tree

If the structure of the legend is very simple and does not need themes to convey a hierarchy, all the loaded layers can be loaded at the same time: right click on the legend root and choose the **Add all the layers** item on the context menu:



4.5.4. Validating the legend tree

When the legend tree is finish, click on the button **Validate the Legend Tree** to validate them. Corresponding Java objects of the package `legendcontent` are created at this step.

4.6. How to specify the semantic relations?

Semantic relationships can be specified between the themes of the legend in the second tab of the toolbar:

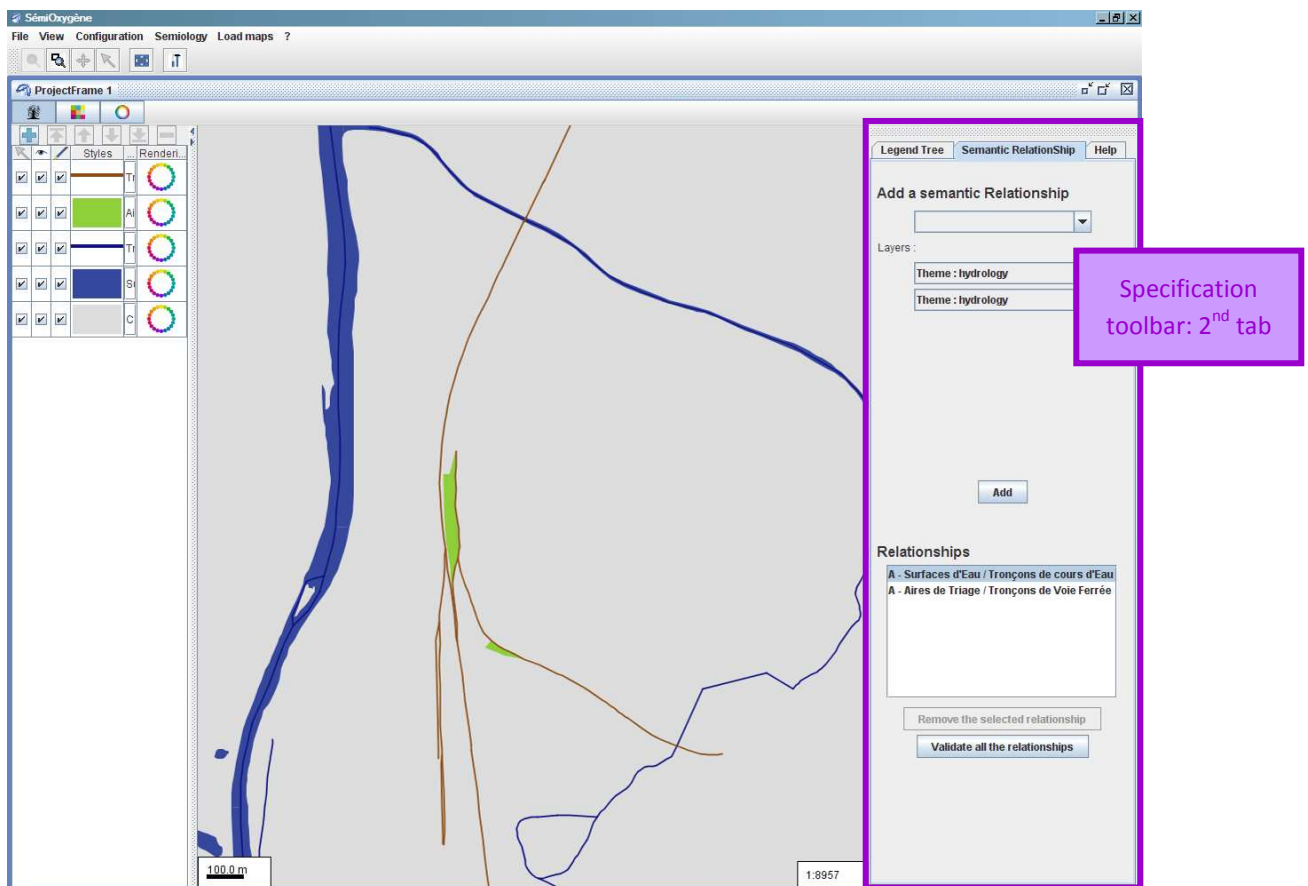


Figure 18 : Semantic relation tab in the specification toolbar

The legend leaves which have the same parent theme have been linked by default by an association relationship. They appear on the “Relationships” list at the bottom of the toolbar:



Moreover, in the “Layers” drop-down lists all the layers and the themes of the build legend tree are available to create other semantic relationships.

