5 Data content and structure

5.1 Application schemas – Overview

Application schemas included in the IRs

Articles 3, 4 and 5 of the Implementing Rules lay down the requirements for the content and structure of the data sets related to the INSPIRE Annex themes.

IR Requirement
Article 4
Types for the Exchange and Classification of Spatial Objects

1. For the exchange and classification of spatial objects from data sets meeting the conditions laid down in Article 4 of Directive 2007/2/EC, Member States shall use the spatial object types and associated data types, enumerations and code lists that are defined in Annexes II, III and IV for the themes the data sets relate to.

2. Spatial object types and data types shall comply with the definitions and constraints and include the attributes and association roles set out in the Annexes.

3. The enumerations and code lists used in attributes or association roles of spatial object types or data types shall comply with the definitions and include the values set out in Annex II. The enumeration and code list values are uniquely identified by language-neutral mnemonic codes for computers. The values may also include a language-specific name to be used for human interaction.

The types to be used for the exchange and classification of spatial objects from data sets related to the spatial data theme Bio-geographical Regions are defined in the following application schema

- *Bio-geographicalRegions*

The application schemas specify requirements on the properties of each spatial object including its multiplicity, domain of valid values, constraints, etc.

NOTE The application schemas presented in this section contain some additional information that is not included in the Implementing Rules, in particular multiplicities of attributes and association roles.

TG Requirement 1	Spatial object types and data types shall comply with the multiplicities defined for the attributes and association roles in this section.

An application schema may include references (e.g. in attributes or inheritance relationships) to common types or types defined in other spatial data themes. These types can be found in a sub-section called “Imported Types” at the end of each application schema section. The common types referred to from application schemas included in the IRs are addressed in Article 3.

IR Requirement
Article 3
Common Types

Types that are common to several of the themes listed in Annexes I, II and III to Directive 2007/2/EC shall conform to the definitions and constraints and include the attributes and association roles set out in Annex I.

NOTE Since the IRs contain the types for all INSPIRE spatial data themes in one document, Article 3 does not explicitly refer to types defined in other spatial data themes, but only to types defined in external data models.

Common types are described in detail in the Generic Conceptual Model [DS-D2.7], in the relevant international standards (e.g. of the ISO 19100 series) or in the documents on the common INSPIRE models [DS-D2.10.x]. For detailed descriptions of types defined in other spatial data themes, see the corresponding Data Specification TG document [DS-D2.8.x].

Recommendation 1	Additional and/or use case-specific information related to the theme *Bio-geographical Regions* should be made available using the spatial object types and data types specified in the following application schema: Bio-geographicalRegions These spatial object types and data types should comply with the definitions and constraints and include the attributes and association roles defined in this section. The enumerations and code lists used in attributes or association roles of spatial object types or data types should comply with the definitions and include the values defined in this section.

5.2 Basic notions

This section explains some of the basic notions used in the INSPIRE application schemas. These explanations are based on the GCM [DS-D2.5].

5.2.1 Notation

5.2.1.1. Unified Modeling Language (UML)

The application schemas included in this section are specified in UML, version 2.1. The spatial object types, their properties and associated types are shown in UML class diagrams.

NOTE For an overview of the UML notation, see Annex D in [ISO 19103].

The use of a common conceptual schema language (i.e. UML) allows for an automated processing of application schemas and the encoding, querying and updating of data based on the application schema – across different themes and different levels of detail.

The following important rules related to class inheritance and abstract classes are included in the IRs.

IR Requirement
Article 5
Types

(…)

2. Types that are a sub-type of another type shall also include all this type's attributes and association roles.

3. Abstract types shall not be instantiated.

The use of UML conforms to ISO 19109 8.3 and ISO/TS 19103 with the exception that UML 2.1 instead of ISO/IEC 19501 is being used. The use of UML also conforms to ISO 19136 E.2.1.1.1-E.2.1.1.4.

NOTE ISO/TS 19103 and ISO 19109 specify a profile of UML to be used in conjunction with the ISO 19100 series. This includes in particular a list of stereotypes and basic types to be used in application schemas. ISO 19136 specifies a more restricted UML profile that allows for a direct encoding in XML Schema for data transfer purposes.

To model constraints on the spatial object types and their properties, in particular to express data/data set consistency rules, OCL (Object Constraint Language) is used as described in ISO/TS 19103, whenever possible. In addition, all constraints are described in the feature catalogue in English, too.

NOTE Since "void" is not a concept supported by OCL, OCL constraints cannot include expressions to test whether a value is a *void* value. Such constraints may only be expressed in natural language.

5.2.1.2. Stereotypes

In the application schemas in this section several stereotypes are used that have been defined as part of a UML profile for use in INSPIRE [DS-D2.5]. These are explained in Table 1 below.

