

Title:

**Title: A7.1-D2 Concept of application-specific harmonised data models**

Title: A7.1-D2 Concept of application-specific harmonised data models

**Author(s)/Organisation(s):**

Marian de Vries (TUD), Pasquale di Donato (ULSoR), Hugo Ledoux (TUD), Sisi Zlatanova (TUD), Dominique Laurent (IGN)

**Working Group:**

WP7

**References:**

A3.5D1: 0627-a3\_5\_d1-ethz-001-final

A7.0D1: 0707-a7\_0\_d1\_\_concept\_of\_data\_harmonisation\_processes-ethz-002-final

A7.0D2: 0954-a7\_0\_d2\_\_concept\_of\_data\_harmonisation\_processes-ethz-001-final

A7.0D3: 0987-schema\_translation\_\_data\_model\_harmonisation\_-fhg-igd-001-final

A7.5D3: 1045-a7\_5-d3\_profiles\_for\_atmosphere-tud-001-rfc

A7.9D3: 1046-a7\_9-d3\_profiles\_for\_security-tud-001-rfc

A9.1D1: 0687-a9\_1-d1\_overall\_concept\_for\_scenario\_developments-uwe-002-final

HUMBOLDT Core Metadata: 0774-humboldt\_metadata\_report-ulsor-003-final

Glossary of terms: 0819-glossary\_of\_terms-tud-001-new

**Quality Assurance:**

- |  |                             |
|--|-----------------------------|
| <input checked="" type="checkbox"/> Review WP Leader | (WP Leader: )               |
| <input type="checkbox"/> Review dependend WP Leaders | (Depended WP Leaders: )     |
| <input type="checkbox"/> Review Executive Board      | (Executive Board Members: ) |
| <input type="checkbox"/> Review others               | (Other Reviewers: )         |

**Delivery Date:**

**Short Description:**

This is the final version of the document A7.1 Concept of application-specific harmonised data models. Together with deliverables A7.0-D1, D2 and D3 this document provides background information and guidelines for data harmonisation in the HUMBOLDT Scenarios. The focus in this document is on a number of "what" questions: What is an 'application-specific' harmonised data model? What are the requirements for a "god" harmonised model? And what role do these harmonised models play in data harmonisation in the HUMBOLDT Scenarios and the HUMBOLDT project as a whole.

**Keywords:**

Application-specific data model, harmonised data model, INSPIRE, ISO, data product

Title:

specification, Scenario information model

<b>History:</b>			
<i>Version</i>	<i>Author(s)</i>	<i>Status</i>	<i>Comment</i>
000	Marian de Vries	new	
001	Sisi Zlatanova	update	Added examples to chapter 6, added data modelling process steps to 6. Some edits in chapters 1, 2 and 3.
002	Hugo Ledoux	review	Small edits in chapter 3
003	Marian de Vries	update	Revision of chapter 5 and Annex A (because of changes in INSPIRE documents D2.5 and D2.6), Cadastral Parcels example now from CP Theme data specification
004	Marian de Vries	final	Textual edits

Title:

## Table of contents

1 Introduction .....	5
1.1 Purpose and scope.....	5
1.2 Terms and definitions.....	5
1.3 Overview .....	5
2 Data models in HUMBOLDT.....	6
2.1 Data harmonisation.....	6
2.2 Application-specific data models.....	7
2.2.1 Real-world objects, data and data models.....	7
2.2.2 What is an “application-specific” data model (in HUMBOLDT).....	8
2.2.3 Levels of abstraction: from conceptual to physical.....	8
2.3 Components of an application-specific data model.....	8
2.4 What is a “harmonised” application-specific data model.....	10
2.5 Role of data models in HUMBOLDT.....	11
3 Modelling spatial information.....	12
3.1 Feature or field-based spatial model.....	13
3.2 Vector or raster storage model.....	14
3.3 Data and metadata.....	16
3.4 Semantics.....	18
3.5 Spatial data and maps.....	20
4 ISO/OGC: standards and profiles.....	21
4.1 The ISO General Feature Model.....	21
4.2 Coverages.....	23
4.3 The concept of „Profiles“.....	24
5 INSPIRE.....	25
5.1 Generic Conceptual Model.....	25
5.2 Data specifications for the Annex Themes.....	25
5.2.1 Application schema.....	25
5.2.2 Feature catalogue.....	27
5.2.3 Components of an INSPIRE Theme data specification.....	28
5.3 INSPIRE base types.....	28
5.4 Consolidated INSPIRE UML model.....	30
6 Harmonisation issues.....	31
6.1 Data format and/or type of web service.....	31
6.2 Spatial reference system.....	31

---

Title:

6.3 (Conceptual) data model.....	31
6.4 Classification schemes.....	34
6.5 Terms and concepts: thesaurus, ontology.....	35
6.6 Metadata profile.....	36
6.7 Aggregation, multiple representation.....	36
6.8 Portrayal rules.....	37
6.9 Processing functions.....	38
6.10 Multi-linguality.....	38
7 Summary.....	39
8 References.....	40
ANNEX A: INSPIRE D2.5 - Modelling application schemas.....	41

Title:

# 1 Introduction

## 1.1 Purpose and scope

This is the final version of the document A7.1 Concept of application-specific harmonised data models. Together with deliverables A7.0-D1, D2 and D3 this document provides background information and guidelines for data harmonisation in the HUMBOLDT Scenarios.

The focus in this document is on so called “what” questions: What is an ‘application-specific’ harmonised data model? What are the requirements for a “good” harmonised model? And what role do these harmonised models play in data harmonisation in the HUMBOLDT Scenarios and the HUMBOLDT project as a whole.

Data models play an important role in human interaction during a software development process, to facilitate discussion about the content and structure of information used in applications, in our case the Scenarios. At the same time, data models – because they are expressed in a formal language – are machine ‘treatable’, and in that capacity they are essential for (semi-)automated data harmonisation processes in HUMBOLDT.

The deliverables A7.0-D1, D2 and D3 have as a subject the “how” questions: how to specify a data model for the Scenario and how to carry out the transformation of existing datasets from source data model to target data model.

## 1.2 Terms and definitions

The list of terms is in a separate Terms and definitions document. The list can more easily be maintained this way (document name: Glossary of terms).

## 1.3 Overview

Chapter 2 gives answer to the questions: what an application-specific data model is, what it consists of, and what role it plays in the HUMBOLDT Scenarios. Chapter 3 provides a general background on the modelling of spatial information such as some basic distinctions are mentioned (feature-based and field-based data structures, vector and raster data formats, data and metadata, and spatial data and maps). Chapter 4 introduces the main ISO standards that are relevant for HUMBOLDT, i.e. ISO 19109, 19107 and 19123. The list is compliant with the data modelling and data harmonisation requirements in the Scenarios. In chapter 5, the attention is on INSPIRE, especially the contributions of the Data Specification drafting team. The question here is: what can HUMBOLDT learn and apply from the INSPIRE documents of the DS drafting team? Chapter 6 is not primarily about data modelling or the harmonisation of data models, but about the broader list of harmonisation issues that were discussed within the task “Other harmonisation issues”.

Title:

## 2 Data models in HUMBOLDT

The purpose of this Chapter is to define what an application-specific data model is in HUMBOLDT, what it consists of, and why data models differ, even in the same application field.

We also point shortly at the role that data models play in the data harmonisation processes in HUMBOLDT (this second topic is treated in more detail in 7.0-D1).

### 2.1 Data harmonisation

The purpose of data harmonisation is to resolve heterogeneity in spatial data, so that data from different data providers in different countries can be easily and effectively combined.

Data harmonisation has many aspects. The RISE project created a list of data harmonisation components, that was adopted by the INSPIRE drafting team on Data Specifications (see Figure 2.1).

(A) INSPIRE Principles	(B) Terminology	(C) Reference model
(D) Rules for application Schemas and feature catalogues	(E) Spatial and temporal aspects	(F) Multi-lingual text and cultural adaptability
(G) Coordinate referencing and units model	(H) Object referencing modelling	(I) Data translation model/guidelines
(J) Portrayal model	(K) Identifier Management	(L) Registers and registries
(M) Metadata	(N) Maintenance	(O) Quality
(P) Data Transfer	(Q) Consistency between data	(R) Multiple representations
(S) Data capturing	(T) Conformance	

*Figure 2.1. INSPIRE data harmonisation components*

Looking at the long list of harmonisation aspects, a distinction can be made between:

- Harmonisation of the general characteristics of datasets: the data model of the datasets, terms and concepts used, classifications used, scale (in the sense of aggregation level / level-of-detail) of the datasets, portrayal models that specify the visualisation rules, etc.
- Harmonisation of (individual) data instances: solving conflicts in case of spatially overlapping datasets (conflation issue), and in cross-border situations: merging the geometry of (the same) spatial objects at both sides of a border so that the merged object's geometry is correct (edge matching), and detecting and possibly solving gaps in the spatial coverage, and other data quality aspects.

Title:

This document A7.1 deals with the first category of harmonisation aspects: harmonisation of the general characteristics of datasets, more specifically with (in the numbering of Figure 2.1):

- B Terminology
- D Rules for application schemas
- E Spatial and temporal aspects
- F Multilingual text
- G Coordinate referencing – units of measurements
- H Object referencing modelling
- J Portrayal
- K Identifier management
- M Metadata
- P Data transfer
- R Derived reporting/multiple representations

The term ‘application-specific data model’ does not show in this list: as we will see in Chapter 5, in INSPIRE and ISO the comparable term “application schema” is used (see component D).

## 2.2 Application-specific data models

### 2.2.1 Real-world objects, data and data models

If we consider digital information as representation of real-world (or virtual) objects, what then is a data model?

A ‘data model’ formally describes content and structure of datasets that are in accordance with that model, including constraints such as a list of permitted values for a certain attribute (codelists and enumerations). Data models abstract from the real content of a dataset, but have to do with the general characteristics: data structure, permitted attribute values, operations that are allowed on certain attributes, and other constraints on the possible content. A data model plays a role in different stages of application development and implementation (c.f. Section 2.5).

The term ‘data model’ is a general ICT term, used in software engineering and in database design and implementation. That is the reason this term is also used in HUMBOLDT. As we will see in Chapter 5, other terms can be used as well as synonyms: for example in the INSPIRE documents the term ‘application schema’ is used to denote a data model for a certain application or application domain. Another term often used in this context is: conceptual schema.

A short definition of a data model could be as follows: “A data model is a formal description of structure and permitted content of data sets”. The term ‘formal’ has to do with the fact that a data modelling language is used (and not natural language), with a set of well-defined constructs.

Title:

## 2.2.2 What is an “application-specific” data model (in HUMBOLDT)

In HUMBOLDT a complete Scenario can be seen as an application. At the same time, in a more technical sense, the software that supports the use cases in that Scenario can also be called applications.

For the definition of ‘application-specific data model’ we take the first meaning, and consider a Scenario as one application. In HUMBOLDT therefore ‘application-specific’ amounts to ‘Scenario-specific’. Hence, the ‘Scenario data model’ specifies (= describes and prescribes) the structure and permitted content of the (spatial and non-spatial) data that is used in that Scenario.

A related term that is used in this document is ‘application domain’ (or ‘application field’). We define application domain as follows: “An application domain is a thematic area of interest, often with an information community that has a common (explicit or implicit) conceptual view on that theme of interest”.

An ‘information community’ is defined by OGC as: “A collection of people (a government agency or group of agencies, a profession, a group of researchers in the same discipline, corporate partners cooperating on a project, etc.) who, at least part of the time, share a common digital geographic information language and common spatial feature definitions.” (OGC 2008)

## 2.2.3 Levels of abstraction: from conceptual to physical

A data model describes the information used in an application, and it can do so at different levels of abstraction. A common distinction is that between a conceptual, logical and physical data model:

- At a conceptual level = from the viewpoint of the business process as expressed in the use cases, and as much as possible in terms of the end-user;
- At a logical level = the data structures, codelists, classifications etc. as they will be implemented in a data store or in an exchange file;
- The physical data model = detailed instructions on how to physically store the data: including indexes, table spaces, (de)compression techniques, and in the XML world: XLink relations, namespaces, etc.

In practical situations these three types of data model are often reduced to two levels: conceptual/logical and physical. In ISO and INSPIRE documents for example there is no distinction made between conceptual and logical levels; what is called “conceptual schema” in the ISO world, is in fact often a logical model, because the schema already takes implementation requirements into account.

## 2.3 Components of an application-specific data model

A data model that describes the spatial information used in an application will always consist of these two categories of model elements:

1. Generic elements
  - a. the simple datatypes for numeric, string, date and time
  - b. the datatypes specific to the spatial component: when the location attribute holds point coordinates the datatype will be "GM\_Point", etc ...

Title:

- c. elements that make the data model conform to the ISO General Feature Model, or the Coverage base model (ISO19123) (depending on whether discrete spatial objects are modelled, or continuous phenomena) (see Chapter 3)
- d. other characteristics that are the result of the implementation of ISO, OGC or INSPIRE rules and recommendations

## 2. Application-specific elements

- a. model elements (classes especially, but also codelists or enumerations) that are "borrowed" from another application (or application domain) data model that is already in use elsewhere (this can be a de-facto or de-jure standard, but does not have to be).

In the HUMBOLDT case we could e.g. use parts of the LADM (Land Administration Domain Model, for cadastral parcels and ownership), WFD (Water Framework Directive), OGC's Observations & Measurements model, or EuroRoadS. When the INSPIRE Theme Working Groups have specified the conceptual schemas for there Theme, also these can be used (completely or partly).

