*Journal of Visual Languages and Computing* (2002) **13,** 97–116
doi:10.1006/jvlc.2001.0228, available online at http://www.idealibrary.com on IDEAL®

# SpaceML: A Mark-up Language for Spatial Knowledge

MATTEO CRISTANI*, AND ANTHONY G. COHN†

**Dipartimento di Informatica, Università di Verona, Strada Le Grazie 15, I-37134 Verona, Italy, cristani@sci.univr.it and †School of Computing, University of Leeds, LS2 9JT Leeds, U.K., agc@comp.leeds.ac.uk*

*Received 14 April 2001; revised 24 September 2001; accepted 24 September 2001*

We introduce the syntax of a new mark-up language, SpaceML, an evolution of the Geo–ML Language, previously developed by one of the authors, for marking up Geographic Data on the Web. The language is specified in XML by means of tag definitions. It allows the description of three aspects of spatial knowledge: (1) the category attribution to space regions: for instance, whether the region is self-connected, bounded or regular; (2) the ascription of regions to topological, distance, morphological and orientation relations; (3) the definition of a region in terms of its boundary. We provide the grammar of SpaceML, define some constraints on the behaviour of browsers reading SpaceML pages, and specify the tags of SpaceML by an XML DTD, which can be easily translated into an XML Schema. 

*Keywords:* Mark-up language; XML; DTD; Geographical Data; Qualitative Spatial Representation

## 1. Introduction

IN THIS PAPER, WE INTRODUCE a new mark-up language, SpaceML, an extension of Geo–ML, a language previously developed by Cristani [1], for describing geographic objects to which a Web site refers. The language allows:

- the definition of constant spatial objects;
- the definition of spatial categories;
- the definition of spatial regions (in two dimensions, for the moment);
- the definition of variables for each of the above three sorts.

In addition, the language allows for a variety of relations between these entities:

- the mereotopological language, RCC-8, originally introduced by Randell and Cohn [2] and Randell *et al.* [3];
- the qualitative morphological system MC-4 [4];
- quantitative distances;
- the orientation relations as described in [7];
- the cardinal relations framework [8].

Qualitative spatial reasoning has received an increasing amount of interest in the Computer Science literature. The main reason for this, as already observed by Jonsson and Drakengren [9], is probably that qualitative spatial reasoning has the potential to be applicable to real-world problems, for instance, Geographic Information Systems (e.g. [10, 11]) and Molecular Biology [12, 13].

The specific stress on qualitative reasoning on space, as observed by Zimmermann [14], is justified by the fact that frequently qualitative spatial relations can be treated as efficiently as their quantitative counterparts, but they seem to be closer to the model of relations humans adopt for spatial reasoning [15].

Various relation systems have been studied, which allow the representation of mereo-topological knowledge [11, 16, 17], qualitative distance, by Hernandez *et al.* [6], qualitative morphology by Cristani [18] and orientation, both in general e.g. by Isli and Cohn [7] and in the specific case of Geography, with respect to Cardinal Directions by Frank [5] and Ligozat [8]. An overview of qualitative spatial representations can be found in [19, 20].

Even though qualitative spatial reasoning has a relatively extended literature, in spite of its relatively short history, certain aspects of the discipline have been neglected. In particular, no analysis has yet been performed of the possible extensions of mark-up languages by means of qualitative spatial representation operators.

An interesting parallel research, which is described further in Section 2, has been carried out in the field of Geographic Information Systems and it has come to very similar methodological conclusions. Specifically, Wu *et al.* [21] have been proposing the development of a language, GeoML, which allows Geospatial information representation. Simultaneously, the OpenGIS Consortium is developing the GML language [22], again with similar purposes.

The main objectives of these approaches are:

- the provision of a set of tags to specify spatial objects, in particular to detail the coordinates of the junction points of a polygonal representation of boundary;
- the invention of tags which can be used to specify general information about a geographic object such as its name;
- the definition of specific tags to be processed by an application; for instance, the definition of FeatureCollection for use in the context of engineered applications by ESRI products like ArcView.

The present paper's purpose, instead, is to define abstract and qualitative properties of the spatial objects, though we also provide tags for the description of physical aspects. Our main purpose is thus to propose a language which can be used in the description of objects not based on their shape, but on their relational properties.

In this paper, we discuss various extensions of a mark-up language which can be used to record spatial data. We introduce spatial relations, in the following groups:

- qualitative topological relations;
- quantitative distance relations;
- quantitative orientation relations;
- morphological congruence relations.

We allow the regions to be specified in terms of absolute coordinates of some abstract Cartesian system.

The part of the language used for building new categories is introduced for capturing very general ontological features of space as originally discussed in many papers for instance, by Smith [23], Casati and Varzi [24] and Borgo *et al.* [25, 26].

The stress on XML standardization by Wu *et al.* [21], is something we agree on. The approach of the OpenGIS Consortium is different and goes in the direction of definition of non-standard proprietary languages. We concur with the idea that the more standard a web language is, the easier it can be represented by a document type definition (DTD), a standard notion of XML—see [27] or by an XML Schema[a] [28–30], the simpler it can be processed by Java elementary semantic tasks, and thus the better. In fact, we do not mean to specify a generic language nor a specific DTD or XML Schema; the purpose of the paper is to propose an intermediate approach: a grammar, and a DTD specifying the syntax (potentially specifiable by an XML Schema as well), along with constraints on the behaviour of the semantics. This accords with the philosophy of the Semantic Web [31].

The paper is organized as follows. Section 2 reviews related work in the field of spatial reasoning, and in specialized spatial mark-up languages for the web. Section 3 introduces terminology and definitions, specifically discussing the main properties of the region connection calculus model, often called RCC, in its extended version RCC-8. The syntax of the language SpaceML is presented in Section 4. Section 5 summarizes the results of the paper and discusses further developments.

## 2. Related Work

From the perspective of finding primitives to describe space, a significant effort has been spent in the direction of defining binary relations between spatial regions, which may be used as a model of space in qualitative terms.

The RCC model has been developed by Randell and Cohn [2] and Randell *et al.* [3]. The RCC acronym stands for Region Connection Calculus. The model allows for reasoning at a mereological level by using a restricted set of basic relations and at a topological one by using the set of all basic relations. Thus it can be formulated in two versions, called RCC-5 and RCC-8. The RCC-5 model is a model in which we cannot distinguish the boundary of a spatial region (so that 'discrete' covers both 'disconnected' and 'externally connected'), while in the RCC-8 model this distinction is possible.