Table 1 – Stereotypes (adapted from [DS-D2.5])

Stereotype	**Model element**	**Description**
applicationSchema	Package	An INSPIRE application schema according to ISO 19109 and the Generic Conceptual Model.
leaf	Package	A package that is not an application schema and contains no packages.
featureType	Class	A spatial object type.
type	Class	A type that is not directly instantiable, but is used as an abstract collection of operation, attribute and relation signatures. This stereotype should usually not be used in INSPIRE application schemas as these are on a different conceptual level than classifiers with this stereotype.
dataType	Class	A structured data type without identity.
union	Class	A structured data type without identity where exactly one of the properties of the type is present in any instance.
enumeration	Class	An enumeration.
codeList	Class	A code list.
import	Dependency	The model elements of the supplier package are imported.
voidable	Attribute, association role	A voidable attribute or association role (see section 5.2.2).
lifeCycleInfo	Attribute, association role	If in an application schema a property is considered to be part of the life-cycle information of a spatial object type, the property shall receive this stereotype.

version	Association role	If in an application schema an association role ends at a spatial object type, this stereotype denotes that the value of the property is meant to be a specific version of the spatial object, not the spatial object in general.

5.2.2 Voidable characteristics

The «voidable» stereotype is used to characterise those properties of a spatial object that may not be present in some spatial data sets, even though they may be present or applicable in the real world. This does *not* mean that it is optional to provide a value for those properties.

For all properties defined for a spatial object, a value has to be provided – either the corresponding value (if available in the data set maintained by the data provider) or the value of *void.* A *void* value shall imply that no corresponding value is contained in the source spatial data set maintained by the data provider or no corresponding value can be derived from existing values at reasonable costs.

Recommendation 2 The reason for a *void* value should be provided where possible using a listed value from the VoidReasonValue code list to indicate the reason for the missing value.

The VoidReasonValue type is a code list, which includes the following pre-defined values:

- *Unpopulated*: The property is not part of the dataset maintained by the data provider. However, the characteristic may exist in the real world. For example when the “elevation of the water body above the sea level” has not been included in a dataset containing lake spatial objects, then the reason for a void value of this property would be ‘Unpopulated’. The property receives this value for all spatial objects in the spatial data set.
- *Unknown*: The correct value for the specific spatial object is not known to, and not computable by the data provider. However, a correct value may exist. For example when the “elevation of the water body above the sea level” *of a certain lake* has not been measured, then the reason for a void value of this property would be ‘Unknown’. This value is applied only to those spatial objects where the property in question is not known.
- *Withheld*: The characteristic may exist, but is confidential and not divulged by the data provider.

NOTE It is possible that additional reasons will be identified in the future, in particular to support reasons / special values in coverage ranges.

The «voidable» stereotype does not give any information on whether or not a characteristic exists in the real world. This is expressed using the multiplicity:

- If a characteristic may or may not exist in the real world, its minimum cardinality shall be defined as 0. For example, if an Address may or may not have a house number, the multiplicity of the corresponding property shall be 0..1.
- If at least one value for a certain characteristic exists in the real world, the minimum cardinality shall be defined as 1. For example, if an Administrative Unit always has at least one name, the multiplicity of the corresponding property shall be 1..*.

In both cases, the «voidable» stereotype can be applied. In cases where the minimum multiplicity is 0, the absence of a value indicates that it is known that no value exists, whereas a value of void indicates that it is not known whether a value exists or not.

EXAMPLE If an address does not have a house number, the corresponding Address object should not have any value for the «voidable» attribute house number. If the house number is simply not known or not populated in the data set, the Address object should receive a value of *void* (with the corresponding void reason) for the house number attribute.

5.2.3 Enumerations

Enumerations are modelled as classes in the application schemas. Their values are modelled as attributes of the enumeration class using the following modelling style:

- No initial value, but only the attribute name part, is used.
- The attribute name conforms to the rules for attributes names, i.e. is a lowerCamelCase name. Exceptions are words that consist of all uppercase letters (acronyms).

IR Requirement
Article 6
Code Lists and Enumerations

(…)

5) Attributes or association roles of spatial object types or data types that have an enumeration type may only take values from the lists specified for the enumeration type."

5.2.4 Code lists

Code lists are modelled as classes in the application schemas. Their values, however, are managed outside of the application schema.

5.2.4.1. Code list types

The IRs distinguish the following types of code lists.

IR Requirement
Article 6
Code Lists and Enumerations

1) Code lists shall be of one of the following types, as specified in the Annexes:
   a) code lists whose allowed values comprise only the values specified in this Regulation;
   b) code lists whose allowed values comprise the values specified in this Regulation and narrower values defined by data providers;
   c) code lists whose allowed values comprise the values specified in this Regulation and additional values at any level defined by data providers;
   d) code lists, whose allowed values comprise any values defined by data providers.

   For the purposes of points (b), (c) and (d), in addition to the allowed values, data providers may use the values specified in the relevant INSPIRE Technical Guidance document available on the INSPIRE web site of the Joint Research Centre.

The type of code list is represented in the UML model through the tagged value *extensibility*, which can take the following values:

- *none*, representing code lists whose allowed values comprise only the values specified in the IRs (type a);
- *narrower*, representing code lists whose allowed values comprise the values specified in the IRs and narrower values defined by data providers (type b);
- *open*, representing code lists whose allowed values comprise the values specified in the IRs and additional values at any level defined by data providers (type c); and
- *any*, representing code lists, for which the IRs do not specify any allowed values, i.e. whose allowed values comprise any values defined by data providers (type d).