- b. new elements, defined in the Scenario data model itself. Here there also two possibilities:
  - all the application-specific elements of the model are new, there is no borrowing from other application models;
  - there is a mix of new elements and borrowed elements: the Scenario data model builds on existing application models and adds new elements.

There are several possibilities of mixes of a. and b.: only borrow, only new, or mixed: and when borrowed this can amount to re-use of complete data models or of subset.

Title:

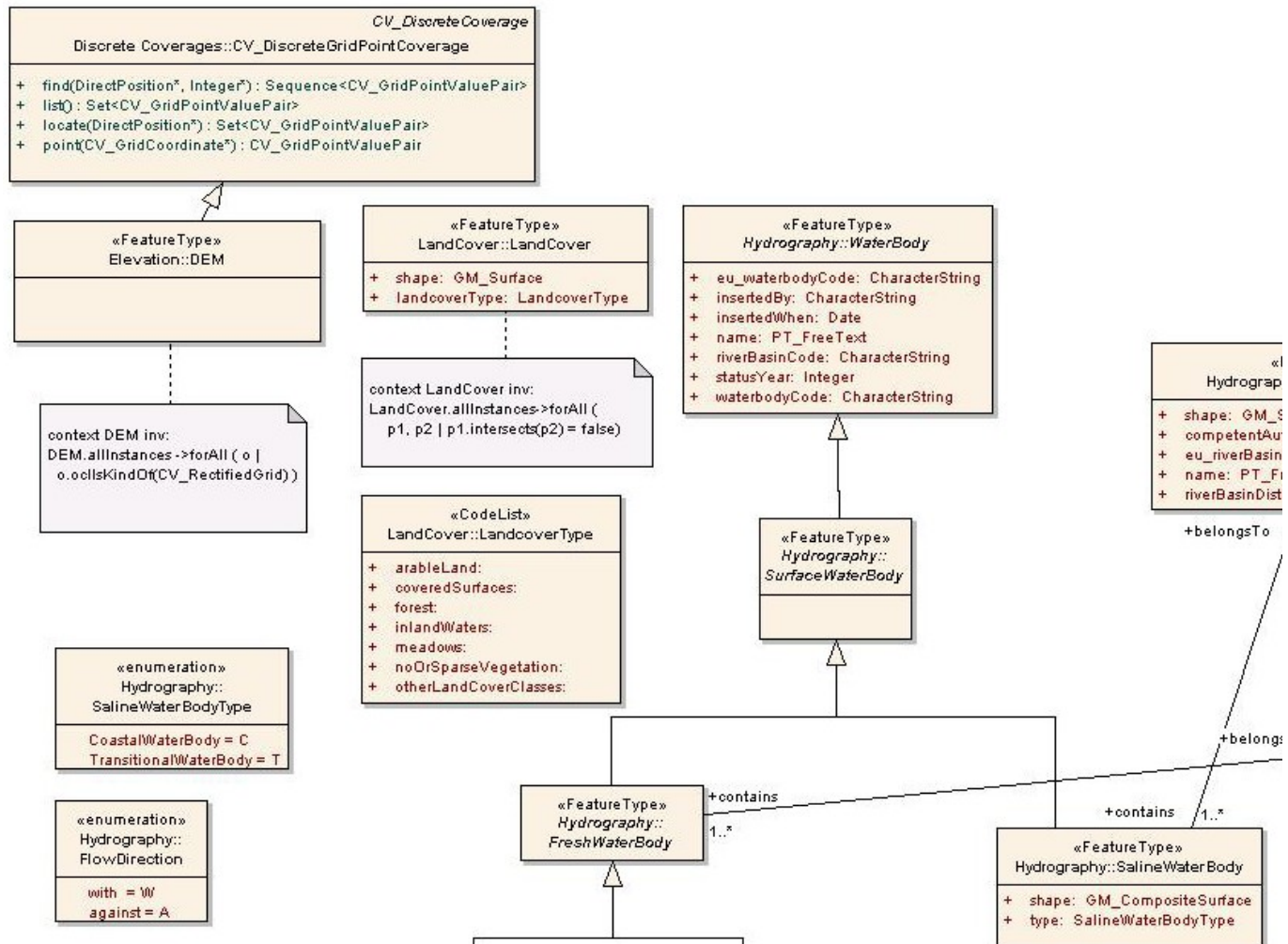


Figure 2.2. Combining different application domain models (RISE project)

Whatever the case, important is that a Scenario-specific data model will always consist of both generic and application-specific parts.

## 2.4 What is a “harmonised” application-specific data model

To answer this question first a distinction must be made between two scopes of data harmonisation in HUMBOLDT: harmonisation within the Scenario and harmonisation between the Scenarios, and when required, with the ‘outside world’.

A “harmonised” Scenario data model is then:

1. the overall data model for the Scenario, that describes all information used and produced in that Scenario, and to which all datasets in that Scenario conform;

This involves the specification of a common Scenario data model, and the transformation of datasets to that new target model, as basis for easy and effective data integration.

Title:

2. a data model that fulfils a number of criteria so that data exchange with other Scenarios and/or the outside world (outside HUMBOLDT) is possible.

This involves (also, as in the first case) adherence to international, vendor-neutral standards, to ensure syntactic interoperability when data is exchanged. In addition, extra attention for semantic interoperability is needed, because conceptual viewpoints, terminology, and ways of interpreting geodata can differ largely between application fields and consequently between Scenarios.

## 2.5 Role of data models in HUMBOLDT

What is the purpose of formally specifying the information used in the Scenarios into data models?

Having formal data models serves at least three goals:

1. Specifying the data model (in UML) helps the information analysis in the Scenarios: it is a good way to decide what information is exactly needed, what classifications to use, which attributes can have null values, and which are mandatory, etc.
2. Having 'machine treatable' data models is essential for the transformation of existing datasets to that new target model (this can be on-the-fly or beforehand, offline) (model-driven data transformation) (see 7.0-D1). Part of this is also: validation of the transformed datasets: do they conform to the specified model?
3. The data models can be published as part of the metadata descriptions of the datasets (or data services) in the Scenario itself, or in a cross-Scenario portal (see 9.1D1)

Title:

### 3 Modelling spatial information

GI users/professionals interact with operational GIS in order to query, analyse, and derive knowledge from spatial data. Anyway, the results of data manipulation will provide meaningful outcomes only if data reflect the nature of the “real situation” they are intended to represent. Choosing an inappropriate representation will compromise the picture we have of the real world and our reasoning about it.

Data modelling is the process of abstracting and simplifying the complexity of the “real world” for the purpose of representing and analysing a domain of interest within the finite “reality” of a computer.

Modelling geographic information involves three main steps: (i) identifying the spatial entities of interest in the context of a specific application, and choosing how to represent them in a conceptual model; (ii) mapping the conceptual model to a spatial data model (logical model); (iii) choose a specific data structure to store data in the computer (physical model) (Heywood *et al.*, 1998) (Burrough & McDonnell, 1998). The following picture provides a snapshot of this process:

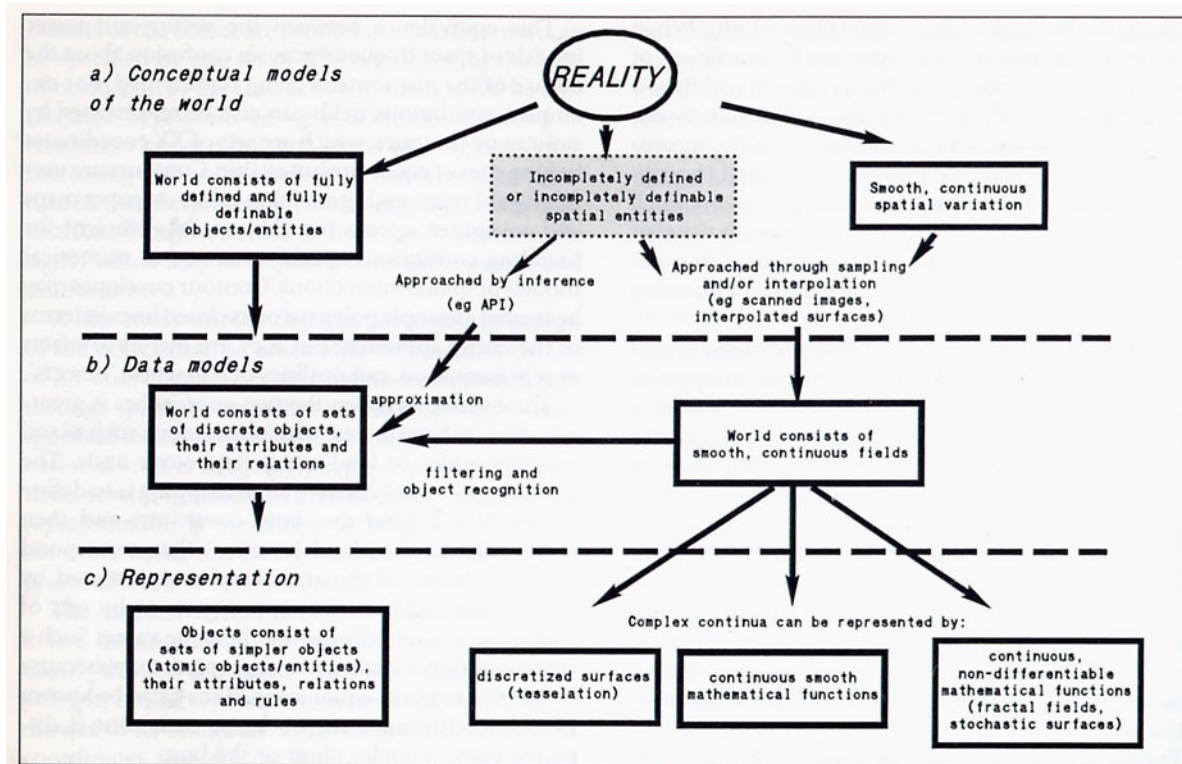


Figure 3.1: the spatial data modelling process (© 1998 Burrough & McDonnell)

What is not explicitly shown in this figure (top level) is the start of any GIS or EA project: the phase called “identifying the spatial entities of interest in the context of a specific application”. This information analysis phase, leading to a conceptual data model of the relevant data in the project, then leads to an important other decision: is my spatial information a set of discrete spatial objects, or does it deal with continuous phenomena (in 2D or in 3D space), such as land cover, temperature, wind velocity etc.

Title:

This has a strong relation with the existence of different approaches to the modelling and storage of spatial data: the object-based (or feature-based) and the field-based approach for the logical and physical data model, see the next section.

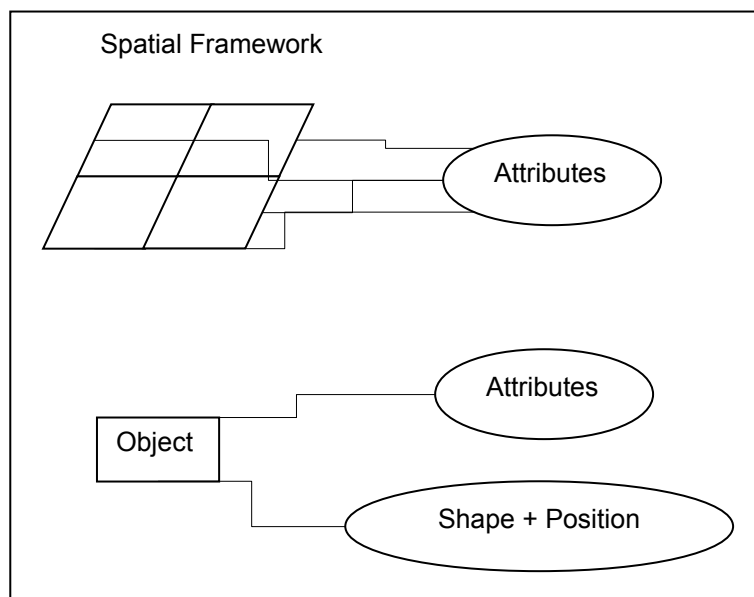
### 3.1 Feature or field-based spatial model

In the field of Geographic Information two main approaches exist to formalise space and spatial properties: the feature-based (also called object-based) and the field-based approach.

In the feature-based approach space is perceived as occupied by discrete, identifiable objects<sup>1</sup> with a position (defined against a specific coordinate system or spatial reference system), a shape and non-geometric characteristics (attributes).

In the field-based approach earth's surface is perceived as a spatial (or spatio-temporal) continuum, where attributes of interest vary continuously over space: the spatial distribution on attributes variation is often formalised as a mathematical function from a spatial framework, a partition of space into a finite tessellation of spatial entities, to the attributes domain (Worboys, 1995).

Opting for one of the two approaches is a crucial task, since this choice determines the logical and physical data models (the concrete data structures, and with that eventually the way this spatial information is stored). If one considers, for example, a mountain area: should this be conceptualised with the field-approach or with the object-based one? Opting for an object-based approach will enable to count and name the peaks of mountains, but will not enable to derive slopes (Burrough & McDonnell, 1998).



<sup>1</sup> Objects, entities, features are often used as synonymous. Feature ("*abstraction of real world phenomena*" [ISO 19101]) is the term widely used in standards (e.g. ISO TC 211) and specifications (e.g. OGC)

Title:

*Figure 3.2: field-based (top) and object-based (bottom) perception of space*

## 3.2 Vector or raster storage model

At logical level the conceptual model is mapped to one of the two main spatial storage models used within GIS: the vector model or the raster model.

The vector model uses the geometric primitives point, line, and polygon, geographically referenced by a Cartesian coordinate system, to model real world entities. The raster model is a result of developments in the field of computer graphics. The raster model uses an array of cells (usually a grid of square cells), called pixels, to represent real world entities.