4.6.1. Creating semantic relationships

Semantic relationships give sense to the legend. [B67] described four kinds of semantic relationships: association, differentiation, order and quantity. Here, we consider association, differentiation and order relationships to improve the contrasts on the map.

a) Between layers

Creating a new semantic relationship between layers is simple:

- Select the type of relationship (association, differentiation, order) in the first drop-down list
- Select both layers in each “Layers” drop-down lists.
- Validate the relationship by clicking on the button .

Association and differentiation relationships link two layers.

Order relationships can link up to seven layers. The top drop-down list specifies the highest layer in an order relationship. The bottom drop-down list specifies the lowest layer in an order relationship.

b) Between themes

The specification of an association relationship between two themes creates an association relationship between all the leaves of each theme.


The specification of a differentiation relationship between two themes creates a differentiation relationship between all the leaves of each theme.

Currently, two themes can not be ordered.


c) Between layers and themes

Currently, layer and themes can not be linked by a relationship at the moment.

4.6.2. Removing a semantic relation

If an incorrect relationship has been created, select it on the “Relationships” list and click on the button  to remove it.

4.6.3. Validating semantic relations

When all the semantic relationships have been specified, click on the button  to validate them. The corresponding java objects of the package `metadata` are created at this step.

4.7. How to launch the contrast improvement process?

First the contrast analysis toolbar must be loaded on the GeOxygene application (Cf. paragraph 4.1.3 Menus and Toolbars):



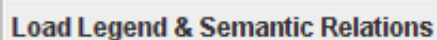
A spinner make possible to define the neighborhood radius used on the contrasts improvement application described in the paragraph 3.1 Description of the process. This radius is expressed in meter.

A combo list provides different contrast analysis to launch. Only one is available at the moment. Another combo list provides different stop criteria.

A checkbox makes possible to weight contrasts (as described in paragraph 1.3) or not.

A legend tree and a set of semantic relationships must be specified before launching the contrast improvement process. Two methods are available to do that:

- Using the specification toolbar (Cf. paragraph 4.5 and 4.6).
- It is also possible to store legend trees and semantic relations in xml files (Cf. annexes for the format). These configurations will be loaded using the button



Finally, click on the button  to launch the contrast improvement application

Annexes

Annexes 1 and 2 provide examples of legend tree and semantic relations xml file for the following map legend:

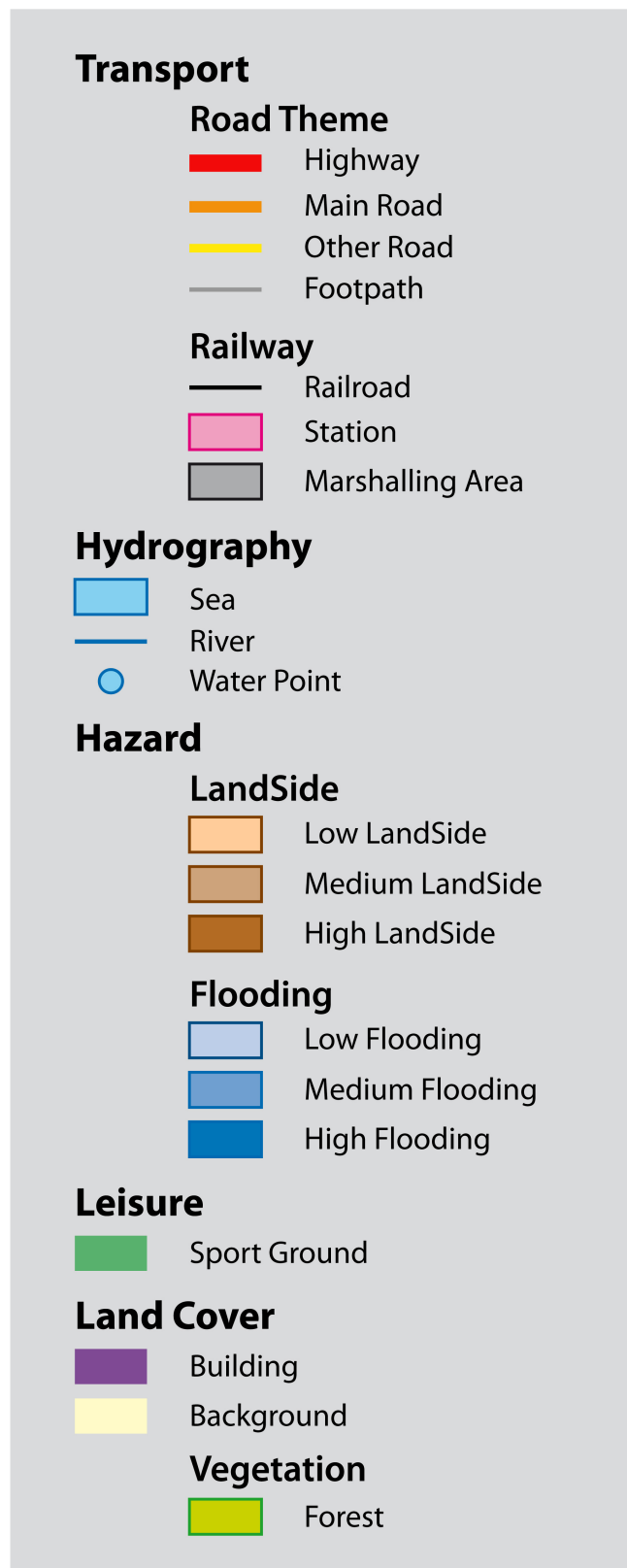


Figure 10: Legend example specified by the legend tree of Annex 1 and the semantic relationships of Annex 2.

Annex 1: Legend tree xml format

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Legend>
  <Name>Legend</Name>
  <legendRoot>
    <Name>LegendRoot</Name>
    <Component>
      <Name>Transport</Name>
      <Component>
        <Name>Road theme</Name>
        <Component>
          <Name>Footpath</Name>
        </Component>
        <Component>
          <Name>Other Road</Name>
        </Component>
        <Component>
          <Name>Main Road</Name>
        </Component>
        <Component>
          <Name>Highway</Name>
        </Component>
      </Component>
      <Component>
        <Name>Railway</Name>
        <Component>
          <Name>Marshalling area</Name>
        </Component>
        <Component>
          <Name>Railroad</Name>
        </Component>
        <Component>
          <Name>Station</Name>
        </Component>
      </Component>
    </Component>
    <Component>
      <Name>Hydrography</Name>
      <Component>
        <Name>Water point</Name>
      </Component>
      <Component>
        <Name>River</Name>
      </Component>
      <Component>
        <Name>Sea</Name>
      </Component>
    </Component>
    <Component>
      <Name>Hazard</Name>
      <Component>
        <Name>Landside</Name>
        <Component>
          <Name>Low Landside</Name>
        </Component>
        <Component>
          <Name>Medium Landside</Name>
        </Component>
        <Component>
          <Name>High Landside</Name>
        </Component>
      </Component>
    </Component>
  </legendRoot>
</Legend>
```

```
    <Name>Flooding</Name>
    <Component>
      <Name>Low Flooding</Name>
    </Component>
    <Component>
      <Name>Medium Flooding</Name>
    </Component>
    <Component>
      <Name>High Flooding</Name>
    </Component>
  </Component>
</Component>
<Component>
  <Name>Building</Name>
</Component>
<Component>
  <Name>Leisure</Name>
  <Component>
    <Name>Sport Ground</Name>
  </Component>
</Component>
<Component>
  <Name>Background</Name>
</Component>
<Component>
  <Name>Vegetation</Name>
  <Component>
    <Name>Forest</Name>
  </Component>
</Component>
</legendRoot>
</Legend>
```

Annex 2: Semantic relations xml format

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SemanticRelationDescriptor>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Marshalling Area </Name>
    </Layer>
    <Layer>
      <Name> Railroad </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Marshalling Area </Name>
    </Layer>
    <Layer>
      <Name> Station </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Station </Name>
    </Layer>
    <Layer>
      <Name> Railroad </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Water Point </Name>
    </Layer>
    <Layer>
      <Name> River </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Water Point </Name>
    </Layer>
    <Layer>
      <Name> Sea </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> River </Name>
    </Layer>
    <Layer>
      <Name> Sea </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>3</Type>
    <Layer>
      <Name> Highway </Name>
    </Layer>
  </Relation>