Recommendation 3 Additional values defined by data providers should not replace or redefine any value already specified in the IRs.

NOTE This data specification may specify recommended values for some of the code lists of type (b), (c) and (d) (see section 5.2.4.3). These recommended values are specified in a dedicated Annex.

In addition, code lists can be hierarchical, as explained in Article 6(2) of the IRs.

IR Requirement
Article 6
Code Lists and Enumerations

(…)

2) Code lists may be hierarchical. Values of hierarchical code lists may have a more generic parent value. Where the valid values of a hierarchical code list are specified in a table in this Regulation, the parent values are listed in the last column.

The type of code list and whether it is hierarchical or not is also indicated in the feature catalogues.

5.2.4.2. Obligations on data providers

IR Requirement
Article 6
Code Lists and Enumerations

(….)

3) Where, for an attribute whose type is a code list as referred to in points (b), (c) or (d) of paragraph 1, a data provider provides a value that is not specified in this Regulation, that value and its definition shall be made available in a register.

4) Attributes or association roles of spatial object types or data types whose type is a code list may only take values that are allowed according to the specification of the code list.

Article 6(4) obliges data providers to use only values that are allowed according to the specification of the code list. The “allowed values according to the specification of the code list” are the values explicitly defined in the IRs plus (in the case of code lists of type (b), (c) and (d)) additional values defined by data providers.

For attributes whose type is a code list of type (b), (c) or (d) data providers may use additional values that are not defined in the IRs. Article 6(3) requires that such additional values and their definition be made available in a register. This enables users of the data to look up the meaning of the additional values used in a data set, and also facilitates the re-use of additional values by other data providers (potentially across Member States).

NOTE Guidelines for setting up registers for additional values and how to register additional values in these registers is still an open discussion point between Member States and the Commission.

5.2.4.3. Recommended code list values

For code lists of type (b), (c) and (d), this data specification may propose additional values as a recommendation (in a dedicated Annex). These values will be included in the INSPIRE code list register. This will facilitate and encourage the usage of the recommended values by data providers since the obligation to make additional values defined by data providers available in a register (see section 5.2.4.2) is already met.

Recommendation 4 Where these Technical Guidelines recommend values for a code list in addition to those specified in the IRs, these values should be used.

NOTE For some code lists of type (d), no values may be specified in these Technical Guidelines. In these cases, any additional value defined by data providers may be used.

5.2.4.4. Governance

The following two types of code lists are distinguished in INSPIRE:

- *Code lists that are governed by INSPIRE (INSPIRE-governed code lists).* These code lists will be managed centrally in the INSPIRE code list register. Change requests to these code lists (e.g. to add, deprecate or supersede values) are processed and decided upon using the INSPIRE code list register's maintenance workflows.

  INSPIRE-governed code lists will be made available in the INSPIRE code list register at *http://inspire.ec.europa.eu/codelist/<CodeListName>*. They will be available in SKOS/RDF, XML and HTML. The maintenance will follow the procedures defined in ISO 19135. This means that the only allowed changes to a code list are the addition, deprecation or supersession of values, i.e. no value will ever be deleted, but only receive different statuses (valid, deprecated, superseded). Identifiers for values of INSPIRE-governed code lists are constructed using the pattern *http://inspire.ec.europa.eu/codelist/<CodeListName>*/<value>.

- *Code lists that are governed by an organisation outside of INSPIRE (externally governed code lists).* These code lists are managed by an organisation outside of INSPIRE, e.g. the World Meteorological Organization (WMO) or the World Health Organization (WHO). Change requests to these code lists follow the maintenance workflows defined by the maintaining organisations. Note that in some cases, no such workflows may be formally defined.

  Since the updates of externally governed code lists is outside the control of INSPIRE, the IRs and these Technical Guidelines reference a specific version for such code lists.

  The tables describing externally governed code lists in this section contain the following columns:
  - The *Governance* column describes the external organisation that is responsible for maintaining the code list.
  - The *Source* column specifies a citation for the authoritative source for the values of the code list. For code lists, whose values are mandated in the IRs, this citation should include the version of the code list used in INSPIRE. The version can be specified using a version number or the publication date. For code list values recommended in these Technical Guidelines, the citation may refer to the "latest available version".
  - In some cases, for INSPIRE only a subset of an externally governed code list is relevant. The subset is specified using the *Subset* column.
  - The *Availability* column specifies from where (e.g. URL) the values of the externally governed code list are available, and in which formats. Formats can include machine-readable (e.g. SKOS/RDF, XML) or human-readable (e.g. HTML, PDF) ones.

  Code list values are encoded using http URIs and labels. Rules for generating these URIs and labels are specified in a separate table.

Recommendation 5 The http URIs and labels used for encoding code list values should be taken from the INSPIRE code list registry for INSPIRE-governed code lists and generated according to the relevant rules specified for externally governed code lists.