The specific idea on which RCC is based is that we can provide lots of interesting observations about space without getting into the quantitative details. The basic notion on which existing models of spatial knowledge representation on the Web are based instead, is that of 'geometric feature', which allows quantitative representations and leaves implicit much purely qualitative data[b]. We cite the OpenGIS Consortium [22] definition:

> A feature is an abstraction of a real world phenomenon; it is a geographic feature if it is associated with a location relative to the Earth. Vector data consists of geometric and topological primitives used, separately or in combination, to construct objects that express the spatial characteristics of geographic features.

[a] DTD is an obsolescent technology; however, everything which can be done by means of DTDs can also be done by means of XML Schemas. If a specification is easier in DTD, then it is worth using it, for the sake of simplicity since it can always be translated into XML schemas.

[b] Note that, in general, the notion of distance between spatial regions can be viewed as a qualitative one, either through the use of a qualitative measuring scale, by comparing pairs of distances.

The two geographic mark-up languages we will refer to henceforth are:

- GML, developed by the OpenGIS Consortium, [22];
- GeoML as specified by Wu *et al.* [21].

In the GML language, the basic notion is one of an OpenGIS Consortium (OGC) Simple Feature, whose representation is the main motivation for developing GML. The topological and distance relations are specified in a generic way, so that GML is a language for transporting Geospatial Information on the Web.

Wu *et al.*, with respect to the GeoML research, claim instead to provide a more qualitative framework, but they fail in the main character of such an approach: the definition of topological and distance relations. The authors do not provide a proper semantics for their language and merely seem to provide a reformulation of GML without properly addressing the requirements of a qualititative language. Our idea is to begin a fresh theory, which starts with the notions of topological and distance relations.

## 3. Terminology and Definitions

This section is devoted to a brief discussion of the following concepts:

- Constraint Processing basic notions;
- the spatial theory RCC-8 [2, 3], which is the framework of reasoning for the topological spatial relations of SpaceML;
- the spatial theory MC-4 [4], which is the framework of reasoning for the morphological spatial relations of SpaceML;
- the cardinal direction framework [5, 8], which is the framework for the cardinal spatial relations of SpaceML;
- the Simple Temporal Constraint Networks [32], the formalism for the distance relations of SpaceML;
- the orientation framework [7];
- the extended Backus–Naur form (EBNF) which is used in Section 4 to introduce the syntax of SpaceML.

The concept of a constraint is commonly used in Database and Artificial Intelligence literature. Let us give a formal definition of what a constraint is. A fuller description of constraints can be found in [33].

**Definition 1.** Given a set $D$, a binary relations $R \subseteq D \times D$, and two variables $x_1$ and $x_2$ over $D$, an expression $x_1 \, Rx_2$, is said to be a (binary) constraint on $D$ for $x_1$ and $x_2$.

For any set of variables $V = \{x_1, x_2, \dots, x_n, \dots\}$, and any set of values $W$ of a domain $D$, $W = \{d_1, d_2, \dots, d_n, \dots\}$, the collection of pairs $\{(x_1, d_1); (x_2, d_2); \dots; (x_n, d_n); \dots\}$ is said to be an assignment for $V$. We assume that for any $i$ we assign $d_i$ to $x_i$. When no confusion arises we refer to $W$ as an assignment. For any constraint $C = x_i R x_j$, an assignment $\{d_i, d_j\}$ is valid for $C$ iff the pair $\langle d_i, d_j \rangle$ is in $R$.

A finite set of constraints $\mathscr{C} = \{C_1, C_2, C_3 \dots, C_n\}$, is interpreted as $\bigwedge_{i=1}^{n} C_i$.

**Definition 2.** A finite set of constraints $\mathscr{C} = \{C_1, C_2, C_3, \dots, C_n\}$, where for any $1 \leq i \leq n$ $C_i = x_{j_i}, R_i x_{k_i}$, is consistent iff there exists an assignment for the variables

$\{x_{j_1}, x_{j_2}, \ldots, x_{j_n}, x_{k_1}, x_{k_2}, \ldots, x_{k_n}\}$, which is valid for each constraint in $\mathscr{R}$. Such an assignment is said to be a solution for $\mathscr{C}$.

The problem of deciding whether a constraint set $\mathscr{C}$ is consistent, is called consistency checking. Given an algebra $\mathscr{A}$, the problem of consistency checking in $\mathscr{A}$ is indicated as SAT($\mathscr{A}$).

We now briefly present the Region Connection Calculus, indicated henceforth by RCC, developed by Randell and Cohn [2] and Randell *et al.* [1, 3], which can be shown to be essentially equivalent to the 9-intersection model [10], though they are not equivalent in terms of complexity of reasoning, as proved by Grigni *et al.* [34].

RCC-8 is a constraint algebra formed by the disjunctions of the eight relations DC, EC, PO, TPP, TPP$^{-1}$, NTPP, NTPP$^{-1}$ and EQ. These eight relations form a partition of the universal relation over spatial regions. The names of the relations are disconnected (DC), externally connected (EC), properly overlaps (PO), tangential proper part (TPP), having as a tangential proper part (TPP$^{-1}$), non-tangential proper part (NTPP), having as a non-tangential proper part (NTPP$^{-1}$), and equals (EQ). In Figure 1, a 2D pictorial representation of the RCC-8 relations is provided.

The problem of deciding the consistency of a network of constraints in RCC-8 has been proved to be NP-complete by Nebel [35]. Nebel also proved that this problem is polynomially solvable for constraint sets where we only allow RCC-8 basic relations. Renz and Nebel [36] improved the results above, by showing that there exists a maximal tractable subclass of RCC-5, indicated by $\hat{\mathscr{H}}_5$, formed by 28 relations out of 32, which includes all the basic relations, and a maximal tractable subclass of RCC-8, indicated by $\hat{\mathscr{H}}_8$, formed by 148 relations out of 256 including the basic relations. A maximal tractable subclass $A$, is a subset of a constraint algebra, such that, problems defined on $A$ are tractable, while problems defined on proper supersets of $A$ are not. Subsequently, Renz [37] showed that there are in fact three maximal tractable subsets of RCC-8 (which include all eight base relations).