It is often said that the vector model is best suited to encode the feature-based conceptual model, while the raster data model is best suited to encode the field-based conceptual model. This does not mean that a field-based conceptual model cannot be encoded with a vector-based data model, and vice versa. In fact the spatial framework used in field-based conceptual model can be encoded with a vector data model: triangulated irregular networks (TIN) or the Thiessen/Dirichlet/Voronoi diagrams are just examples of vector space partitions used to represent continuous phenomena.

Logical data models are implementation-oriented, accordingly to the specific GIS and/or DBMS (Data Base Management System) used to manage and store data. The ESRI software ArcGIS as well as the Intergraph software Geomedia are basically vector-based, while GRASS and IDRISI, for example, are raster-based. Again it does not mean that one cannot implement a feature-based model within a raster-based GIS.

The physical level deals with how data are digitally stored into a computer; spatial data models need to be further specified in order to provide to the computer rules to reconstruct these models in digital format. Both vector and raster spatial models may be digitally encoded with a variety of spatial data structures (this diversity is the main cause of difficulties in exchanging data between different GIS): spatial data structures are the physical way in which data are encoded in digital format into a computer for the purpose of storage and analysis.

In the field of Geographic Information the so-called geo-relational model has become popular over the years: this is a hybrid approach used by most software companies to manage the complexity of spatial data (early relational DBMS were unable to manage geometries and topological rules). With this approach the geometry (and the topological rules) are stored in ordinary computer files (often binary encoded in proprietary formats), while the associated attributes are stored in relational DBMS: relationships are reconstructed at runtime by the software using keys.

During the last years relational DBMS have been extended with features from object oriented DBMS: ORDBMS (Object-Relational Data Base management Systems) are now capable of managing both geometric/topological and attributes data<sup>2</sup>.

Vector data structures may be classified in two macro-categories: non-topological data structures and topological data structures. Non-topological data structures (also known as “spaghetti” data) use a simple list of XY coordinates to store geometries, but are not able to manage topology (relationships

---

<sup>2</sup> The ISO standard 19125-2 specifies an SQL schema for storage, retrieval, query and update geospatial feature. This standard is implemented by software such as Oracle Spatial, Geomedia Professional, ESRI ArcSDE, ESRI ArcGIS.

Title:

between feature such as “Point A is connected with point B by line L” or “Polygon A and B are adjacent since they share part of their boundary”).

Topological data structures may be further classified according to the way they manage topology: explicitly or implicitly. Explicitly coded topology means that topology is directly stored within the file structure (e.g. the ESRI coverage<sup>3</sup> data structure), while for implicitly encoded topology (e.g. ESRI Shapefile) the relationships are managed at runtime by software<sup>4</sup>.

In raster data structures spatial data is stored as an array of grid values, with metadata (e.g. coordinates of the upper-left corner of the grid, pixel size, number of rows and columns, number of features represented, etc.) in the file header; this array may be stored as a file (usually compressed) or as a database record (Longley *et al*, 2001).

```
ncols 157
nrows 171
xllcorner -156.08749650000
yllcorner 18.870890200000
cellsize 0.00833300
0 0 1 1 1 2 3 3 5 6 8 9 12 14 18 21 25 30 35 41 47 53
59 66 73 79 86 92 97 102 106 109 112 113 113 113 111 109 106
103 98 94 89 83 78 72 67 61 56 51 46 41 37 32 29 25 22 19
.....
```

Figure 3.3: example of file header of raster data file (ArcINFO ASCII Grid Format)

Raster data structures may be classified as simple or complex. In simple raster data structures a number of overlay layers is used, each layer containing only one feature type (cfr. figure 3.4)

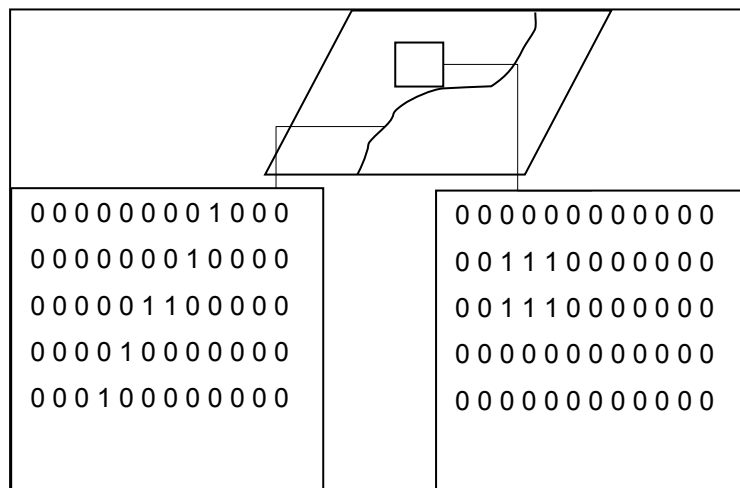


Figure 3.4: simple raster data structure

<sup>3</sup> This should not be confused with the term coverage as intended in ISO TC 211 standards. Here we are dealing with file formats, in ISO standards the term coverage is related to the field-based view of the world (examples of coverages include rasters, triangulated irregular networks, point coverages and polygon coverages).

~~<sup>4</sup> Note that in ORDBMS topological rules are defined and managed within the DMBS itself.~~

Title:

In complex raster data structures it is assumed that entities do not occupy the same space, so that it is possible to store several features in the same grid with an entity code assigned to each pixel.

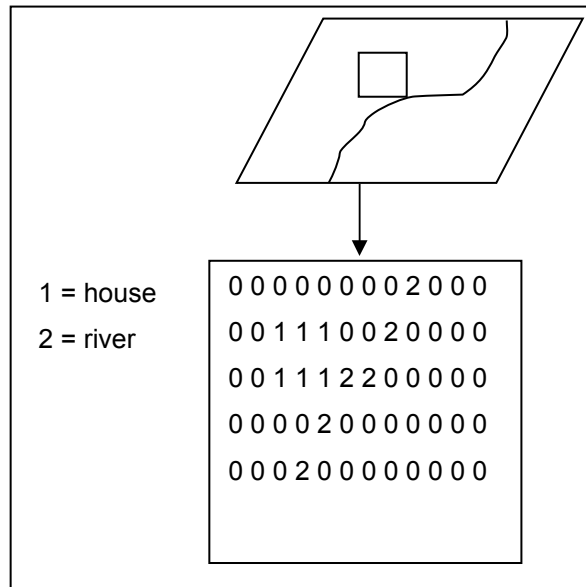


Figure 3.5: complex raster data structure

Since raster data structures are “storage-hungry”, a number of compression techniques (either lossy or lossless) is used to encode raster data. JPEG2000 is a fairly new standard that enable to compress raster data both in lossy or in lossless mode.

### 3.3 Data and metadata

In the area of geographic information metadata provides information about the What, Who, Where, Why, When and How of the data: compared to metadata collected in other sectors there is a major emphasis on the spatial component, the where element.

There are significant benefits to such asset management both from the data providers and the final user point of views. Comprehensive and quality metadata enable data providers to maintain and organise their data assets, avoid duplication of data collection, advertise and promote their data availability. From a final user perspective metadata provide facilities to discover data and assess their “fitness of use”.

Metadata may be applied at different levels with different objectives and are usually classified as follows:

- Discovery metadata mainly addresses the issue of data advertising and discovery;
- Exploration metadata enables a user to investigate the “fitness of use” of data;
- Exploitation metadata provides information on how to access and use data, also in terms of policy/licensing.

Depending on requirements, metadata may be applied at different levels of data granularity, from datasets to single feature, as well as to services. What constitutes a dataset is a matter of interpretation: the European standard EN ISO 19115 Geographic information – Metadata (ISO 19115:2003), as well as the INSPIRE Draft Implementing Rules for Metadata (Version 3), provides

Title:

quite an ambiguous definition of dataset as an “*identifiable collection of data*”, and states that “*the definition of what constitutes a “dataset” is more problematic and reflects the institutional and software environments of the originating organization*”.

Metadata may also be implemented hierarchically, when different datasets share similar characteristics in terms of theme, source, and creation process, as is the case, for example, of a collection of raster maps captured from a common series of paper maps.

In order to ensure interoperability and harmonisation, metadata should conform to a standard: in the area of Geographic Information the reference standards for metadata are ISO19115:2003<sup>5</sup> (EN ISO 19115), ISO 19139:2007.

ISO 19115:2003 is a comprehensive standard with almost 300 metadata elements, organised in packages (see figure 2), required for describing geographic information and services in terms of identification, extent, quality, spatial and temporal schema, spatial reference, distribution.

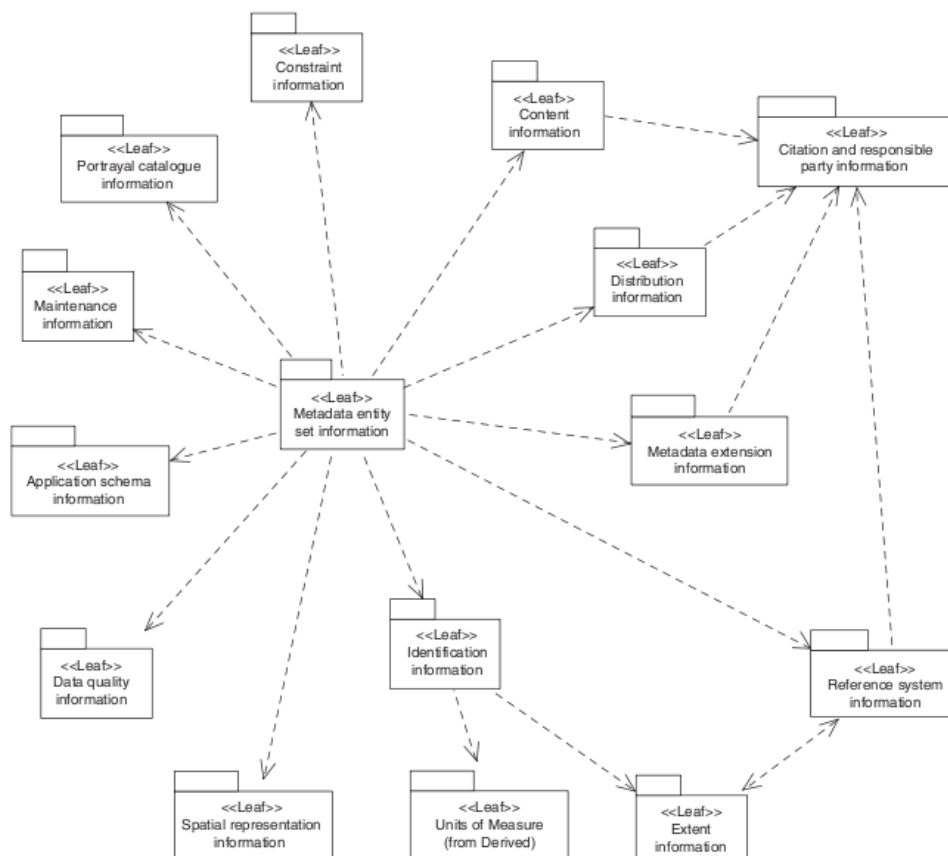


Figure 3.5: ISO 19115 metadata packages (© ISO 2003)

Though ISO 19115 is mainly intended for digital data it can be applied to other forms of geographic data such as map, charts, textual documents, as well as to non-geographic data.

The standard defines mandatory, conditional (mandatory under certain conditions), and optional metadata sections, entities, and elements; it also defines a minimum set (so called Core) of metadata

<sup>5</sup> A Technical Corrigendum to ISO 19115 has been published by ISO in 2006

Title:

required to provide a range of information for discovery, fitness of use, data access and transfer. The value of some metadata elements are constrained to predefined list of values (codelists and enumerations).

The standard is applicable to datasets, dataset series, single geographic feature or single feature property (attribute) and also enables for hierarchy in metadata.

Rules for extending the elements set through user-defined elements are also available. If extensions are quite extensive, the standard recommends creating a so-called “Community Profile”: a profile usually contains the core elements (should be part of the profile)) plus some more elements from the standard plus user-defined elements (see figure 3.6).

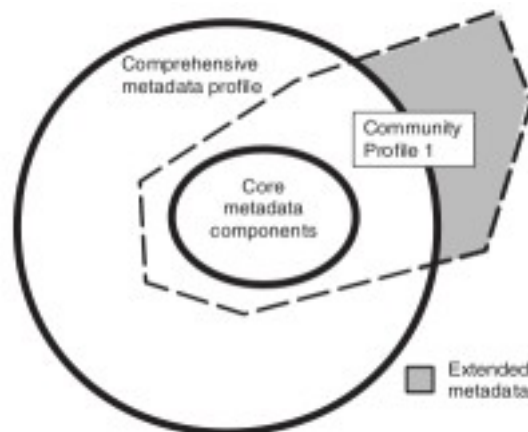


Figure 3.6: community profile (© ISO 2003)

While ISO 19115 is an abstract specification, since it does not provide rules for metadata encoding, ISO 19139:2007 provides a spatial metadata encoding in XML, an XML schema implementation derived from ISO 19115<sup>6</sup>. ISO 19139 is designed to provide a common XML specification for describing, validating and exchanging geographic metadata towards interoperability.

There are two more documents that are useful for metadata in the field of Geographic Information. ISO 19119:2005, which identifies and defines the architecture patterns for service interfaces used for geographic information, provides specifications on how to manage metadata for geo-services. ISO/DIS 19115-2, which is actually a draft standard, provides extensions for imagery and gridded data metadata.

### 3.4 Semantics

‘Semantics’ deals with aspects of meaning as expressed in a language, either natural or technical such as a computer language, and is complementary to syntax which deals with the structure of signs (focusing on the form).