NOTE Where practicable, the INSPIRE code list register could also provide http URIs and labels for externally governed code lists.

5.2.4.5. Vocabulary

For each code list, a tagged value called “vocabulary” is specified to define a URI identifying the values of the code list. For INSPIRE-governed code lists and externally governed code lists that do not have a persistent identifier, the URI is constructed following the pattern *http://inspire.ec.europa.eu/codelist/<UpperCamelCaseName>*.

If the value is missing or empty, this indicates an empty code list. If no sub-classes are defined for this empty code list, this means that any code list may be used that meets the given definition.

An empty code list may also be used as a super-class for a number of specific code lists whose values may be used to specify the attribute value. If the sub-classes specified in the model represent all valid extensions to the empty code list, the subtyping relationship is qualified with the standard UML constraint "{complete,disjoint}".

5.2.5 Identifier management

IR Requirement
Article 9
Identifier Management

1. The data type Identifier defined in Section 2.1 of Annex I shall be used as a type for the external object identifier of a spatial object.
2. The external object identifier for the unique identification of spatial objects shall not be changed during the life-cycle of a spatial object.

NOTE 1 An external object identifier is a unique object identifier which is published by the responsible body, which may be used by external applications to reference the spatial object. [DS-D2.5]

NOTE 2 Article 9(1) is implemented in each application schema by including the attribute *inspireId* of type Identifier.

NOTE 3 Article 9(2) is ensured if the *namespace* and *localId* attributes of the Identifier remains the same for different versions of a spatial object; the *version* attribute can of course change.

5.2.6 Geometry representation

IR Requirement
Article 12
Other Requirements & Rules

1. The value domain of spatial properties defined in this Regulation shall be restricted to the Simple Feature spatial schema as defined in Herring, John R. (ed.), OpenGIS® Implementation Standard for Geographic information – Simple feature access – Part 1: Common architecture, version 1.2.1, Open Geospatial Consortium, 2011, unless specified otherwise for a specific spatial data theme or type.

NOTE 1 The specification restricts the spatial schema to 0-, 1-, 2-, and 2.5-dimensional geometries where all curve interpolations are linear and surface interpolations are performed by triangles.

NOTE 2 The topological relations of two spatial objects based on their specific geometry and topology properties can in principle be investigated by invoking the operations of the types defined in ISO 19107 (or the methods specified in EN ISO 19125-1).

5.2.7 Temporality representation

The application schema(s) use(s) the derived attributes "beginLifespanVersion" and "endLifespanVersion" to record the lifespan of a spatial object.

The attributes "beginLifespanVersion" specifies the date and time at which this version of the spatial object was inserted or changed in the spatial data set. The attribute "endLifespanVersion" specifies the date and time at which this version of the spatial object was superseded or retired in the spatial data set.

NOTE 1 The attributes specify the beginning of the lifespan of the version in the spatial data set itself, which is different from the temporal characteristics of the real-world phenomenon described by the spatial object. This lifespan information, if available, supports mainly two requirements: First, knowledge about the spatial data set content at a specific time; second, knowledge about changes to a data set in a specific time frame. The lifespan information should be as detailed as in the data set (i.e., if the lifespan information in the data set includes seconds, the seconds should be represented in data published in INSPIRE) and include time zone information.

NOTE 2 Changes to the attribute "endLifespanVersion" does not trigger a change in the attribute "beginLifespanVersion".

IR Requirement
Article 10
Life-cycle of Spatial Objects

(…)

3. Where the attributes beginLifespanVersion and endLifespanVersion are used, the value of endLifespanVersion shall not be before the value of beginLifespanVersion.

NOTE The requirement expressed in the IR Requirement above will be included as constraints in the UML data models of all themes.

Recommendation 6	If life-cycle information is not maintained as part of the spatial data set, all spatial objects belonging to this data set should provide a void value with a reason of "unpopulated".

5.3 Application schema Bio-geographicalRegions

5.3.1 Description

5.3.1.1. Narrative description

The application schema “Bio-geographicalRegions” provides the means for a common pan-European representation of bio-geographical regions and other types of environmental stratifications and ecological regions. The feature type “Bio-geographicalRegion” is the key spatial object of this application schema for representing regions or areas of relatively homogenous ecological conditions with common characteristics. This spatial object type will allow for a proper description of the bio-geographical classification that has been applied to identify and classify the bio-geographical region each feature represents. Within this respect it needs to be emphasized that the application schema not only supports the classification of bio-geographical regions as mandated by the European Habitats Directive, but also meets the requirements raised by INSPIRE stakeholders with regard to alternative and more precise sets of different types of ecological regions.

Currently the *Bio-geographicalRegions* application schema includes four distinct European classification schemes, however through the mechanism of codelists the model can be extended to define and include other classifications as well.

Because of the limited number of bio-geographical region datasets on the one hand and the objective of INSPIRE to strive for maximum harmonisation of datasets on the other hand, the structure of this application schema has been kept simple on purpose: one spatial object comprising information on geometry, an identifier and classification properties. It should also be realized that a strong link exists between the Bio-geographicalRegions application schema and the Annex III theme "*Area management/restriction/regulation zones and reporting units*".