The morphological reasoning framework MC-4, originally introduced by Cristani [4], is a constraint algebra with four basic relations: CG, CGPP, CGPP$^{-1}$ and CNO. A pictorial illustration of the MC-4 basic relations is shown in Figure 2.

In the same paper [4], Cristani proved that the problem of deciding the consistency of finite sets of constraints on MC-4 is NP-complete. As in RCC-8, deciding the consistency of basic relation constraint sets is polynomially solvable and our reasoning framework in SpaceML only allows basic relations for MC-4 as well.[c]

Although there have been proposals for purely qualitative representations of distance, e.g. [5, 6], here we propose to use a simple interval metric representation since this is likely to be of more use in the envisaged contexts. In this context, work on temporal representations is also relevant since time, like distance, is one dimensional. A simple temporal constraint network (STN) [32], is a graph, labelled on vertices with names of variables, and each edge by one real (or rational) closed interval. The interpretation of the label of one edge is that the distance between the variables labelling the vertices is included between the minimal and maximal values of the interval itself. The

[c] Note that there has been rather little work done on combining systems of relations covering different aspects of spatial relations and then analysing their resulting complexity; one exception is the work on combining topology and distance [38].

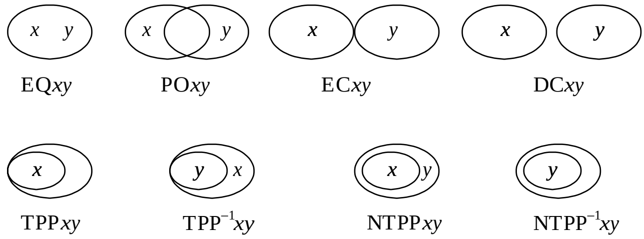


**Figure 1.** The basic relations of RCC-8

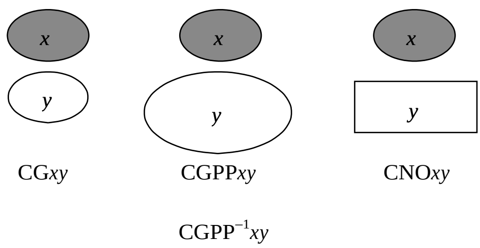


**Figure 2.** The MC-4 basic relations pictorially illustrated

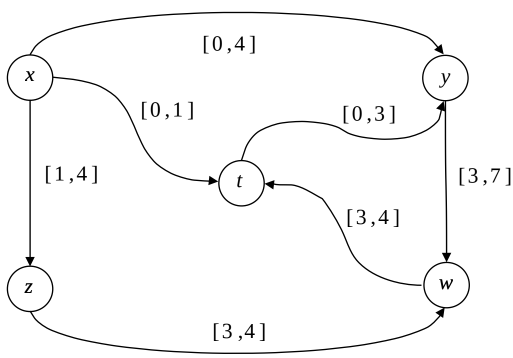


**Figure 3.** An example of an STN

interpretation of unlabelled edges is that the distance is not constrained, or that the variability interval is $(-\infty, +\infty)$. Simple Temporal Constraint Networks can be checked to be consistent in $O(n^3)$ [32]. In Figure 3, we show an example of an STN.

Note that, since distance is a metric on $\mathbb{R}^2$, the composition of two intervals as given in [32] only constrains the variability of distance, because of the triangular inequality. In the original framework, the composition of two intervals is generated by the definition of composition for distances in one-dimensional space; i.e. given three points on the real axis $x_1, x_2$ and $x_3$, the distance $d(x_1, x_3) = d(x_1, x_2) + d(x_2, x_3)$, whilst, in general, in the real plane, $d(x_1, x_3) \leq d(x_1, x_2) + d(x_2, x_3)$.

The cardinal relation framework we use here is one of the models which have been developed in [5], and investigated in depth in [8]. In the model we assume, there are eight possible relations: N, NE, E, SE, S, SW, W, NW. For instance, N*ab* means *a* is to the north of *b*. In Figure 4, we show the basic relations.

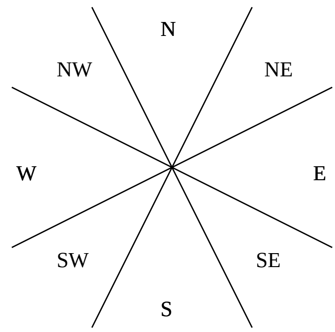


**Figure 4.** The eight basic relations of the cardinal direction framework

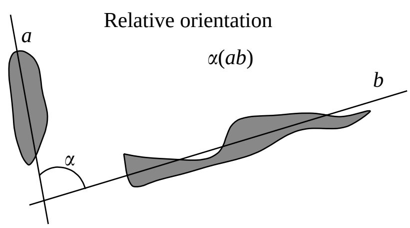


**Figure 5.** Interpreting angles in the orientation framework: for regions which have a principal axis, their relative orientation can be expressed using the orientation calculus of Isli and Cohn [17]

An alternative orientation framework is [7] which expresses the qualitative relation (same, opposite, left, right) between two orientations. This calculus can also be used to relate the principal axes of two regions, see Figure 5. In fact, in the formal language definition below, we regard this aspect of the language as taking values from the full quantitative range, rather than the four qualitative values of Isli and Cohn [7] as this allows greater precision which may be useful in geographic contexts.

The following description of the extended Backus–Naur form (EBNF) is taken from the W3C Recommendation REC-xml-19980210 [27]. The formal grammar of SpaceML is given here in this specification language. Only one modification to the grammar has been made: the introduction of a grouping notation { } distinct from the ( ) notation. Each rule in the grammar defines one symbol, in the form:

symbol ::= expression

Symbols are written with an initial capital letter if they are defined by a regular expression, or with an initial lower case letter otherwise. Literal strings are quoted.

Within the expression on the right-hand side of a rule, "string" and 'string' match a literal string matching that given inside the quotes.

These symbols may be combined to match more complex patterns as follows, where A and B represent simple expressions:

- (expression) expression is treated as a unit and may be combined as described in this list;

- A? matches A or nothing; optional A.
- A B matches A followed by B. {A B} also matches A followed by B.
- ($A_1$ $A_2$, … , $A_n$) matches any permutation of the $n$ elements within the parentheses.
- A|B matches A or B but not both.
- A − B matches any string that matches A but does not match B.
- A+ matches one or more occurrences of A.
- A* matches zero or more occurrences of A.