```



```

    <Layer>
      <Name> Main Road </Name>
    </Layer>
  </Layer>
  <Layer>
    <Name> Other Road </Name>
  </Layer>
  <Layer>
    <Name> Footpath </Name>
  </Layer>
</Relation>
<Relation>
  <Type>1</Type>
  <Layer>
    <Name>Forest</Name>
  </Layer>
  <Layer>
    <Name> Sport Ground </Name>
  </Layer>
</Relation>
<Relation>
  <Type>3</Type>
  <Layer>
    <Name> High Landside </Name>
  </Layer>
  <Layer>
    <Name> Medium Landside </Name>
  </Layer>
  <Layer>
    <Name> Low Landside </Name>
  </Layer>
</Relation>
<Relation>
  <Type>3</Type>
  <Layer>
    <Name> High Flooding </Name>
  </Layer>
  <Layer>
    <Name> Medium Flooding </Name>
  </Layer>
  <Layer>
    <Name> Low Flooding </Name>
  </Layer>
</Relation>
<Relation>
  <Type>1</Type>
  <Layer>
    <Name> Water Point </Name>
  </Layer>
  <Layer>
    <Name> Low Flooding </Name>
  </Layer>
</Relation>
<Relation>
  <Type>1</Type>
  <Layer>
    <Name> Water Point </Name>
  </Layer>
  <Layer>
    <Name> Medium Flooding </Name>
  </Layer>
</Relation>
<Relation>
  <Type>1</Type>
  <Layer>
    <Name> Water Point </Name>
  </Layer>

```

```

    </Layer>
    <Layer>
      <Name> High Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> River </Name>
    </Layer>
    <Layer>
      <Name> Low Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> River </Name>
    </Layer>
    <Layer>
      <Name> Medium Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> River </Name>
    </Layer>
    <Layer>
      <Name> High Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Sea </Name>
    </Layer>
    <Layer>
      <Name> Low Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Sea </Name>
    </Layer>
    <Layer>
      <Name> Medium Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>1</Type>
    <Layer>
      <Name> Sea </Name>
    </Layer>
    <Layer>
      <Name> High Flooding </Name>
    </Layer>
  </Relation>
  <Relation>
    <Type>2</Type>
    <Layer>
      <Name>Building</Name>
    </Layer>
  </Layer>

```

```
<Name>Background</Name>
</Layer>
</Relation>
</SemanticRelationDescriptor>
```

Bibliography

- [GHJV95] Gamma E., Helm R., Johnson R., Vlissides J., 1995, *Design Patterns*, Addison-Wesley Professional Computing Series (pp 163-173).
- [B67] Bertin J. 1967, *Sémiologie graphique. Les diagrammes, les réseaux, les cartes*, Paris.

Figures Table

Figure 1: Overview of the model.....	5
Figure 2. Main classes of the model	7
Figure 3: Package <code>legendContent</code>	8
Figure 4: Package <code>mapLegend</code>	9
Figure 5: Package <code>symbol</code> and sub-package <code>color</code>	11
Figure 6: Package <code>metadata</code>	12
Figure 7 : COGIT chromatic wheels.....	13
Figure 8 : Package <code>color</code>	14
Figure 9: Typical result: map before (left) and after (right) contrast improvement.....	15
Figure 10 : Semiology menu in the GeOxygene main frame	18
Figure 11 : Items of the semiology menu	19
Figure 12 : Semiology toolbar in the GeOxygene main frame	19
Figure 13 : Contrast improvement toolbar in the GeOxygene main frame	19
Figure 14 : Data loaded in the GeOxygene main frame	20
Figure 15 : Chromatic wheels, the COGIT color reference system	21
Figure 16 : Exploring color contrasts on the COGIT chromatic wheels.....	21
Figure 17 : Specification toolbar loaded in the main frame.	22
Figure 18 : Semantic relation tab in the specification toolbar	24