5.3.1.2. UML Overview

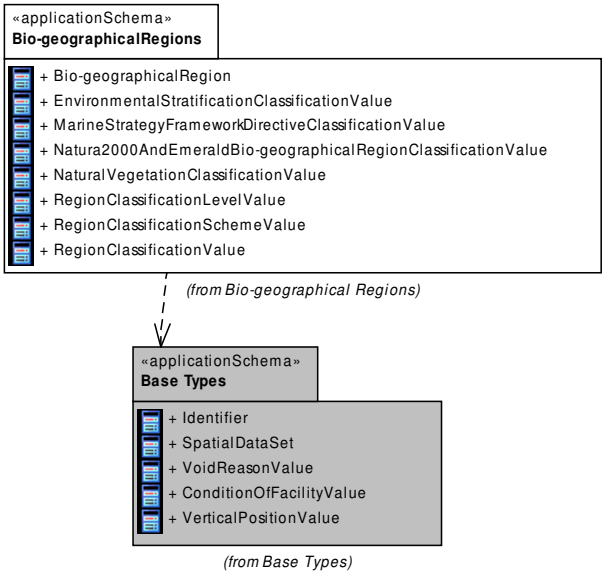

Figure 4 – UML class diagram: Overview of the Bio-geographicalRegions application schema

An overview of the Bio-geographicalRegions package and referenced packages is depicted in figure above. Basically, the Bio-geographicalRegion spatial object type refers to the package Base Types of the General Conceptual Model to include an Identifier.

The complete application schema for Bio-geographical Regions is shown in Figure 5 and described in detail below.

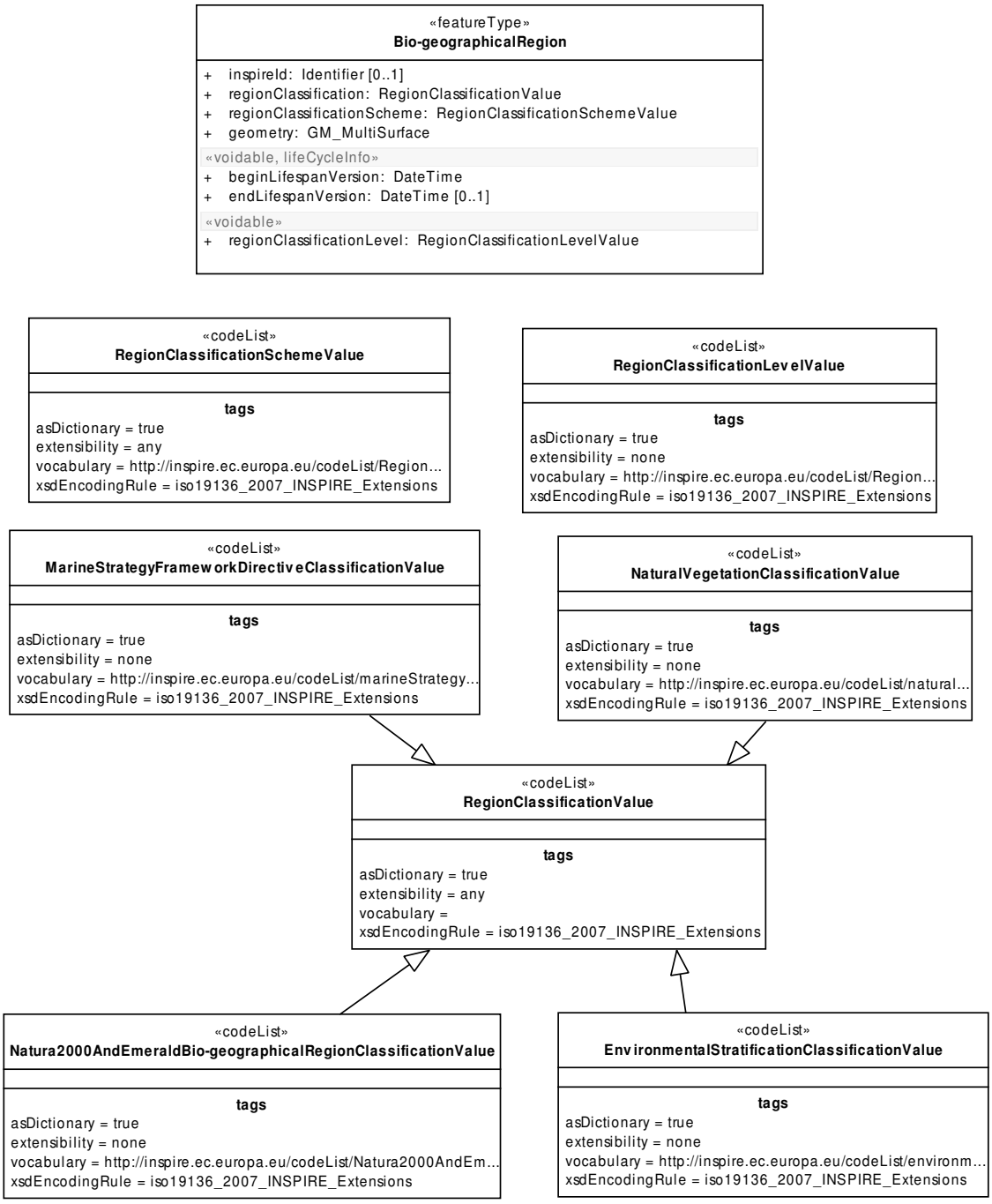

Figure 5 – UML class diagram: Overview of the Bio-geographicalRegions application schema