Other notations used in the productions are:

- /* … */: comment.
- [WFC: … ]: well-formedness constraint; this identifies by name a constraint on well-formed documents associated with a production.
- [VC: … ]: validity constraint; this identifies by name a constraint on valid documents associated with a production.

The essential difference between a WFC and a VC is that the former is essentially a syntactic condition which can be checked by pure symbol manipulation without reference to the meaning of any symbols, whilst the latter requires semantic knowledge of what the symbols denote. We give the following reference definitions.

**Definition 3.** A data object $T$ is a well-formed SpaceML document iff it respects the specification of tags of Section 4 and all the well-formedness constraints appearing in those definitions.

**Definition 4.** A SpaceML document $T$ is valid iff:

- $T$ has an associated document-type declaration *x*:*dtd* including the tag definitions of Section 4 or an XML Schema declaring the same tags, and
- the document T complies with the well-formedness constraints expressed in the tag definitions of Section 4, and
- $T$ satisfies all the validity constraints expressed in the tag definitions of Section 4.

## 4. The Language Definition of SpaceML

In this section, we introduce the SpaceML language. We can define the syntax of SpaceML by a DTD of XML[d]. The file spaceml.dtd, listed in Appendix A, contains the syntactic definitions of the tags of SpaceML.

Note that this DTD of XML simply defines the syntax of SpaceML. In order to define the semantics of SpaceML, we need to provide the Processing Instructions of the language. Thus we provide the syntactic definition of the language, and in order to be able to use it in practice for marking up spatial knowledge, we shall also provide an interpretation schema. All these goals can be achieved by means of an XML Schema specification of the language, and for certain given purposes, in conjunction with RDFS documents, namely RDF Schema. In this paper, we do not want to be particularly precise as to the kind of

[d] Clearly, the same specification can be produced by an XML Schema. However, the DTD specification is much easier to read for this kind of language.

technology to be used for implementing SpaceML for two main reasons:

- the method for specifying languages has changed rather often in the recent past, and it is very likely to change again. Our specification in terms of tag definitions is hopefully more stable;
- the development of a specific DTD or an XML Schema is not a defined merit in such a research field, so we do not focus on it but on the language specifications.

We now introduce the syntax of SpaceML.

The general syntax of a SpaceML document is the syntax of an XML document provided that:

- the DTD file (or the DTD instructions) or the XML Schema provides the syntax of SpaceML as defined below;
- The document contains exactly one Spatial-block.

The DTD we provide in Appendix A only specifies the syntax of SpaceML in one model where the SpaceML element is the root (document element). It is compatible with the specification of this paper to provide the SpaceML element as a non-root element. We then define the syntax of a Spatial-block.

In the definitions below, we assume the following:

- the expression Cdata represents the implicit definition of a piece of an XML page not containing a tag;
- the expression S represents any admitted kind of spacing: space, tab, line break, page break, and so on;
- the expression Name refers to any valid name in the syntax of XML, therefore, in general, a string not containing control characters, not empty, starting with an alphabetic character;
- the expression URL defines the correct syntax of an Internet address.

We will illustrate SpaceML through a running example which concerns an imaginary web page concerning the European Union. In this example, the definition framework looks like this:

Main Example

```
  ⟨?xml version = "1.0"?⟩
  ⟨!DOCTYPE Spaceml (View Source for full doctype ... )⟩
+ ⟨Spaceml Name = "The European Union" Description = "This page contains the general description
  of the European Union viewed as a Geographic object."⟩
```

The Spatial block of a SpaceML document is started by ⟨Spaceml ... ⟩ and it is ended by ⟨/Spaceml⟩. The general syntax of a Spatial block is thus

```
[1] Spatial-block :: =
    '⟨Spaceml' S 'Name = " ' Cdata ' " '
    S ('Description = " 'Cdata ' " ')? '⟩'
    (Const-block Var-block Relations-block)
    '⟨/Spaceml⟩'
```

[WFC: The relations in any Relations-block hold among the objects defined in the Const or Var block.]

where the Const-block, the Var-block and the Relations-block child blocks are defined further. The well-formedness constraint establishes that the objects referred to in the Relations-block have to be defined in Const-block, and Var-block.

Let us go back to our guiding example.

**Main Example** (*continued)*

```
  <?xml version = "1.0"?>
  <!DOCTYPE Spaceml (View Source for full doctype ...)>
—<Spaceml Name = "The European Union" Description = "This page contains the general description
of the European Union viewed as a Geographic object.">
+ <Const>
+ <Var>
+ <Relations>
  </Spaceml>
```

Pursuing a top-down presentation strategy, we now make explicit the structure of the sub-blocks introduced above. The following definition of Const-block is basic to understanding the general philosophy of the SpaceML language.

```
[2] Const-block ::=
    '<Const>' (Object-def Region-def Category-def )*
    '</Const>'
```

Our elementary ontology of space is thus defined by three primitive sets:

- The Spatial Objects, namely those entities whose nature is spatial, but which are not the physical structure of a spatial region; i.e. entities which are not purely spatial (such as regions) but which are spatially located (such as countries), and have other, non-spatial properties.
- The Spatial Regions;
- The Spatial Categories, namely any reasonable grouping of Spatial Objects whose ontological commitment depends on a particular set of properties. For instance, the category 'Static', which we use in the guiding example, refers to spatial objects which last for some time, like a State, as compared to, say, an epidemic or a migration, which are intrinsically 'Dynamic' objects.

A synthetic schema of our basic ontological assumptions is shown in Figure 6.

The schema introduced above are illustrated in the SpaceML example below.

**Main Example** (*continued*)

```
— <Const>
<Object Name = "Europe" Cat = "Static">
        The Europe reference is the context in which the member States of the European Union are
        defined.
</Object>
<Object Name = "Germany" Cat = "Static"
        Region = "Germany_Territory">
        Germany is the widest country within the
        European Union.
  </Object>
```

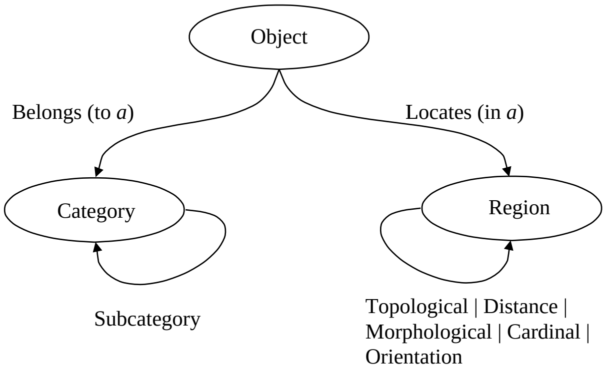


**Figure 6.** The ontological choices underlying the introduction of the three groups above

+ ⟨Region Name = "European_Territory"⟩
+ ⟨Physics⟩
⟨/Region⟩
+ ⟨Region Name = "Germany_Territory"⟩
+ ⟨Physics⟩
⟨/Region⟩
⟨Category Name = "General"/⟩
⟨Category Name = "Static" Sub = "General"/⟩
⟨/Const⟩

We now present the definitions of Object-def, Region-def and Category-def.