<sup>6</sup> Unofficial ISO 19139 XML Schemas are available from [http://eden.ign.fr/xsd/isote211/index\\_html?set\\_language=en&cl=en](http://eden.ign.fr/xsd/isote211/index_html?set_language=en&cl=en)

Title:

In the area of distributed data sources and services, semantic interoperability refers to the ability of systems to exchange data and functionalities in a meaningful way. Semantic heterogeneity occurs when there is no agreement on the meaning of the same data and/or service functionality.

Data creation happens in a context or in an application domain where concepts and semantics are clear to the data creator, either because they are explicitly formalised or because they are naturally applied due to a yearly experience. But with distributed data resources this context is missed and unknown to the end user. This means that, in order to achieve semantic interoperability, semantics should be formally and explicitly represented (Kuhn, 2005).

Dealing with data integration basically implies addressing two main types of heterogeneity: data heterogeneity and semantic heterogeneity. Data heterogeneity refers to differences in data in terms of data type and data formats, while semantic heterogeneity applies to the meaning of the data (Hakimpour and Geppert, 2001). Semantic heterogeneities may be classified in two macro-categories: naming heterogeneities (when different words/expressions are used for the same concept) and conceptual heterogeneities (when different concepts are expressed by the same words/expressions/symbols) (Kuhn, 2005).

Application schemas and metadata may be considered as a means to provide information about the context in which data have been created, but schemas do not provide explicit semantics of their related data, and metadata values are not machine-readable (Klien, 2007).

There are several ways (controlled vocabularies, taxonomies, thesaurus, ontologies) to explicitate the semantics of a dataset or of an application domain; the approaches vary in terms of complexity, formalism, and amount of information they provide.

- **Controlled vocabulary:** a controlled vocabulary is a list of terms that have been enumerated explicitly (controlled means that there is registration authority responsible for it). Controlled vocabularies solve the problems of homonymy (a group of words that share the same spelling or pronunciation (or both) but have different meanings), synonymy (different words with identical or at least similar meanings) and polysemy (is a word or phrase with multiple, related meanings) by ensuring that each concept is described using only one authorized term and each authorised term in the controlled vocabulary describes only one concept<sup>7</sup>: when different terms are used to mean the same thing, one of the terms is identified as preferred and the others are listed as aliases. An example of controlled vocabulary is the DCMI Type Vocabulary used in Dublin Core<sup>8</sup>.
- **Taxonomy:** a taxonomy (taxonomy is the science of classification) is a classification that arranges the terms of a controlled vocabulary into a hierarchy (mainly parent-child).
- **A thesaurus** is a networked collection of controlled vocabulary terms. Basically a thesaurus takes taxonomies as described above and extends them by allowing associative relationships (e.g. term A is related to term B) in addition to parent-child relationships.
- The term ontology is often used to mean different things, such as vocabularie, thesauri, taxonomies, schemas, data models, and formal ontologies. Here we refer to formal ontologies as defined in (Studer, 1998), "... *an explicit formal specification of a shared conceptualization. A 'conceptualisation' refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. 'Explicit' means that the types of concepts used, and the constraints on their use are explicitly defined*" or in (Guarino, 1998) "...

---

<sup>7</sup> [http://en.wikipedia.org/wiki/Controlled\\_vocabulary](http://en.wikipedia.org/wiki/Controlled_vocabulary)

<sup>8</sup> <http://dublincore.org/documents/dcmi-type-vocabulary/index.shtml>

Title:

*engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words”.*

An ontology is a means to express the intended meaning of a universe of discourse: it can be seen as a conceptual model, in which concepts are related to other concepts with more expressiveness than in case of UML class models. A formal ontology is expressed in an ontology representation language (e.g. RDF<sup>9</sup>, OWL<sup>10</sup>, etc.), which is machine-readable: this means that reasoning software can apply inference rules in order to integrate different ontologies.

### 3.5 Spatial data and maps

It should be noted that spatial data are not maps: the way we portray/symbolise/visualise spatial data should be independent from the data itself. In traditional cartography this distinction was not applied and in a certain way data were the symbols and vice versa.

Nowadays the approach is different and in all fields of ICT the trend is to separate data from the way they are portrayed. This is the case of XML (eXtensible Markup Language)<sup>11</sup>, a text-based markup language both human and machine readable; XML enable to structure data and share them among different information systems, but a XML document does not carry information on how to display the data. In order to style its content a XML document refers to an external file (stylesheet) which contains rules about how to render the data: these stylesheets can be in the form of CSS<sup>12</sup> or XSL<sup>13</sup> files. This means that the same XML document may be portrayed in different ways depending on the context and purpose of its use. In the area of Geographic Information GML is an XML encoding for spatial data: the same GML file may be portrayed in different ways depending on the purpose of the map visualisation and on local context.

---

<sup>9</sup> [http://en.wikipedia.org/wiki/Resource\\_Description\\_Framework](http://en.wikipedia.org/wiki/Resource_Description_Framework)

<sup>10</sup> [http://en.wikipedia.org/wiki/Web\\_Ontology\\_Language](http://en.wikipedia.org/wiki/Web_Ontology_Language)

<sup>11</sup> <http://en.wikipedia.org/wiki/XML>

<sup>12</sup> [http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets)

<sup>13</sup> [http://en.wikipedia.org/wiki/Extensible\\_Stylesheet\\_Language](http://en.wikipedia.org/wiki/Extensible_Stylesheet_Language)

Title:

## 4 ISO/OGC: standards and profiles

In this chapter a number of ISO standards from the 19xxx series are shortly described. The choice for these specific standards (19109, 19107 and 19123) is made because of their relevance for the modeling of spatial information. The General Feature Model (ISO 19109) provides the foundation and basic principles and influences a large number of other ISO 19xxx standards, and ISO 19107 (Spatial Schema) contains the model elements for geometry and topology.

For ISO 19123 (about coverages) it must be kept in mind that discussion about how to deal with continuous phenomena (how to effectively model and store this type of spatial information) are still going on (in ISO, OGC and elsewhere). Alternative approaches come at the moment from the Earth Observation community. Also in HUMBOLDT we have to look at this discussion (for the Ocean, Galileo and possibly other Scenarios).

### 4.1 The ISO General Feature Model

Figure 4.1 shows part of the General Feature Model (GFM) specified in ISO 19109.

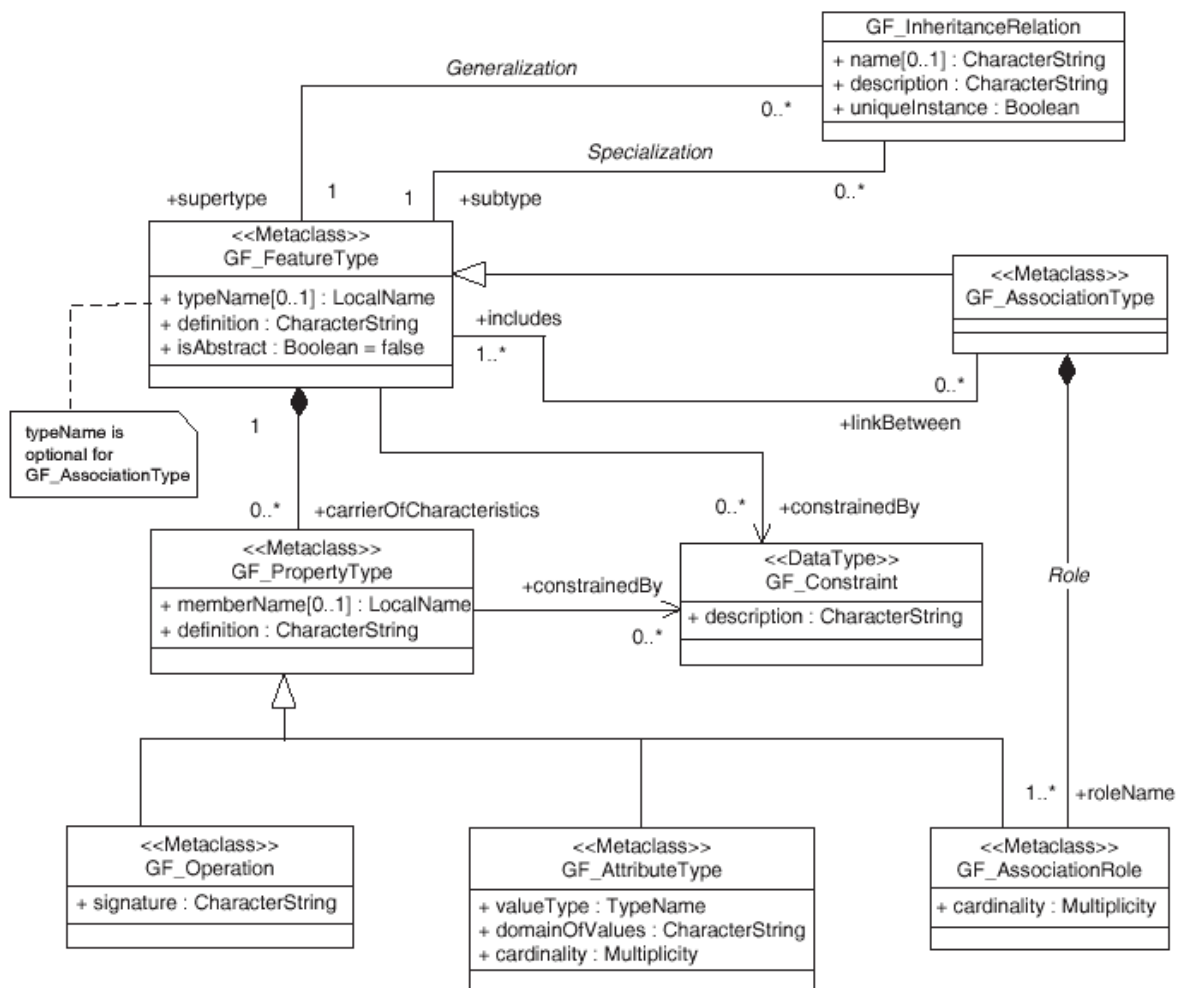


Fig. 4.1 part of the ISO General Feature Model (GFM), source: ISO 19109

Title:

The General Feature Model is a meta-model, which means in this case that concrete data models for spatial information (such as the Scenario data models) are built using the (abstract) building blocks of the meta-model. It is not necessary that a Scenario data modeller knows the details of the GFM or other ISO standards. The data modelling tools that are developed in HUMBOLDT will hide this complexity as much as possible, so that the modeller can focus on the application-specific part of the data models.

In the conceptual data modelling step (the first step, see chapter 3), the relevant information elements for the application are chosen, and grouped into object classes, or in General Feature Model terminology: feature classes (GF\_FeatureType). A feature class has characteristics, called properties: attributes, associations and operations.

An important principle of the General Feature Model is that the geometry of an object is 'just' like other (thematic, non-spatial) properties. There is one difference only: a geometry attribute has another data type than for example a Name (string data type) or a RoadWidth (integer or float data type). The most commonly used 2D geometry data types are: point, line, polygon. Official ISO names for these geometry data types are: GM\_Point, GM\_Curve and GM\_Surface. The 3D equivalent is GM\_Solid. See Figure 4.2.

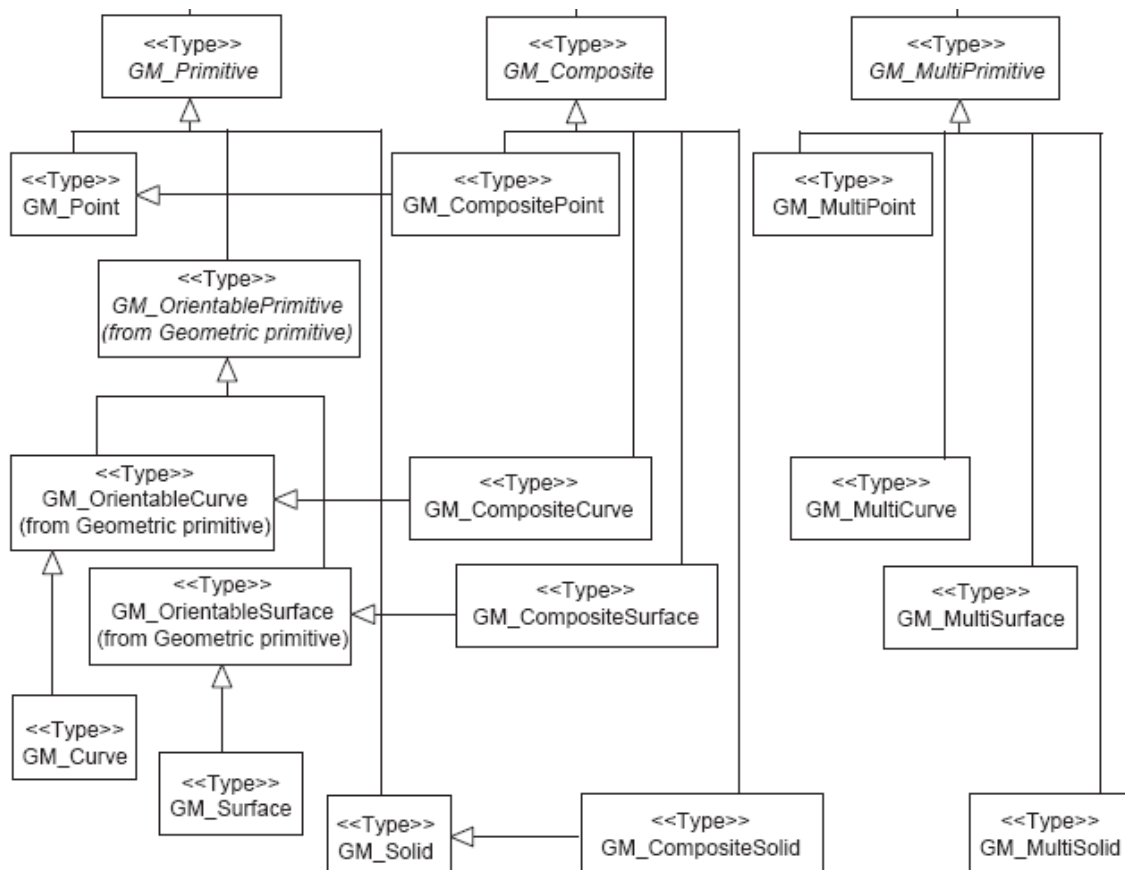


Figure 4.2. Geometry types, source: ISO 19107 Spatial schema (part from figure 5)

Title:

There are also composite geometries, and multi-geometries. The multi-geometry types are sometimes needed when a spatial object can consist of different parts, e.g. a province or other administrative unit that does not consist of a continuous area but has islands before the coast, or a city that is split in two by a river.