“Bio-geographicalRegion” is the only single spatial object type included in the application schema and represents any type of bio-geographical region. Since bio-geographical regions can consist of polygons or multi-part polygons, their geometric representation is expressed by GM_Multisurface type. Each single bio-geographical region (i.e. instance of the “Bio-geographicalRegion” spatial object type) is described by a bio-geographical classification. The classification system is specified by three attributes: information on the classification scheme and the classification value that is applicable to the instance is provided by two mandatory attributes respectively called “regionClassificationScheme” and “regionClassification”. Values for these attributes can be selected from a codelist. A third voidable attribute “regionClassificationLevel” has been defined to document the level of the classification system. Potential values for this level are: international, national, regional or local. Many different classification systems exist at different levels; the current application schema only includes information on the classification scheme and corresponding values of 4 European Classification systems, but the schema may be extended to include other classification systems via the codelist mechanism:

- Natura2000 and Emerald Biogeographical regions
  The Natura 2000 and Emerald Biogeographical regions as outlined in respectively the Habitats Directive and the Bern Convention are reporting units that support the process of nature conservation, and more specifically the conservation of species and habitat types under similar natural conditions across a suite of countries, irrespective of political and administrative boundaries. These bio-geographical regions are terrestrial in order to create a similar unit that can be used for assessment an additional 5 marine regions, based on the European marine conventions, have been added. Unlike the terrestrial bio-geographical regions, these marine regions do not have a legal basis. There will be only one corresponding data set for Natura 2000 and Emerald Biogeographical regions, which will be provided by EEA.
- Environmental Stratification of Europe
  The Environmental Zones of Europe are derived from the Environmental Stratification of Europe (see Metzger et al 2005 and Jongman et al 2005). The stratification is based on climate data, data on the ocean influence and geographical position.
- Natural Vegetation of Europe
  The codes for Natural Vegetation in Europe are derived from the map of Natural Vegetation of Europe (Bohn et al 2000).
- Marine Strategy Framework Directive regions
  The codes for the Marine Strategy Framework Directive regions are derived from the Directive itself.

A detailed feature catalogue is included later in this section.

5.3.1.3. Consistency between spatial data sets

It is worth noting that bio-geographical regions are derived features, being based on more detailed work and being modified to make them easier to use at different scales. For instance the Natura2000 and Emerald Biogeographical regions as outlined in the Habitats Directive and the Bern Convention are derived from an interpretation of the digital version of the ‘Map of Natural Vegetation of the member countries of the European Community and of the Council of Europe’ (Noirfalise A., 1987) and the regions have been modified to make them easier to use administratively.

Currently, there are no other consistency rules than those defined within the application schema and no consistency rules between bio-geographical regions and other spatial datasets have been identified.

5.3.3 Feature catalogue

Feature catalogue metadata

Application Schema	INSPIRE Application Schema Bio-geographicalRegions
Version number	3.0

Types defined in the feature catalogue

Type	Package	Stereotypes
Bio-geographicalRegion	Bio-geographicalRegions	«featureType»
EnvironmentalStratificationClassificationValue	Bio-geographicalRegions	«codeList»
MarineStrategyFrameworkDirectiveClassificationValue	Bio-geographicalRegions	«codeList»
Natura2000AndEmeraldBio-geographicalRegionClassificationValue	Bio-geographicalRegions	«codeList»
NaturalVegetationClassificationValue	Bio-geographicalRegions	«codeList»
RegionClassificationLevelValue	Bio-geographicalRegions	«codeList»
RegionClassificationSchemeValue	Bio-geographicalRegions	«codeList»
RegionClassificationValue	Bio-geographicalRegions	«codeList»

5.3.3.1. Spatial object types

5.3.3.1.1. Bio-geographicalRegion

Bio-geographicalRegion

Name: bio-geographical region
Definition: An area in which there are relatively homogeneous ecological conditions with common characteristics.
Description: EXAMPLE Europe is divided into eleven broad bio-geographical terrestrial zones and 5 zones for marine bio-geographical regions.