[3] Object-def ::=
'⟨Object' S 'Name = " 'Cdata '" ' S ('Cat = " '
Cdata ' " ' S)? ('Region = " ' Cdata ' " ' S)? '⟩' Cdata
'⟨/Object⟩'

[WFC: The Category assigned to a spatial object has to exist among the defined categories of the document.]
[WFC: The Region assigned to a spatial object has to exist among the defined regions of the document.]

The definition of Object-def recurs on both regions and categories. A region can be defined simply, just by naming it, or by linking the name of the region to a physical definition. A physical definition is a sequence of two-dimensional (just for practical reasons) points connected by straight segments.

[4] Region-def ::=
'⟨Region' S 'Name = " ' Cdata ' " ' S ' ⟩' (Physical-def)? '⟨/Region⟩'

[5] Physical-def ::=
'⟨Physics⟩' (Point)* '⟨/Physics⟩'

[WFC: The points of one set of Physics are all distinct.]

[VC: The polygonal line defined by the junctions of points occurring in a Physics tag, closing the last point to the first, is not interlaced.

[6] Point :: =
'⟨Point' 'X = " ' S Cdata '"' 'Y = " ' Cdata '" / ⟩'

The definition of categories has a very simple schema. A category is used to distinguish groups of spatial objects which differ conceptually in such a way that it is worth representing the notion of that category explicitly. For instance, STATIC and DYNAMIC are interesting categories, since dynamic objects are distinguishable with respect to static ones as mentioned above. Here, we do not make assumptions about the representation schema for categories, such as Description Logic schemata or Semantic Networks [39].

[7] Category-def:: =
'⟨Category' 'Name = " ' S Cdata ('Sub = ?' Cdata '"' S)? '/⟩'

[WFC: The category to which Sub refers has to exist among the defined categories.]

[VC: The set of all defined categories forms a DAG, namely the relation implicitly defined by the Sub attribute in category tags is a partial order.]

Variables are three-sorted: object variables, region variables, category variables. Our guiding example specifies some variables, as follows.

**Main Example** (*continued*)

— ⟨Var⟩
⟨Var_Category Name = "c1" /⟩
⟨Var_Category Name = "c2" /⟩
⟨Var_Object Name = "o1" Value = "Germany" /⟩
⟨Var_Region Name = "r1" Value = "Andorran_Territory"/⟩
⟨Var_Region Name = "r2" Value = "French_Territory"/⟩
⟨Var_Region Name = "r3" /⟩
⟨/Var⟩

We now provide SpaceML definitions of the new syntactic categories just introduced.

[8] Var-block :: =
'⟨Var⟩' (VarObject-def VarRegion-def VarCategory-def)* '⟨/Var⟩'

[9] VarObject-def :: =
'⟨Var-Object' S 'Name = " ' Cdata '" ' S ('Value = "'Cdata '"' S)? '/⟩'

[WFC: The Value assigned to a Var_Object has to exist among the defined objects.]

[10] VarRegion-def :: =
'⟨Var_Region' S 'Name = " ' Cdata '"' S ('Value = "' Cdata '"')? '/⟩'

[WFC: The Value assigned to a Var_Region has to exist among the defined regions.]

```
[11] VarCategory-def ::=
     '⟨Var_Category' S 'Name = "' Cdata '"' S ('Value = "' Cdata '"')? '/'
```

[WFC: The Value assigned to a Var_Category has to exist among the defined categories.]

The relations block is the most important in terms of structure definitions. Note that the implicit semantic constraints on the behaviour of what the W3C Recommendation generically call 'The Application' (see [27]) are not specified here (except for validity constraints). In fact, there exist a variety of admissible behaviours. For instance, the application can manage inconsistency as invalidation of the document, or can try to behave incrementally, or use some sort of refined logical model of the contradictory knowledge, like Epistemic Logic, or Modal Systems, or even Linear Logic. Here, we simply leave the semantics unspecified, and only deal with the pure syntactic structure.

Our guiding example finishes therefore with the definition of some spatial relations as follows.

**Main Example** (*last passage*)

```
— ⟨Relations⟩
  ⟨Topological A = "r1"
               B = "Germany_Territory" Attr = "DC" /⟩
  ⟨Topological A = "r1"
               B = "Europe_Territory" Attr = "NTPP" /⟩
  ⟨Topological A = "r2"
               B = "Germany_Territory" Attr = "EC"/⟩
  ⟨Distance A = "r3"
            B = "Germany_Territory" Attr = "100" /⟩
  ⟨Morphological A = "r1"
                 B = "Germany_Territory" Attr = "CGPP" /⟩
  ⟨Morphological A = "r2"
                 B = "Germany_Territory" Attr = "CNO" /⟩
  ⟨/Relations⟩
```

```
[12] Relations-block ::=
     '⟨Relations⟩' (Topological-relation|Distance-
     relation| Cardinal-relation|Orientation-relation
     |Morphological-relation '⟨/Relations⟩'
[13] Topological-relation ::=
     '⟨Topological' S 'A = " ' Cdata '"' S'B = "' Cdata '" ' S
     'Attr = "'TopoAttribute'"/⟩'
```

[WFC: The values assigned to A and B have to be existing VarRegions or defined constant Regions or defined constant Objects with the Region attribute assigned to a defined spatial region.]

We also introduce a parameter entity which specifies the possible values.

```
[14] TopoAttribute ::=
       'EQ' | 'TPP' | 'NTPP' | 'PO' | 'EC' | 'DC'
```

Analogously, we define the morphological relations.