The relevance of these geometry types is especially for discrete spatial objects (see chapter 3, the feature-based approach). For continuous phenomena other data structures are needed, see the next section.

ISO 19107 also contains the modelling elements for topology. When topology is important for one or more Scenarios, information will be added to this chapter.

## 4.2 Coverages

For continuous phenomena the coverage data type is commonly used, also increasingly in the context of Web Coverage Services. As indicated above this is a subject of discussion in a number of projects. Alternatives are netCDF and related data structures, probably for HUMBOLDT also the GRIB data format. More information about the data harmonisation requirements related to coverage data will be collected from the Scenarios, and can when needed be added to this section in an updated version of this document.

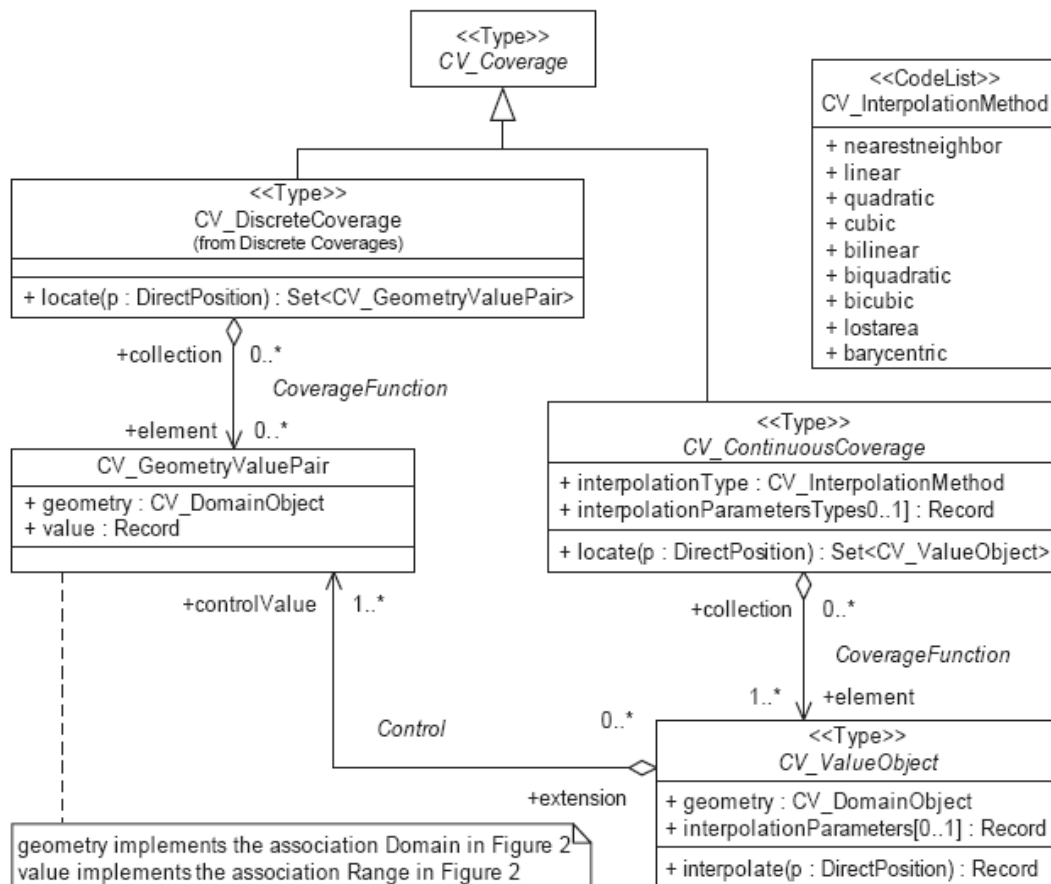


Figure 4.3 CV\_Coverage and its subclasses, source: ISO 19123:2004 (Figure 3)

Title:

### 4.3 The concept of „Profiles“

A “profile” denotes a subset of a larger data model. Working with profiles has the advantage that the overhead in using very large, all-embracing data models is reduced.

Also in HUMBOLDT profiles (in the sense of subsets) are used: a profile for the metadata (see chapter 3), and a profile for the range of geometry types that are supported in the HUMBOLDT software. This latter profile can be extended later, but at the moment contains the following geometry data types (it is a small subset of ISO 19107):

GM\_Point, GM\_Curve, GM\_Surface, GM\_MultiPoint, GM\_MultiCurve, GM\_MultiSurface, GM\_Envelope / LinearRing

(Figure 4.4 shows the corresponding names of these ISO geometry types in GML (Geography Markup Language), the ISO/OGC standard for encoding feature-based vector data.)

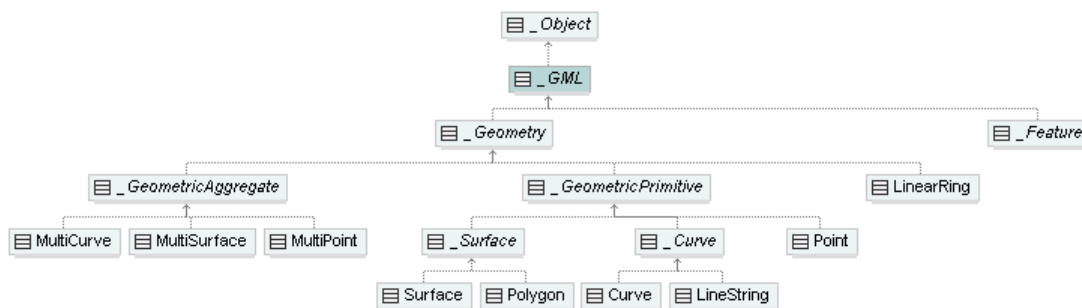


Figure 4.4 part of Simple Features Profile of GML (Geography Markup Language)

Title:

## 5 INSPIRE

Important for the creation of the harmonised, common data models for the HUMBOLDT Scenarios is the work of the INSPIRE Drafting Team on Data Specifications. The main sources for this chapter are the documents DS-D2.5 (Generic Conceptual Model), and DS-D2.6 (Methodology for the development of data specifications).

In Chapter 2 of this document we have distinguished between two scopes of data harmonisation in the Scenarios: harmonising the data of all data providers in a Scenario, and secondly, making sure that also between Scenarios data exchange is possible.

Also in the INSPIRE documents of the DS-DT we see this split: requirements and recommendations for the data specifications per Theme and, secondly, guidelines for the more generic parts of the data harmonisation work, such as recommendations for the so-called „Consolidated INSPIRE UML model“, that must ensure consistency and interoperability between the Themes.

### 5.1 Generic Conceptual Model

The INSPIRE Data Specifications DT has defined the „Generic Conceptual Model“ (GCM) as a foundation for the work on the Annex Theme data specifications in the respective Theme Working Groups.

The Generic Conceptual Model is not a ready-to-use common data model for INSPIRE, but a combination of:

- requirements and recommendations for the creation of the Theme data specifications;
- a set of base types to use in the Theme data models (called application schemas in INSPIRE).

### 5.2 Data specifications for the Annex Themes

Two core artifacts of the data specification of each INSPIRE Theme are: the Application schema, and the Feature catalogue.

#### 5.2.1 Application schema

In the INSPIRE context UML (Unified Modeling Language) is used as the Conceptual Schema Language for developing and defining the Theme data models.

In the INSPIRE documents the term “application schema” is used as a synonym for “application-specific data model”:

As example a fragment of the Cadastral Parcels data model (Annex I Theme) is shown in Figure 5.1 (source: D2.8.I.6, Data Specification on Cadastral Parcels – Guidelines, v3.0.1).

Title:

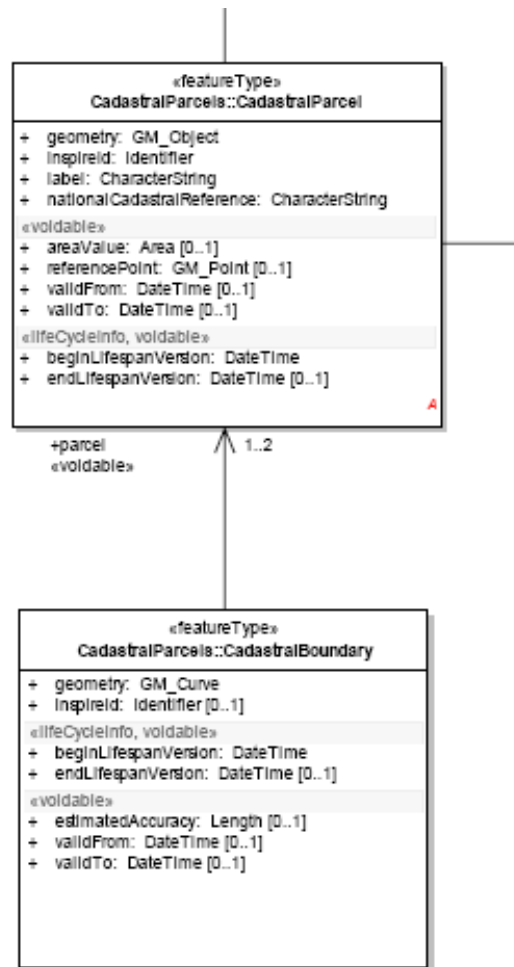


Figure 5.1. Part of UML application schema for Cadastral Parcels (D2.8.1.6)

The usual components of a UML class diagram can be seen in this example (also see Chapter 3):

- classes, attributes, associations;
- code lists (not in this fragment) and enumerations. Difference between the two is that an enumeration contains a limited list of permitted values for an attribute, while a codelist contains “suggested” values: the data provider for a dataset that conforms to that application schema can, but does not have to choose values from the codelist (also see ISO 19136);
- methods / operations that can be performed on (attributes of) the class, for example to calculate derived data such as ‘computedArea’ or ‘forecastedTrajectory’;
- additional constraints, expressed either with OCL (Object Constraint Language) or in natural language as part of the documentation.

As stated above, DS-D2.5 gives a series of requirements and recommendations for the specification of the UML application schemas per Theme. See Annex A of this A7.1 document for a copy of the list.

Title:

## 5.2.2 Feature catalogue

The second way to describe a data model in INSPIRE is by using a Feature Catalogue (according to the ISO 19110 standard).

<b>CadastralParcel</b>	
Definition:	Areas defined by cadastral registers or equivalent.
Description:	SOURCE [INSPIRE Directive:2007].
	NOTE As much as possible, in the INSPIRE context, cadastral parcels should be forming a partition of national territory. Cadastral parcel should be considered as a single area of Earth surface (land and/or water), under homogeneous real property rights and unique ownership, real property rights and ownership being defined by national law (adapted from UN ECE 2004 and WG-CPI, 2006). By unique ownership is meant that the ownership is held by one or several joint owners for the whole parcel.
Status:	Proposed
Stereotypes:	«featureType»
<b>Attribute: areaValue</b>	
Value type:	Area
Definition:	Registered area value giving quantification of the area projected on the horizontal plane of the cadastral parcel.
Multiplicity:	0..1
Stereotypes:	«voidable»
<b>Attribute: beginLifespanVersion</b>	
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:	«lifeCycleInfo,voidable»
<b>Attribute: endLifespanVersion</b>	
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:	«lifeCycleInfo,voidable»
<b>Attribute: geometry</b>	
Value type:	GM_Object
Definition:	Geometry of the cadastral parcel.
Description:	As much as possible, the geometry should be a single area.
Multiplicity:	1
<b>Attribute: inspireId</b>	
Value type:	Identifier
Definition:	External object identifier of the spatial object.

Figure 5.2. Part of Cadastral Parcels Feature catalogue

A feature catalogue contains about the same information as the UML model, but in another form. The fragment in Figure 5.2 shows the part of the Cadastral Parcels Feature catalogue that describes the spatial object type “CadastralParcel”.

Title:

### **5.2.3 Components of an INSPIRE Theme data specification**

Chapter 7 of INSPIRE D2.6 document “Methodology for the development of data specifications” provides a list of preferred components of a Theme data product specification. The list is based on ISO 19131:

1. General: the specification scope or scopes (there can be more per Theme)
2. Identification information
3. Content and structure
4. Reference system
5. Data quality
6. Data capture information (optional)
7. Maintenance information (optional)
8. Portrayal information (optional)
9. Delivery: delivery medium and delivery format (Note: covered by the implementing rule on download services and by the guidelines for the encoding of data, i.e. D2.7)
10. Additional information (optional): “all kind of requirements that are not predefined in ISO 19131 or underlying standards”
11. Metadata
12. Service specifications (optional)

Compared to the INSPIRE list of harmonisation components (see Chapter 2) there is, not surprisingly, much overlap. There is as a consequence also overlap with the checklist that is used in WP7/WP9 to collect the harmonisation requirements (see Chapter 6), except for items 6 and 7, although depending on the specific HUMBOLDT Scenario remarks about data capture and maintenance can be relevant.