NOTE The marine regions are used in the context of Natura2000 due to practical / technical reasons only; they do not have any legal status in contrast to the "terrestrial" Bio-geographical Regions.
Stereotypes: «featureType»

Attribute: inspireId

Name: inspire id
Value type: Identifier
Definition: External object identifier of the spatial object.
Description: An external object identifier is a unique object identifier published by the responsible body, which may be used by external applications to reference the spatial object. The identifier is an identifier of the spatial object, not an identifier of the real-world phenomenon.
Multiplicity: 0..1

Attribute: geometry

Bio-geographicalRegion	
Name:	geometry
Value type:	GM_MultiSurface
Definition:	The geometry defining the ecological region.
Multiplicity:	1
Attribute: regionClassification	
Name:	region classification
Value type:	RegionClassificationValue
Definition:	Regionclass code, according to a classification scheme.
Multiplicity:	1
Attribute: regionClassificationScheme	
Name:	region classification scheme
Value type:	RegionClassificationSchemeValue
Definition:	Classification scheme used for classifying regions.
Multiplicity:	1
Attribute: regionClassificationLevel	
Name:	region classification level
Value type:	RegionClassificationLevelValue
Definition:	The classification level of the region class.
Multiplicity:	1
Stereotypes:	«voidable»
Attribute: beginLifespanVersion	
Name:	begin lifespan version
Value type:	DateTime
Definition:	Date and time at which this version of the spatial object was inserted or changed in the spatial data set.
Multiplicity:	1
Stereotypes:	«voidable,lifeCycleInfo»
Attribute: endLifespanVersion	
Name:	end lifespan version
Value type:	DateTime
Definition:	Date and time at which this version of the spatial object was superseded or retired in the spatial data set.
Multiplicity:	0..1
Stereotypes:	«voidable,lifeCycleInfo»

5.3.3.2. Code lists

5.3.3.2.1. EnvironmentalStratificationClassificationValue

EnvironmentalStratificationClassificationValue	
Name:	environmental stratification classification value
Definition:	Codes for climatic stratification of the Environment of Europe.
Description:	Based on environmental variables (climate, geomorphology, oceanicity and northing) using a Principal component analysis and ISODATA clustering routine. The Environmental Stratification of Europe (EnS) consists of 84 Strata, which have been aggregated to 13 Environmental Zones with a spatial resolution of 1 km². NOTE This stratification is after Metzger et al. 2005.
Extensibility:	open

EnvironmentalStratificationClassificationValue	
Identifier:	http://inspire.ec.europa.eu/codelist/environmentalStratification
Values:	The allowed values for this code list comprise the values specified in "Descriptions of the European Environmental Zones and Strata, Wageningen, 2012." and additional values at any level defined by data providers.

5.3.3.2.2. MarineStrategyFrameworkDirectiveClassificationValue

MarineStrategyFrameworkDirectiveClassificationValue	
Name:	marine strategy framework directive classification value
Definition:	Codes for the Marine Stategy Framework Diretive classification.
Extensibility:	open
Identifier:	http://inspire.ec.europa.eu/codelist/marineStrategyFrameworkDirective
Values:	The allowed values for this code list comprise the values specified in "DIRECTIVE 2008/56/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 17 June 2008" and additional values at any level defined by data providers.

5.3.3.2.3. Natura2000AndEmeraldBio-geographicalRegionClassificationValue

Natura2000AndEmeraldBio-geographicalRegionClassificationValue	
Name:	natura 2000 and emerald bio-geographical region classification value
Definition:	Codes for the bio-geographic region classification.
Description:	http://www.eea.europa.eu/data-and-maps/data/biogeographical-regions-europe/codelist-for-bio-geographical-regions
Extensibility:	open
Identifier:	http://inspire.ec.europa.eu/codelist/Natura2000AndEmeraldBio-geographicalRegionClassificationValue
Values:	The allowed values for this code list comprise the values specified in "COMMISSION IMPLEMENTING DECISION of 11 July 2011 concerning a site information format for Natura 2000 sites (notified under document C(2011) 4892) (2011/484/EU)" and additional values at any level defined by data providers.

5.3.3.2.4. NaturalVegetationClassificationValue

NaturalVegetationClassificationValue	
Name:	natural vegetation classification value
Definition:	Codes for the natural vegetation classification.
Extensibility:	open
Identifier:	http://inspire.ec.europa.eu/codelist/naturalVegetation
Values:	The allowed values for this code list comprise the values specified in "Map of the natural vegetation of Europe: scale 1:2,500,000, Part 2: Legend, Bundesamt für Naturschutz (German Federal Agency for Nature conservation), Bonn, 2000." and additional values at any level defined by data providers.

5.3.3.2.5. RegionClassificationLevelValue

RegionClassificationLevelValue	
Name:	region classification level value
Definition:	Codes defining the classification level of the region class.
Extensibility:	none
Identifier:	http://inspire.ec.europa.eu/codelist/RegionClassificationLevelValue
Values:	The allowed values for this code list comprise only the values specified in *Annex C* .

5.3.3.2.6. RegionClassificationSchemeValue

RegionClassificationSchemeValue	
Name:	region classification scheme value
Definition:	Codes defining the various bio-geographical regions.
Extensibility:	any
Identifier:	http://inspire.ec.europa.eu/codelist/RegionClassificationSchemeValue

RegionClassificationSchemeValue	
Values:	The allowed values for this code list comprise any values defined by data providers. *Annex C* includes recommended values that may be used by data providers.