[15] Morphological-relation::=
'⟨Morphological' S'A = " ' Cdata ' " 'S ' B = "' Cdata '"'
S 'Attr = "'MorphoAttribute '" /⟩'

[WFC: The values assigned to A and B have to be existing
VarRegions or defined constant Regions or defined
constant Objects with the Region attribute
assigned to a defined spatial region.]

For these relations, a parameter entity has also been introduced.

[16] MorphoAttribute ::=
'CG' | 'CGPP' | 'CNO'

The cardinal relations are specified as follows:

[17] Cardinal-relation ::=
'⟨Cardinal' S'A = "' Cdata '"' S' B = " ' Cdata '" ' S
'Attr = " ' CardAttribute '" /⟩'

[WFC: The values assigned to A and B have to be existing
VarRegions or defined constant Regions or defined
constant Objects with the Region attribute
assigned to a defined spatial region.]

Again we have a parameter specifying the possible values of the relation.

[18] CardAttribute ::=
'N' | 'S' | 'W' | 'E' | 'NE' | 'NW' | 'SE' | 'SW'

Finally, we have two infinite relation algebras, for distance and orientation.

[19] Distance–relation ::=
'⟨Distance' S'A = "' Cdata '"'S 'B = " ' Cdata '" ' S
'Min = " ' Cdata'" 'S'Max = " ' Cdata'" /⟩'

[WFC: The values assigned to A and B have to be existing
VarRegions or defined constant Regions or defined
constant Objects with the Region attribute
assigned to a defined spatial region.]

[20] Orientation–relation ::=
'⟨Orientation' S'A = "' Cdata '"' S 'B = " ' Cdata '" ' S
'Attr = " 'Cdata'" /⟩'

[WFC: The values assigned to A and B have to be existing
VarRegions or defined constant Regions or defined
constant Objects with the Region attribute
assigned to a defined spatial region.]

## 5. Conclusions and Further Work

We have proposed a new mark-up language, the SpaceML language, whose purpose is marking up the Spatial knowledge contained in a web page. The language syntax has been presented, and we have discussed some constraints to the semantics, to be intended as recommendations, suggestions and standardization issues for developers who undertake the design of a SpaceML-oriented browser.

There are various directions in which the research can be taken further. In particular, we are designing a formal language in which the well-formedness and the validity constraints conditions can be formally stated (and thus checked). A browser which can read SpaceML documents and check well-formedness constraints is a syntactic checker, whilst one which can additionally check the validity constraints analysis modules is a validating bowser.

In SpaceML, one can specify both physical attributes and qualitative attributes (one or the other or both or neither). Obviously if they are both present, then there is the question of checking for consistency of the two kinds of spatial information; if just physical information is present then there is the question of inferring qualitative information from the physical (and in this case there is the issue of tolerance—when computing, e.g. topological relationships, depending on the accuracy of one's arithmetical calculations, then one might get two regions being DC, EC or PO, for example). We can, therefore, refine a validating browser.

Another issue is the answer to the question: How can we automate or partially automate the process of tagging pages with SpaceML?

Finally, we note that it would be very useful to be able to extend the language in the direction of being able to specify pages at different levels of spatial granularity.

## Acknowledgements

The authors are indebted to the attendees of the CollECTer Conference on Electronic Commerce U.S.A. 2000, who have suggested some interesting modifications to the structure of the language. We are also indebted to the anonymous reviewers of the Special Issue of the Journal of Visual Languages and Computing for their careful and useful comments.

The second author gratefully acknowledges the financial support of the EPSRC under grant GR/M56807.

## References

1. M. Cristani (2000) Geo-ML: a Mark-up language for geography-related WebPages. In: *CollECTer Conference on Electronic Commerce,* Breckenridge, CO, U.S.A.
2. D. A. Randell & A. G. Cohn (1989) Modelling topological and metrical properties of physical processes. In: *Proceedings of the 1st International Conference on Knowledge Representation and Reasoning, KR-89* (R. J. Brachman, H. J. Levesque & R. Reiter, eds), Toronto, Ont, Canada, May. Morgan Kaufmann, Los Altos, CA, pp. 55–66.
3. D. A. Randell, Z. Cui & A. G. Cohn (1992) A spatial logic based on regions and connection. In: *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning, KR-92* (B. Swartout & B. Nebel, eds), Cambridge, MA, U.S.A., October. Morgan Kaufmann, Los Altos, CA, pp. 165–176.