## **5.3 INSPIRE base types**

The Generic Conceptual Model also contains a set of “base types” that should be used when building the Theme application schemas. These base types can be used in the data models of the INSPIRE Themes.

The INSPIRE base types that are at the moment specified in the Generic Conceptual Model / base application schema are:

- SpatialDataSet: a spatial data set is an identifiable composition of other spatial objects.
- Identifier: a complex data type that consists of three attributes that in combination provide a unique object identifier
- ConditionOfFacilityValue: a codelist with values that indicate the „status of a facility with regards to its completion and use“

Title:

- VerticalPositionValue: an enumeration of values that indicate the „relative vertical position of a spatial object“
- VoidReasonValue: a codelist with reasons for void values. At the moment two reasons for a null value are distinguished: ‚Unknown‘ and ‚Unpopulated‘.

The two codelists (for ConditionOfFacilityValue and VoidReasonValue) are centrally managed in a INSPIRE code list register.

When creating the harmonised data models for the Scenarios in WP7 especially the Identifier data type has been used (e.g. in the ERisKA, the Protected Areas and the Transboundary Catchment data profiles).

Figure 5.3 shows the INSPIRE base types in a class diagram.

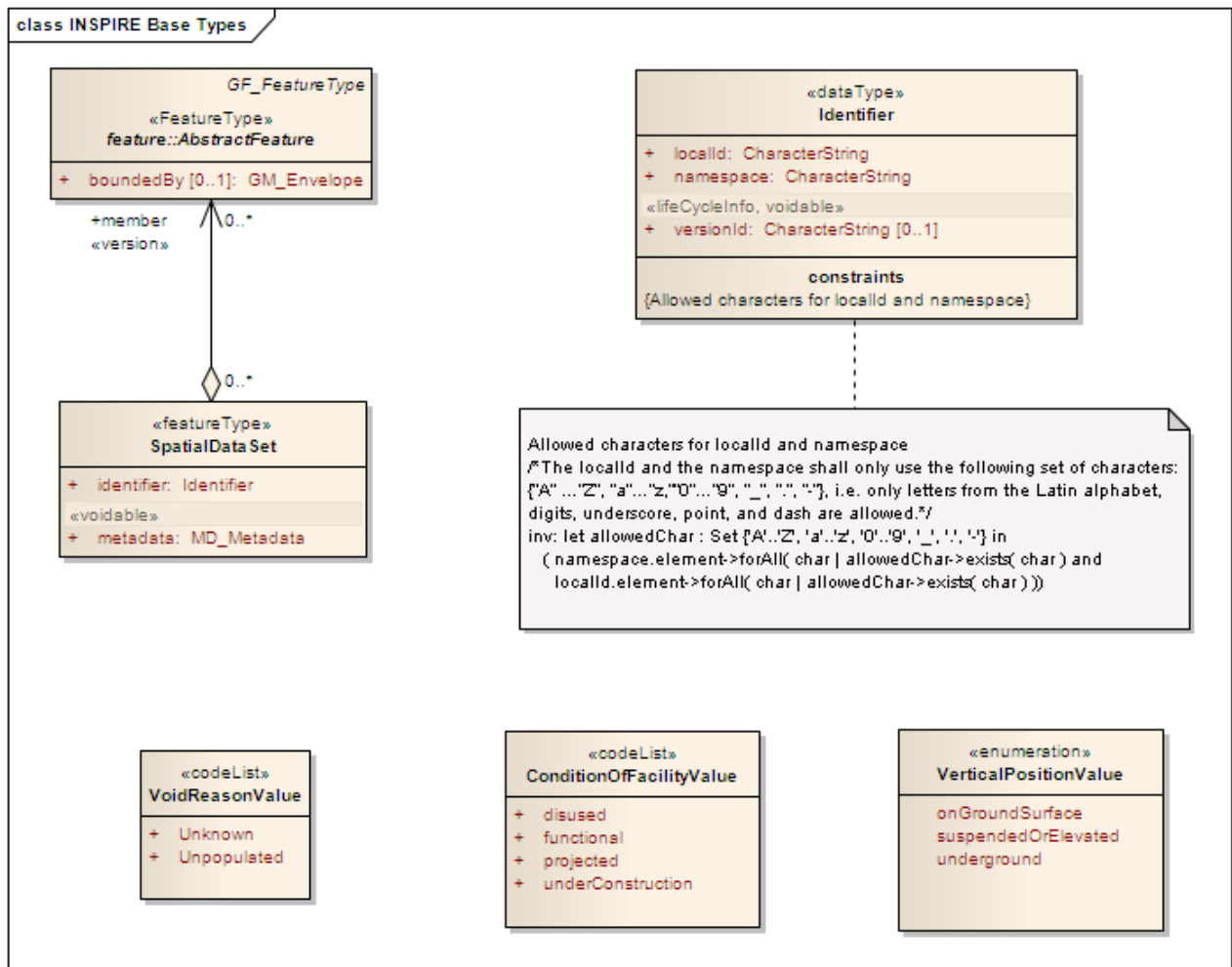


Figure 5.3. INSPIRE base types (DS-D2.5)

Title:

## **5.4 Consolidated INSPIRE UML model**

The INSPIRE “base types” provide a “minimum” set of common data model elements between the Theme application schemas. The text of the DS2.5 document proposes the following strategy for cross-Theme harmonisation: after the drafting teams for the Themes have created the data specifications per Theme, then the following step is to move parts of the application schemas per Theme “up” towards the so called Consolidated INSPIRE UML model. In case of conflicts (when a spatial object type exists in more than one INSPIRE Theme) a harmonisation decision has to be made.

Suggestions on how to do this are provided by a number of recommendations.

In this context (and with the same purpose to harmonise between the 34 Themes) also another component of the INSPIRE data specification framework is relevant, i.e. the Feature Concept Dictionary Register, based on ISO 19126. In ISO 19126 a Feature concept dictionary is used to define concepts at a high abstraction level, without decisions about implementation: attributes (properties) are defined as concepts that are not bound to a specific class (feature type).

In INSPIRE the Feature Concept Dictionary Register is meant to centrally manage the names, definitions and descriptions of all spatial object types used in INSPIRE application schemas. According to the text of D2.5, in the future the register may be extended to manage not only feature types in one register, but also properties.

Title:

## 6 Harmonisation issues

The WP7/WP9 reports for each Scenario contain a list of relevant harmonisation issues and the approach taken to solve these. In this chapter, we shortly present these issues.

The numbering of the sections is the same as in the checklist in the WP9 reports (i.e. Chapter 5 in most of these reports). The same list appears in the WP7 reports with the data specification for each Scenario.

Remark: there is strong relation between especially the items 3.4 (data model), 3.5 (classifications used) and 3.6 (terms and concepts). In fact the choice of classifications to use, is part of the specification of the harmonised data model. And in 3.6, an ontology can be seen as a high-level conceptual model of the information used in the Scenario.

### 6.1 Data format and/or type of web service

Within the GIS/CAD world or the Earth Observation world, differences in data format do not have to be an issue any more with current geo-ICT technology. Most GIS/CAD systems can import and export also the data formats of other GIS/CAD software. Secondly, when web services are used to publish geodata, the web services act as wrappers around the native formats.

However, there is an issue in those Scenarios where GIS data and Earth Observation data must be made interoperable; combining the data formats or web services of these two communities in one Scenario is not trivial.

This issue seems especially relevant for the Ocean Scenario.

### 6.2 Spatial reference system

To check the relevance of this issue it is necessary to check, as early in the Scenario development as possible, what the EPSG numbers are of the spatial reference system of the datasets used. See the website: <http://www.epsg-registry.org/>

If coordinate transformation to/from that spatial reference system is supported in current tools, or in existing Web Coordinate Transformation services, then there will be no major problem to convert datasets to another (supported) coordinate system.

Also in the case of a geographical grid, having good documentation in an early stage is important.

### 6.3 (Conceptual) data model

Also because ISO and INSPIRE use UML as conceptual schema language, each Scenario specifies their common data model in one or more UML diagrams: class diagrams, at the conceptual level, or data model diagrams as the logical level, that also contain primary and foreign key constraints.

Some examples of UML data models used in the scenarios are given in Figure 6.1 (part of the Border Security data model) and Figure 6.2 (Atmosphere Scenario).

Title:

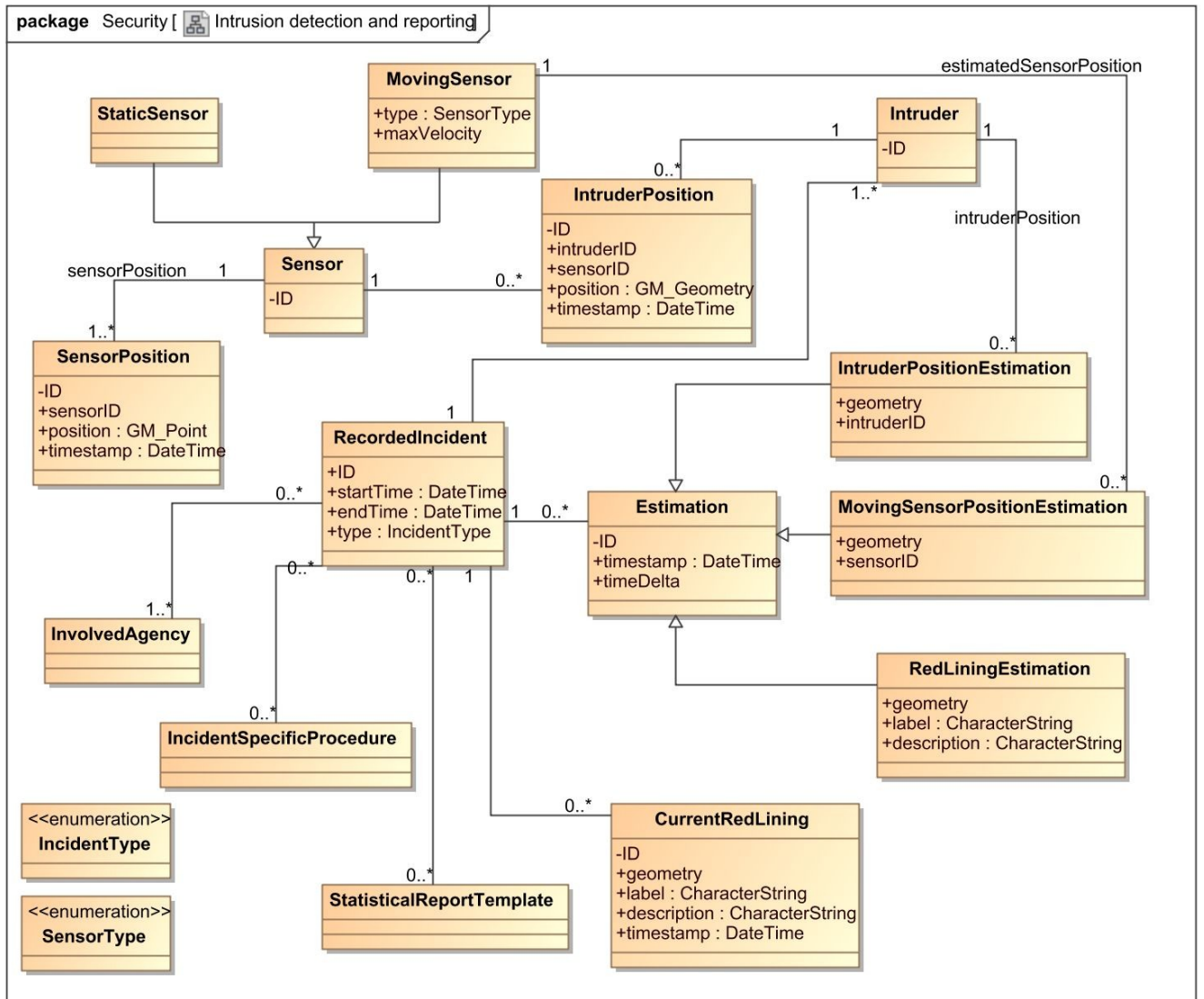


Figure 6.1 Border Security: Intrusion detection and reporting package

Title:

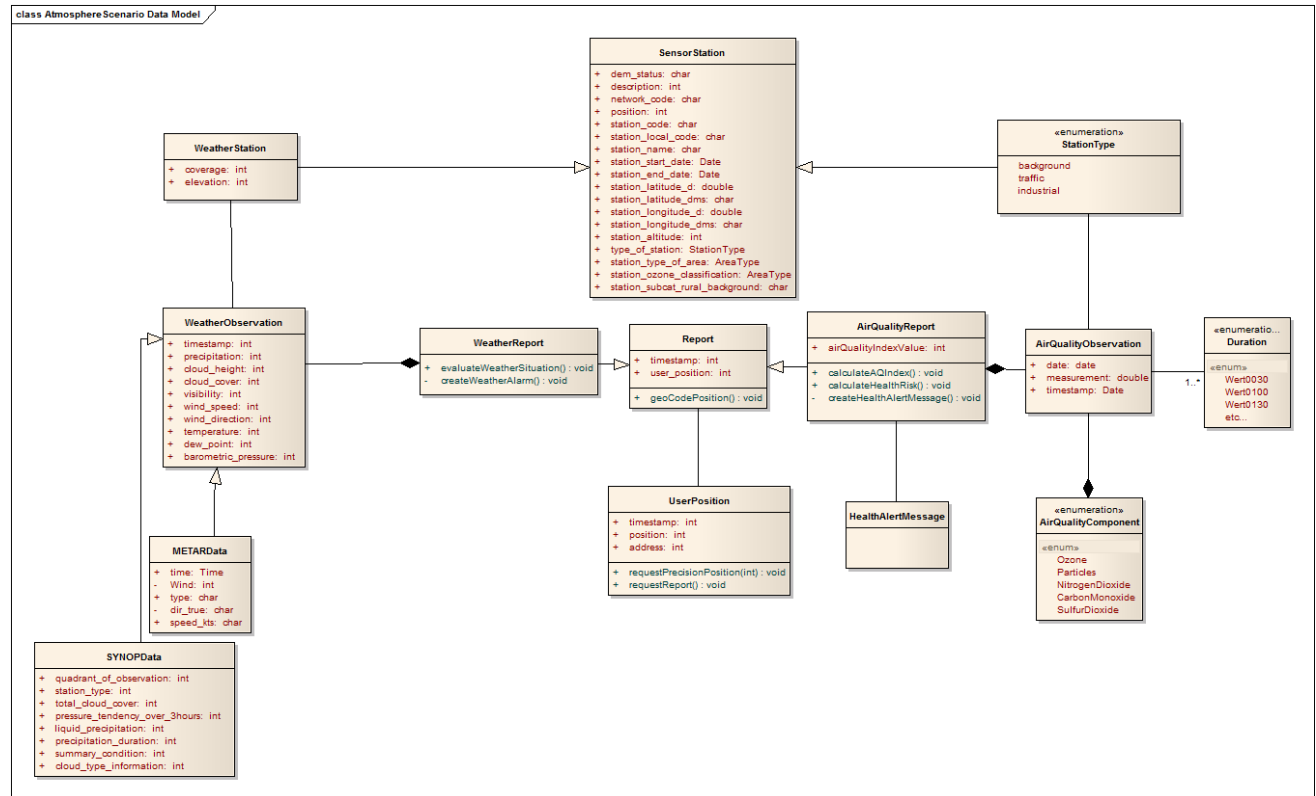


Figure 6.2: Atmosphere scenario data model

It is not intended that the Scenario developers have to deal with the physical data models. These (e.g. the XML/GML application schemas) can be generated from the conceptual/logical data model by HUMBOLDT framework tools (or by existing GIS/CAD tools when they have the required functionality). Specifying the conceptual and/or logical data model should have the focus.

Specifying the harmonised common data model in the Scenario is not something “out of the blue”. The existing pre-harmonised data sources are taken into account. Otherwise the risk is that a common, target data model might be designed that is not feasible or practical.

The design of the Scenario data model went through a number of steps: the first step is to decide which information is used (processed, published, exchanged) in the Scenario, based on the use cases that have been specified; the second step is to sketch a first data model (interim version). This first sketchy model contained all object types (classes) that are relevant, with their most important attributes and associations. The final data models were extended with respect to a broader scope.

Specifying the data model is an iterative process. The most general guidelines about this process are:

- Define use cases and study the information needs;
- Define important features (classes), their attributes and relationships between the classes;
- Investigate standardisation organisations (ISO, OGC, etc.) and projects (INSPIRE, GMES, etc.) dealing with the same or similar application domains;
- Decide on a common vocabulary (data dictionary, see section 6.5), classifications schemes (see section 6.4), codelists etc.

Title:

- Define common portrayal rules (see section 6.8) since they may influence the geometric representation;
- Draft the common data model as UML class diagram;
- Investigate available data sets and identify possible problems with transforming the data to the common target data model;
- Possibly adapt the common data model with respect to available / exemplary data sets (at least in a prototype situation as is the case in HUMBOLDT).

Role of specifying a (conceptual) data model in each Scenario is (also see Chapter 2):

- Helps the information analysis phase in the Scenario development, and is one of the results of that phase;
- Necessary for (model-driven) data transformation from old, „source“ data model to new, harmonised „target“ data model;
- Data models can also be part of the metadata that is published by the Scenarios catalogue services.

## 6.4 Classification schemes

With classification scheme is meant a classification such as the CORINE land cover classification, or the NUTS classification of administrative units.

It is also very well possible that classifications are part of more extensive content standards e.g. of the ocean community, or the emergency response community.

It is very important in the Scenario to agree on the classifications to use. Classifications strongly influence the data model, in either of two ways: classifications can be implemented as a hierarchy of classes and subclasses (and an inheritance relation or an aggregation relation), or as an attribute with a (hierarchical) codelist or enumeration type that lists the permitted attribute values.

INSPIRE document DS-D2.3 (Definition of Annex Themes and Scope) contains a first inventory per INSPIRE Theme of existing, possibly relevant classification systems.

Often these classification standards for a certain application domain (topic) are maintained on web sites of European agencies. The list below is based on DS-D2.3.

NUTS	Nomenclature of Territorial Units for Statistics (EUROSTAT)	5.4	Administrative units
ISO 3166	Country codes ISO 3166	5.4	Administrative units
LAU	Local Administrative Units	5.4	Administrative units
CDDA	Common Database on Designated Areas <a href="http://dataservice.eea.europa.eu/dataservice/metadetails.asp?id=1017">http://dataservice.eea.europa.eu/dataservice/metadetails.asp?id=1017</a>	5.9	Protected sites
CORINE land cover	biophysical land cover (44 class nomenclature) <a href="http://terrestrial.eionet.europa.eu/CLC2000">http://terrestrial.eionet.europa.eu/CLC2000</a>	6.2	Land cover

Title:

CORINE biotopes	A classification of European habitat types used to identify Special Areas of Conservation (SAC) under the EC Habitats Directive (92/43/EEC)	7.18	Habitats and biotopes
LCCS	Land Cover Classification System (FAO/UNEP) <a href="http://www.glc-lccs.org/index.php?name=Content&amp;pa=showpage&amp;pid=3">http://www.glc-lccs.org/index.php?name=Content&amp;pa=showpage&amp;pid=3</a>	6.2	Land cover
EUNIS Habitats	EUNIS Habitats 2004 <a href="http://eunis.eea.eu.int/upload/EUNIS_2004_list.pdf">http://eunis.eea.eu.int/upload/EUNIS_2004_list.pdf</a>		

## 6.5 Terms and concepts: thesaurus, ontology

Depending on the Scenario it can be relevant to create a thesaurus or other kind of data dictionary for the definition and explanation of terminology in the application field. The role of the thesaurus in the HUMBOLDT Scenarios is to precisely define the terminology used; this is important as part of the documentation (for the human eye), but can also be used in the HUMBOLDT Scenario portal (see D91.D1) to make searching for datasets and interpreting data more easy for users that are not familiar with the vocabulary of that Scenario.

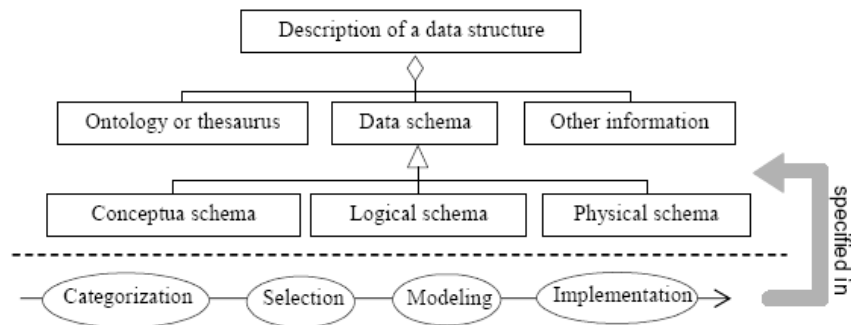


Fig. 6.1 Ontology and thesaurus as part of data specification (Balley et al. 2006)

Strictly speaking an ontology is not about terms but about concepts. In an ontology concepts are defined not in words (although there are comment/description possibilities), but by defining relations with other concepts in a formal way, that is „machine treatable“.

An ontology can be seen as a high-level conceptual model: it is a more conceptual way than UML to model a „universe of discourse“ (i.e. how experts in a certain application domain look at „their“ real-world).

Also in the second stage in the data (model) harmonisation process, when the target data model has been specified and the data needs to be transformed to that target model, ontologies can be put to work:

- The Scenario ontology could be used for Model2Model mapping (under investigation in WP7 and WP5): an ontology language such as OWL has the ‘equivalence’ operator, which makes it a candidate for specifying mappings between source and target data model;
- The classifications that are used in the Scenario (such as CORINE or NUTS) can be turned into an ontology, which offers possibilities for (on-the-fly or offline) re-classification, as part of a model-driven data transformation process.

Title:

## **6.6 Metadata profile**

In the discussions about metadata in the context of Digital Libraries, the concept of „metadata profile“ is already in use for a long time. Every scientific community or industry sector has its own relevant set of metadata key words to describe datasets or other kinds of resources. The same is true in HUMBOLDT: depending on the Scenario other metadata elements can be relevant for that Scenario.

For cross-Scenarios use of data in HUMBOLDT a minimum set of metadata is necessary. Therefore, in WP7, a core (“harmonised”) HUMBOLDT metadata profile has been specified. This profile contains the metadata elements that are relevant to all Scenarios, see: 0774-humboldt\_metadata\_report-ulsor-003-final.pdf.

The document also contains a comparison of the INSPIRE metadata implementing rule, in relation to the ISO specifications.

Because of the differences between Scenarios, every Scenario can decide to extend this core HUMBOLDT metadata profile, with their own specific metadata elements.

Part of the Scenario information model is therefore also the metadata profile for that Scenario (see Chapter 7): the mandatory and optional metadata elements that describe the datasets in such a way that the datasets can be discovered by metadata search services, and their quality, usefulness and relevance can be assessed by users in and outside the Scenarios.

Role of metadata in the HUMBOLDT Scenarios:

- Published by metadata registries (catalogue services) in the HUMBOLDT framework;
- Can also be included in the datasets themselves (exchange files or output streams of a data service);
- In the information analysis and GAP-analysis phase: when datasets are explored for the harmonisation issues inventory and GAP-analysis, it helps when metadata is available.

## **6.7 Aggregation, multiple representation**

In reporting it is often necessary to aggregate (geo)data from a more detailed to a more aggregated level. The following example is taken from the INSPIRE drafting team document DS-D2.6 (Methodology for the development of data specifications).

Title:

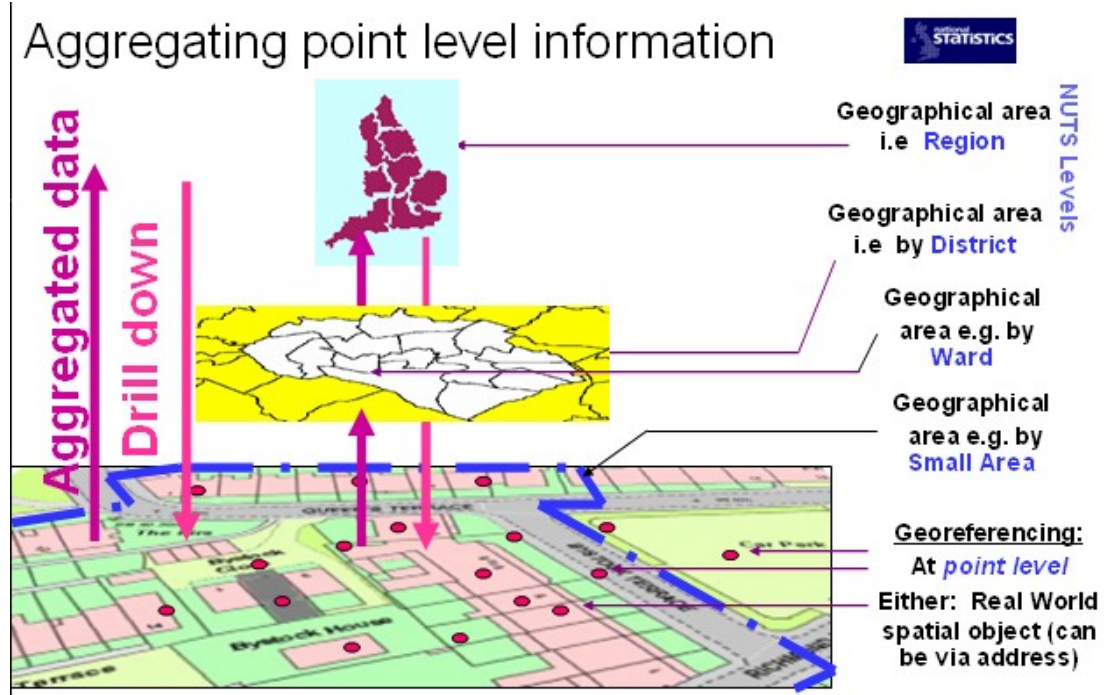


Fig. 6.2 Aggregated Spatial Objects for Statistical Reporting (example Ordnance Survey) (DS-D2.6)

If aggregation for reporting or multiple representation is relevant in the Scenario, this can also have influence on the Scenario data model.

## 6.8 Portrayal rules

When geodata of different providers is combined and visualised in the same (Web) client, it can be important to have harmonised portrayal. For example, on a road map of Europe, colors and map symbology in the different European countries should be consistent: a highway should have the same color and linewidth on both side of a national border.

The step from specifying a common data model for the Scenario to deciding on common portrayal rules is a small step. One could say the latter follows from the former. Why is portrayal then mentioned in this list as separate harmonisation issue? Because the common portrayal rules still have to be established and specified, although the data itself is harmonised. This follows the principle of separation between Digital Landscape Model (the data itself) and Digital Cartographic Model (the graphical representation): the cartographic rules are not part of the data.