5.3.3.2.7. RegionClassificationValue

RegionClassificationValue	
Name:	region classification value
Definition:	Codes defining the various bio-geographical regions.
Extensibility:	open
Identifier:	http://inspire.ec.europa.eu/codelist/RegionClassificationValue
Values:	The allowed values for this code list comprise the values specified in *Annex C* and additional values at any level defined by data providers.

5.3.3.3. Imported types (informative)

This section lists definitions for feature types, data types and enumerations and code lists that are defined in other application schemas. The section is purely informative and should help the reader understand the feature catalogue presented in the previous sections. For the normative documentation of these types, see the given references.

5.3.3.3.1. DateTime

DateTime	
Package:	Date and Time
Reference:	Geographic information -- Conceptual schema language [ISO/TS 19103:2005]

5.3.3.3.2. GM_MultiSurface

GM_MultiSurface	
Package:	Geometric aggregates
Reference:	Geographic information -- Spatial schema [ISO 19107:2003]

5.3.3.3.3. Identifier

Identifier	
Package:	Base Types
Reference:	INSPIRE Generic Conceptual Model, version 3.4 [DS-D2.5]
Definition:	External unique object identifier published by the responsible body, which may be used by external applications to reference the spatial object.
Description:	NOTE1 External object identifiers are distinct from thematic object identifiers. NOTE 2 The voidable version identifier attribute is not part of the unique identifier of a spatial object and may be used to distinguish two versions of the same spatial object. NOTE 3 The unique identifier will not change during the life-time of a spatial object.

5.3.4 Externally governed code lists

The externally governed code lists included in this application schema are specified in the tables in this section.

5.3.4.1. Governance and authoritative source

Code list	Governance	Authoritative Source (incl. version[1] and relevant subset, where applicable)
Natura2000AndEmeraldBio-geographicalRegionClassificationValue	European Environment Agency (EEA)	Natura 2000 STANDARD DATA FORM, annex of document 2011/484/EU, Section 2.6. *Natura2000AndEmeraldBio-geographicalRegionClassificationValue*
MarineStrategyFrameworkDirectiveClassificationValue	DG Mare/EEA	DIRECTIVE 2008/56/EC, Article 4.1 and Article 4.2 *MarineStrategyFrameworkDirectiveClassificationValue*
NaturalVegetationClassificationValue	Flora Web/Bfn	FloraWeb XLS files. *NaturalVegetationClassificationValue*
EnvironmentalStratificationClassificationValue	Alterra Wageningen UR	Descriptions of the European Environmental Zones and Strata, Alterra Report 2281 *EnvironmentalStratificationClassificationValue*

5.3.4.2. Availability

Code list	Availability	Format
Natura2000AndEmeraldBio-geographicalRegionClassificationValue	http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2011:198:0039:0070:EN:PDF	PDF
MarineStrategyFrameworkDirectiveClassificationValue	http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:164:0019:0040:EN:PDF	PDF
NaturalVegetationClassificationValue	http://www.floraweb.de/download/eurovegmap/natural_vegetation_toplevelunits.xls	XLS
EnvironmentalStratificationClassificationValue	http://content.alterra.wur.nl/Webdocs/PDFFiles/Alterrarapporten/AlterraRapport2281.pdf	PDF

[1] If no version or publication date are specified, the “latest available version” shall be used.

5.3.4.3. Rules for code list values

Code list	Identifiers	Examples
Natura2000AndEmeraldBio-geographicalRegionClassificationValue	Append the names from Section 2.6 of http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2011:198:0039:0070:EN:PDF	As an example 'Alpine'
MarineStrategyFrameworkDirectiveClassificationValue	Append the names from Article 4.1 and for sub regions the names from Article 4.2 at http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:164:0019:0040:EN:PDF	As an example 'the North-east Atlantic Ocean'
NaturalVegetationClassificationValue	Append the names from the Field 'UNITABBR2'.	As an example 'PolarDesertNival'
EnvironmentalStratificationClassificationValue	Append the name in the pdf from the Table of Contents, from Chapters 4 to 15. The sub headings of the chapters give the 2nd level codes.	As an example 'Environmental Zone Nemoral (NEM) '

Code list	Labels	Examples
Natura2000AndEmeraldBio-geographicalRegionClassificationValue	Use the name in “Standard Data Form” Section 2.6 in any official EU language as the label.	As an example 'Alpine'
MarineStrategyFrameworkDirectiveClassificationValue	Use the names in “Marine Strategy Framework Directive” Article 4.1 and for sub regions the names from Article 4.2 in any official EU language as the label.	As an example 'the North-east Atlantic Ocean'
NaturalVegetationClassificationValue	Use the names from the Field 'UNITABBR2'.	As an example 'PolarDesertNival'
EnvironmentalStratificationClassificationValue	Use the names in the “Descriptions of the European Environmental Zones and Strata from” the Table of Contents, from Chapters 4 to 15. The sub headings of the chapters give the 2nd level codes.	As an example 'Environmental Zone Nemoral (NEM) '