4. M. Cristani (1999) The complexity of reasoning about spatial congruence. *Journal of Artificial Intelligence Research* **11,** 361–390.
5. A. Frank (1992) Qualitative spatial reasoning about distances and directions. *Journal of Visual Languages and Computing* **3,** 343–371.
6. D. Hernández, E. Clementini & P. Di Felice (1995) Qualitative distances. In: *Spatial Information Theory: A Theoretical Basis for GIS* (A. U. Frank & W. Kuhn, eds), Lecture Notes in Computer Science, No. 988, Semmering, Austria, September COSIT'95. Springer-Verlag, Berlin, pp. 45–58.
7. A. Isli & A. G. Cohn (2000) A new approach to cyclic ordering of 2D orientations using ternary relation algebras. *Artificial Intelligence* **122,** 137–187.
8. G. Ligozat (1998) Reasoning about cardinal directions. *Journal of Visual Languages and Computing* **9,** 23–44.
9. P. Jonsson & T. Drakengren (1997) A complete classification of tractability in RCC-5. *Journal of Artificial Intelligence Research* **6,** 211–221 (Research Note).
10. M. J. Egenhofer (1991) Reasoning about binary topological relations. In: *Advances in Spatial Databases-2nd Symposium SSD '91,* Lecture Notes in Computer Science, No. 525 (O. Gunther & H. J. Schek, eds). Springer-Verlag, New York, NY, U.S.A., pp. 143–160.
11. A. G. Cohn, B. Bennett, J. Gooday & N. Gotts (1997). RCC: a calculus for region based qualitative spatial reasoning. *GeoInformatica* **1,** 275–316.
12. Z. Cui (1994) Using interval logic for order assembly. In: *Proceedings of the 2nd International Conference on Intelligent Systems in Molecular Biology.* AAAI Press, Menlo Park, CA, pp. 103–111.
13. A. G. Cohn (2001) Formalising bio-spatial knowledge. In: *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems* (FOIS'01) (C. Welty & B. Smith, eds), ACM, New York.
14. K. Zimmermann (1995) Measuring without measures: the $\delta$-calculus. In: *Spatial Information Theory: A Theoretical Basis for GIS, COSIT'95* (A. U. Frank & W. Kuhn, eds), Lecture Notes in Computer Science, No. 988, Semmering, Austria, September. Springer-Verlag, Berlin pp. 59–67.
15. M. Knauff, R. Rauh & J. Renz (1997) A cognitive assessment of topological spatial relations: results from an empirical investigation. In: *COSIT 1997 Conference Proceedings* (S. C. Hirtle & A. U. Frank, eds), Laurel Highlands, Pennsylvania, October, pp. 193–206.
16. E. Clementini, P. Di Felice & P. Oosterom (1994) A small set of formal topological relationships suitable for end user interaction. In: *Advances in Spatial Databases. Proceedings of the 3rd International Symposium on Spatial Databases*, SSD '93 (D. Abel & B. C. Ooi, eds), Singapore, Lecture Notes in Computer Science, No. 692. Springer-Verlag, Berlin, pp. 277–295.
17. M. J. Egenhofer & R. Franzosa (1991) Point-set topological spatial relations. *International Journal of Geographical Information Systems* **5,** 161–174.
18. M. Cristani (1997) Morphological spatial reasoning: preliminary report. Technical Report 08/97, LADSEB-CNR Padova, Italy.
19. A. G. Cohn & S. M. Hazarika (2001) Qualitative spatial representation and reasoning: an overview. *Fundamenta Informaticae* **45,** 1–29.
20. A. Galton (2000) *Qualitative Spatial Change.* Oxford University Press, Oxford.
21. S. Wu, A. Chen & Q. Li (2000) GeoML—Geospatial Object Markup Language on Internet. In: *Proceedings of the GeoInformatics 2000 Conference*, Monterey, CA, U.S.A.
22. A. Cuthbert, B. O'Rourke, E. Keighan, S. Margoulies, J. Sharma, P. Daisey, R. Lake & S. Johnson (2000) *Geography Markup Language (GML) v1.0.* Technical Report, OGC Document Number: 00–029, OpenGIS Consortium, http://www.opengis.org/techno/specs/00–029/GML.html.
23. B. Smith (1996) Mereotopology: a theory of parts and boundaries. *Data and Knowledge Engineering* **20,** 287–303. (Special Issue on Modelling Parts and Wholes).
24. R. Casati & A. Varzi (1995) Basic issues in spatial representation. In: *Proceedings of the 2nd World Conference on the Fundamentals of Artificial Intelligence*, Paris, Université Paris VI, France.
25. S. Borgo, N. Guarino & C. Masolo (1996) A pointless theory of space based on strong connection and congruence. In: *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning KR-96* (L. C. Aiello & J. Doyle, Eds), Morgan Kaufmann, San Francisco, CA, U.S.A.

26. S. Borgo, N. Guarino & C. Masolo (1997) An ontological theory of physical objects. In: *Proceedings of Eleventh International Workshop on Qualitative Reasoning* (*QR 1997*), (L. Ironi, ed.), Cortona, Italy, pp. 223–231.
27. T. Bray, J. Paoli, C. M. Sperberg-McQueen & E. Maler (2000) *Extensible Markup Language* (*XML*) *1.0,* (2nd edn). W3C Recommendation. Technical Report, W3C Consortium, http://www.w3.org/TR/REC-xml, October.
28. D. C. Fallside (2001) *XML Schema Part 0: Primer W3C Recommendation.* Technical report, W3C Consortium, http://www.w3.org/TR/xmlschema-0/, May.
29. H. S. Thompson, D. Beech, M. Maloney & N. Mendelsohn (2001) *XML Schema Part 1: Structures* W3C Recommendation. Technical Report, W3C Consortium, http://www.w3.org/TR/xmlschema-1/, May.
30. P. V. Biron & A. Malhotra (2001) XML Schema Part 2: Datatypes W3C Recommendation. Technical Report, W3C Consortium, http://www.w3.org/TR/xmlschema-2/, May.
31. T. Berners-lee, J. Hendler & O. Lassila (2001) The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American* **501,** 137–187.
32. R. Dechter, I. Meiri & J. Pearl (1991) Temporal constraints networks. *Artificial Intelligence* **49,** 61–95.
33. E. Tsang (1993) *Foundations of Constraint Satisfaction.* Academic Press, London, U.K.
34. M. Grigni, D. Papadias & C. Papadimitriou (1995) Topological inference. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence,* IJCAI-95 (C. Mellis, ed.) Montréal, PQ, Canada, August. Morgan Kaufmann, Los Altos, CA, pp. 901–906.
35. B. Nebel (1995) Computational properties of qualitative spatial reasoning: first results. In: *Advances in Artificial Intelligence, KI-95*, (I. Wachsmuth, C.-R. Rollinger & W. Brauer, eds), Bielefeld, Germany. Springer-Verlag, New York, NY, U.S.A., pp. 233–244.
36. J. Renz & B. Nebel (1997) On the complexity of qualitative spatial reasoning: a maximal tractable subclass of the regional connection calculus. In: *Proceedings of the 16th International Conference on Artificial Intelligence*, IJCAI 97, Nagoya, Aichi, Japan, August, 1997 (M. Pollack, ed.). Morgan Kaufmann, Los Altos, CA.
37. J. Renz (1999) Maximal tractable fragments of the region connection calculus: a complete analysis. In: *Proceedings of the 16th International Joint Conference on Artificial Intelligence, IJCAI'99* (T. Dean, ed.), Stockholm, Sweden, July. Morgan Kaufmann, Los Altos, CA.
38. A. Gerevini & J. Renz (1998) Combining topological and qualitative size constraints for spatial reasoning. In: *Principles and Practice of Constraint Programming–CP98,* 4th International Conference, Pisa, Italy, October, 1998, Proceedings (M. J. Maher & J.-F. Puget, eds). Lecture Notes in Computer Science 1520, Springer, pp. 220–234.
39. F. W. Lehmann (ed.) (1992) *Semantic Networks in Artificial Intelligence.* Pergamon Press, Oxford.

## Appendix A: A DTD Conformant to SpaceML Specifications

<!--
Authors: M. Cristani and A. G. Cohn
Version: 1.0
Organization: University of Verona (Italy) and University of Leeds (UK)
Date: 15.04.2001
This DTD expresses the syntax of the language SPACEML, a mark-up language specialized for spatial knowledge.