Some information communities have international standards for visualisation, e.g. the IHO, see: <http://www.iho.int/PUBLICATIONS/download.htm#special>

A Scenario may also adopt an existing portrayal standard.

Therefore, if applicable in the Scenario, also portrayal rules are part of the Scenario information model. These portrayal rules include: Legend classification rules (what Legend classes should be used), and agreement on colors and other cartographic aspects.

Title:

Role of portrayal rules in the HUMBOLDT Scenarios: to harmonise the visualisation of spatial data. This applies to 2D maps, 3D city models, but also to derived / processed raster images.

## **6.9 Processing functions**

If there are special operations on the data (to derive forecast models for example, or to compute land-cover classification based on remotely-sensed images) then it can be necessary to harmonise also these computational functions (same parameter values e.g.).

## **6.10 Multi-linguality**

Multi-linguality has not been an issue during the development of the harmonised Scenario data models, because the choice of language for feature type and attribute names and for the values in codelists and enumerations has been English. For the construction of UI's in web applications etc. it can be an issue, but this falls outside the scope of this document.

Title:

## 7 Summary

1. What is an “application-specific harmonised data model” (in the context of HUMBOLDT)? We defined this concept as follows: a data model that
  - Describes and prescribes structure and possible content (e.g. through constraints on attribute values) of the (combined) data sources used in that Scenario
  - Uses ISO/OGC and other de-facto standards to ensure syntactic interoperability
  - Also provides definitions and semantic relationships between concepts to enhance semantic interoperability (also see question 4)
2. What do INSPIRE and ISO/OGC offer us?
3. What more is needed for the Scenario data specification, besides the harmonised Scenario data model?

In document 9.1D1 the term ‘Scenario information model’ was introduced, as a combination of necessary and optional parts of the Scenario data specification. We repeat the items here, and add one (documentation about spatial reference system, units of measurement, and data quality aspects). (The choice was made not to use the ISO 19131 list of data specification components.)

The Scenario information model shall contain these items:

- The overall, harmonised data model for that Scenario
- Documentation about the classifications that is used in the Scenario. These can be international/European standard classifications, such as the CORINE land cover classification, or the NUTS classification of administrative units in Europe, but also classifications specific to that Scenario.
- A glossary or thesaurus with terms and definitions. If multi-linguality is an harmonisation issue in the Scenario this could be a multi-lingual thesaurus. Another option is to use techniques from the semantic web, and build a Scenario ontology.
- A metadata profile for the Scenario. This profile can be the same as the HUMBOLDT core metadata profile (see Section 6.6), but it can also be an extension of that profile.
- If applicable: rules for portrayal (colors, style, legends) for map representation of (vector) data.
- Documentation about the spatial reference system in the Scenario, units of measurement, agreement on scale/resolution, required data quality. (Whether an ISO 19131 data product specification is a suitable format is not certain yet.)

Title:

## 8 References

Balley, S., B. Bucher, et al. (2006). A service to customize the structure of a geographical dataset, Second International Workshop on Semantic-based Geographical Information Systems (SeBGIS'06), Montpellier, France

Burrough P., McDonnell R. Principles of Geographical Information Systems. Oxford University Press 1998.

EPSG (2008). EPSG Geodetic Parameter Registry, version 6.15. Online: <http://www.epsg-registry.org/>

Hakimpour, F. and A. Geppert (2001). Resolving semantic heterogeneity in schema integration. 2nd International Conference on Formal Ontology in Information Systems (FOIS 2001), Ogunquit, Maine, USA, ACM.

Heywood I., Cornelius S., Cerver C. An introduction to Geographical Information Systems. Addison Wesley Longman 1998

HUMBOLDT Core Metadata, Pasquale Di Donato (ed.), 0774-humboldt\_metadata\_report-ulsor-003-final.pdf

INSPIRE D2.3, Definition of Annex Themes and Scope, v3.0

INSPIRE D2.5, Generic Conceptual Model, v3.3

INSPIRE D2.6, Methodology for the development of data specifications, v3.0

INSPIRE D2.8.I.6, Data Specification on Cadastral Parcels – Guidelines, v3.0.1

ISO 19107:2003 Geographic information - Spatial schema

ISO 19109:2005 Geographic Information - Rules for Application Schemas

ISO 19110:--, Geographic Information – Methodology for feature cataloguing (to be published, an amendment to EN ISO 19110:2006 is currently at Committee Draft stage)

ISO 19115:2005, Geographic Information – Metadata

ISO 19115/Cor.1:2006, Geographic Information – Metadata, Technical Corrigendum 1

ISO CD 19123 Geographic information - Schema for coverage geometry and functions

ISO 19131:2007, Geographic Information - Data Product Specifications

ISO/TS 19139:2007, Geographic Information – Metadata – XML schema implementation

Klien E., 2007. A Rule-Based strategy for the semantic annotation of geodata. Transactions in GIS, 11(3): 437-452

Kuhn W., 2005. Geospatial semantics: Why, of What, and How. Journal of Data Semantics III: 1-24

Longley P., Goodchild M., Maguire D., Rhind D. Geographic Information Systems and Science. John Wiley & Sons 2001

Studer R., Benjamins R., Fensel D., 1998. Knowledge engineering: principles and methods. *Data and Knowledge Engineering*, 25 (1-2): 161-197

Worboys M. GIS. A computing perspective. Taylor&Francis 1995

Title:

## **ANNEX A: INSPIRE D2.5 - Modelling application schemas**

The following table contains a list of the requirements and recommendations related to the modeling of application schemas as specified in the D2.5 Generic Conceptual Model, Version 3.3. It is an almost integral copy, however in some cases there is a reference to the text of the document itself (in case of figures and longer examples).

### ***INSPIRE D2.5 Generic Conceptual Model: 9.4 Modelling application schemas***

**Requirement 8:**

Every INSPIRE application schema shall contain a comprehensive and precise description of its spatial object types.

NOTE 1 For the avoidance of doubt, “comprehensive” is meant as “comprehensive as required by the scope of the INSPIRE data specification”, i.e. an INSPIRE data specification for an Annex III theme will in general require less detail than an INSPIRE data specification for an Annex I/II theme.

**Requirement 9:**

Every INSPIRE application schema shall conform to the General Feature Model as specified in ISO 19109 7.3-7.7.

EXAMPLE Spatial object types are realisations of GF\_FeatureType and are modelled as classes with the stereotype <<featureType>>. Constraints (realisations of GF\_Constraint) are modelled in invariant OCL expressions in the context of the class representing the spatial object type and are additionally described in natural language. Etc.

**Requirement 10:**

Every INSPIRE data specification shall include one or more INSPIRE application schemas modelled according to ISO 19109 Clause 8, with particular attention to 8.2.

NOTE 2 ISO 19109 8.2 specifies a number of requirements necessary for the management and unambiguous interpretation of an application schema including:

- the use of a conceptual schema language;
- the modelling data structures so that data transfer of all relevant information is supported;
- the provision of a name and a version of the application schema;
- the provision of a corresponding feature catalogue;
- the provision of sufficient documentation for all elements in the application schema;
- the provision of references from a spatial object type to the type definition in the corresponding feature catalogue (in INSPIRE: the INSPIRE Feature Concept Dictionary Register);
- the correct integration of the application schema with the standard schemas or other application schemas.

**Requirement 11:**

Every spatial object type specified in an INSPIRE application schema shall be drawn from feature type concepts in the INSPIRE Feature Concept Dictionary Register with status “valid” or proposed as a new register item when no adequate spatial object type already exists.

**Requirement 12:**

If no related concept exists in the INSPIRE Feature Concept Dictionary Register, that can be reused or amended, a concept from another international feature concept dictionary or feature catalogue shall be reused and proposed for adoption in the INSPIRE Feature Concept Dictionary Register, if possible.

In other words, whenever possible, a concept in an INSPIRE application schema shall be drawn from an established dictionary.

Title:

EXAMPLE An example for such an established dictionary is the DFDD (DGIWG Feature Data Dictionary), see <https://www.dgiwg.org/FAD/registers.jsp?register=DFDD>.

**Requirement 13:**

Spatial object types shall be modelled according to ISO 19109 7.1-7.2, 8.1, 8.5-8.9 and according to the additional rules in Clauses 9-12, 18, and 22 of this document.

NOTE ISO 19109 7.1 and 7.2 describe principles for defining spatial objects and their relationship with application schemas. ISO 19109 8.1 specifies core aspects of the process of modelling application schemas. ISO 19109 8.5 to 8.9 specify the rules for the use of metadata, temporal and spatial characteristics, geographic identifiers as well as the relationship to feature catalogues.

**Recommendation 2:**

To allow that multiple spatial objects representing the same real-world phenomenon but in data sets of different Member States can be explicitly associated, an association to other spatial objects of the same type in an adjacent spatial data set should be modelled in the INSPIRE application schema.

NOTE In general, such links will not be available today in most data sets. However, in the process of harmonising the geography of spatial objects representing the same real-world phenomenon that spans the frontier between two or more Member States - see Article 10(2) of the Directive, such mutual references may be added to the source data sets in the Member States.

EXAMPLE The river Danube runs through several Member States. If requirements to aggregate spatial objects of phenomena that span frontiers exist, the association could be modelled ... with an association (here called “neighbour”) to other River instances in other data sets. Note that although the association is symmetric (if A is a neighbour to B, then B is also a neighbour to A) it is modelled as an unidirectional association so that the property is called “neighbour” in all River instances.

(See D2.5 document (version 3.3), p. 43: Figure 8)

**Requirement 14:**

The profile of the conceptual schema defined in the ISO 19100 series that is used in the application schema shall conform with ISO 19109 8.4.

NOTE 1 ISO 19109 8.4 specifies how adjustments can be made to the standard schemas to add information to the types defined in the standard schemas, for example to specify a new curve segment type not foreseen in ISO 19107 (spatial schema), or to restrict elements of a standard schema as permitted by the conformance clause of the standard that specifies that schema.

**Requirement 15:**

Every INSPIRE application schema shall document the profile to be used for the different properties of spatial object types.

EXAMPLE If a spatial property has the type GM\_Curve then it should be specified in the application schema which curve segment types are allowed. For example, the allowed curve segment types for road centrelines could be restricted to GM\_LineString, GM\_Arc and GM\_Clothoid ...

Note that constraints in the model may be shown as part of a class diagram (...) or they may not be shown in class diagrams but documented in a different way.

(See D2.5 document (version 3.3), p. 44: Figure 9)

**Recommendation 3:**

It is recommended that the simplest profile that addresses the requirements will be used to keep the requirements on software that will process INSPIRE data as low as possible.

NOTE 2 Obviously “simplest profile” is not an absolute term and different stakeholders will have different views on what the simplest profile is. The common understanding what the simplest profile meeting

Title:

the minimal functional requirements will therefore be determined by the implementing rule drafting process involving the registered SDICs and LMOs.

**Requirement 16:**

Basic types as specified in ISO/TS 19103 6.5 shall be used in an INSPIRE application schema, whenever applicable.

EXAMPLE 1 Examples of basic types specified in ISO/TS 19103 6.5 are Integer, Real, Character-String, Boolean, Measure.

**Recommendation 4:**

In the case of an attribute type with coded values, an enumeration or code list should be used. If the set of allowed values may be extended by user communities or without a major revision of the data specification, a code list should be used. If the set of allowed values is fixed, an enumeration should be used. For code lists, the use of a code list managed in the INSPIRE code list register should be mandated.

There are at least two cases where code lists may be more suitable:

- the list of possible values of an attribute are difficult to harmonise and some data providers may have to use sub-sets or extensions of the harmonised list
- the list of possible values is likely to evolve, some other values may have to be added later, either because of new user requirements or because of upgrading of existing data.

To reflect the different characteristics of code lists two types of code list governance are distinguished:

- code lists that are managed centrally in the INSPIRE code list register and only values from that register may be used, and
- code lists that may be extended by data providers as long as data providers maintain their extensions in a national code list register and publish its content.

EXAMPLES 2, 3 and 4: see D2.5 document (version 3.3), p. 45

**Requirement 17:**

If a characteristic of a spatial object may be not present or not applicable in the real world, the property shall be modelled with a minimum multiplicity of "0" and an empty value shall imply that the characteristic is not present or not applicable in the real world.

EXAMPLE 1 If a spatial object type Road would carry an attribute "streetName" and it is determined that not all roads in Europe have a street name, the property would receive a minimum multiplicity of "0".

**Requirement 18:**

If a characteristic of a spatial object may not be present in the spatial data set independent of its presence or applicability in the real world, the property shall receive the stereotype <<voidable>>. If and only if a property receives the stereotype <<voidable>>, the value of void may be used as a value of the property which shall imply that the characteristic is not present in the spatial data set, but may be present or applicable in the real world. It shall be possible to qualify a value of void in the data with a reason using the VoidReasonValue type (see 9.8.4.3).

Void is defined by ISO/IEC 11404 as "an object whose presence is syntactically or semantically required, but carries no information in a given instance."

EXAMPLE 2 Using the spatial object type Road from example 1, if a data set has not captured street names for roads, it would report the streetName property values as void with a void value reason of "Unpopulated".

**Recommendation 5:**

All properties of spatial object types except those without which a spatial object is not meaningful should be voidable.

Title:

EXAMPLE 3 A spatial object type GeographicalName without a name property would not be meaningful and thus it would not be voidable.

EXAMPLE 4: see D2.5 document (version 3.3), pp. 46-47

Recommendation 6:

The candidate standard “Observations and Measurements” should be used in INSPIRE application schemas to model observations and measurements.