This DTD is freely distributable and can be modified, at your own risk in any way you think it may be useful. Please do not use the name SPACEML for DTDs obtained by modifying this document or even starting fresh ones if they are not conformant to the syntax expressed here. We will not be responsible for the maintenance of the language in any case. We will not be responsible for any modifications we will make to the language, nor will we take responsibility for approving modifications others make. - - ⟩

⟨ !ENTITY % TopoAttr "(EQ | TPP | NTPP | PO | EC | DC)"⟩

⟨ !ENTITY % CardAttr "(N | S | W | E | NE | NW | SE | SW)" ⟩
⟨ !ENTITY % MorphoAttr "(CG | CGPP | CNO)" ⟩

⟨!- -
ELEMENT: Spaceml
COMMENT: The root tag includes the major three parts of the language. The first (Const) includes the description of physical entities corresponding to spatial regions, spatial objects, spatial categories; the second one declares variables and possibly instantiates them; the third establishes relations among the variables.
- - ⟩

```
⟨ !ELEMENT Spaceml (Const,Var,Relations)⟩
⟨ !ATTLIST Spaceml
          Name            ID          #REQUIRED
          Description     CDATA       #IMPLIED⟩
```

⟨!- -
ELEMENT: Const
COMMENT: The constants are of three different kinds: Objects, Regions, Categories.
- - ⟩

```
⟨ !ELEMENT Const (Object | Region | Category)* ⟩

⟨ !ELEMENT Object ( #PCDATA)⟩
⟨ ! ATTLIST Object
          Name        ID          #REQUIRED
          Cat         IDREF       #IMPLIED
          Region      IDREF       #IMPLIED⟩

⟨!ELEMENT Region (Physics?)⟩
⟨!ATTLIST Region
          Name        ID          #REQUIRED⟩

⟨ !ELEMENT Physics (Point)* ⟩

⟨ !ELEMENT Point EMPTY ⟩
⟨ !ATTLIST Point
          X           NMTOKEN         #REQUIRED
          Y           NMTOKEN         #REQUIRED⟩

⟨ !ELEMENT Category EMPTY⟩
⟨!ATTLIST Category
          Name        ID          #REQUIRED
          Sub         IDREF       #IMPLIED⟩
```

⟨!- -
ELEMENT: Var
COMMENT: The variables are for Objects, Regions,
Categories.
- - ⟩

```
<!ELEMENT Var (Var_Object | Var_Region | Var_Category)* >

<!ELEMENT Var_Object EMPTY>
<!ATTLIST Var_Object
          Name          ID            #REQUIRED
          Value         IDREF         #IMPLIED>

<!ELEMENT Var_Region EMPTY>
<!ATTLIST Var_Region
          Name          ID            #REQUIRED
          Value         IDREF         #IMPLIED>

<!ELEMENT Var_Category EMPTY> <!ATTLIST Var_Category
           Name          ID            #REQUIRED
           Value         IDREF         #IMPLIED>


<!--
ELEMENT: Relations
COMMENT: The Relations tag includes the pairs of any relations defined within the
SpaceML page
-->

<!ELEMENT Relations
           (Topological | Distance | Cardinal |
            Orientation | Morphological)* >

<!ELEMENT Topological EMPTY>
<!ATTLIST Topological
          A          IDREF          #REQUIRED
          B          IDREF          #REQUIRED
          Attr       %TopoAttr;     #REQUIRED>

<!ELEMENT Distance EMPTY>
<!ATTLIST Distance
          A          IDREF          #REQUIRED
          B          IDREF          #REQUIRED
          Min        NMTOKEN        #REQUIRED
          Max        NMTOKEN        #REQUIRED>

<!ELEMENT Cardinal EMPTY>
<!ATTLIST Cardinal
          A          IDREF          #REQUIRED
          B          IDREF          #REQUIRED
          Attr       %CardAttr;     #REQUIRED>

<!ELEMENT Orientation EMPTY>
<!ATTLIST Orientation
          A          IDREF          #REQUIRED
          B          IDREF          #REQUIRED
          Attr       NMTOKEN        #REQUIRED>

<!ELEMENT Morphological EMPTY>
<!ATTLIST Morphological
          A          IDREF          #REQUIRED
          B          IDREF          #REQUIRED
          Attr       %MorphoAttr;   #REQUIRED>
```

## Appendix B: A SpaceML Web Page written in XML

```
<?xml version = "1.0"?>
<!DOCTYPE Spaceml SYSTEM "spaceml.dtd">
<Spaceml Name = "The European Union"
            Description = "This page contains the general description
            of the European Union viewed as a Geographic object.">
  <Const>
    <Object Name = "Europe" Cat = "Static">
    The General Europe reference is the context in which the member
    States of the European Union are defined.
    </Object>
    <Object Name = "Germany" Cat = "Static" Region = "Germany_Territory">
    Germany is the widest country within the European Union.
    </Object>
    <Region Name = "European_Territory">
      <Physics>
        <Point X = "121" Y = "100"/>        <Point X = "250" Y = "100"/>
        <Point X = "250" Y = "300"/>        <Point X = "121" Y = "300"/>
      </Physics>
    </Region>
    <Region Name = "Germany_Territory">
      <Physics>
        <Point X = "120" Y = "100"/>   <Point X = "250" Y = "85"/>
        <Point X = "200" Y = "300"/>   <Point X = "100" Y = "120">
        <Point X = "120" Y = "100"/>
      </Physics>
    </Region>
    <Category Name = "General">
    <Category Name = "Static" Sub = "General"/>
    </Const>
<Var>
    <Var_Category Name = "c1"/>    <Var_Category Name = "c2"/>
    <Var_Region Name = "r1"/>      <Var_Region Name = "r2"/>
    <Var_Region Name = "r3"/>
    <Var_Object Name = "o1" Value = "Germany"/>
</Var>
<Relations>
    <Topological A = "r1" B = "Germany_Territory" Attr = "DC"/>
    <Topological A = "r1" B = "Europe_Territory" Attr = "NTPP"/>
    <Topological A = "r2" B = "Germany_Territory" Attr = "EC"/>
    <Distance A = "r3" B = "Germany_Territory" Attr = "100"/>
    <Morphological A = "r1" B = "Germany_Territory" Attr = "CGPP"/>
    <Morphological A = "r2" B = "Germany_Territory" Attr = "CNO"/>
</Relations>
</Spaceml>
```