Huerta, Schade, Granell (Eds): Connecting a Digital Europe through Location and Place. Proceedings of the AGILE'2014 International Conference on Geographic Information Science, Castellón, June, 3-6, 2014. ISBN: 978-90-816960-4-3

A Geometric Configuration Ontology to Support Spatial Querying

Kristin Stock
Nottingham Geospatial Institute
University of Nottingham
Nottingham, United Kingdom NG7 2RD
kristin.stock@nottingham.ac.uk

Abstract

A number of ontologies of spatial relations have been defined in the literature, but most of these are either confined to a small subset of relations, or focussed on language expressions, and not specified geometrically. This paper presents an ontology of geometric configurations, to reflect and specify the range of spatial relations that have been discussed by previous researchers and that are commonly expressed in natural language, and to provide a sufficiently specific definition of the relations to allow them to be executed as spatial queries. Although this work was motivated by a goal to translate natural language describing location into spatial queries, we anticipate wider applications of the ontology for other purposes.

We define a three level ontology, informed by the literature and the study of a corpus of expressions of natural language geospatial location descriptions, and present the concepts and the definition using spatial queries.

Keywords: ontologies, spatial querying, natural language, ontologies.

1 Introduction

There is a distinction between spatial relations as they are used in language; spatial relations as they are described in the qualitative spatial reasoning (QSR) literature, which in some ways attempts to emulate the way spatial relations are used in language, and spatial relation queries that Geographic Information Systems (GIS) or standards-based Spatial Data Infrastructures (SDI) are capable of executing. In GIS and SDI, a restricted range of spatial operators is available, and the only qualitative spatial relations that are currently commonly supported are the basic topological spatial relations and simple buffer/distance calculations [31]. While topology is acknowledged as an important way of describing relations between objects in space, there are a number of other types of spatial relations that are commonly used in natural language, the meaning of which have been explored in detail in both linguistics and QSR. In order to allow geospatial systems to take advantage of the significant work in both spatial linguistics and QSR, it is necessary to develop a mechanism for translating non-topological spatial relations into actual spatial queries that can be executed in a metric system.

To this end, we present an ontology of geometric configurations (GCO). Notionally, it includes parameters to describe (1) spatial relations between pairs of two dimensional objects (for example, topology, orientation, proximity), and (2) the extensions of spatial objects (for example, shape and size). However, in this first version of the GCO, we do not address extension, but focus on spatial relations, and provide a placeholder for extension parameters to be added later. We consider that the combination of relations and extension is required to reflect many of the configurations between geographic objects that are described in natural language, which is the original motivation for this work.

The ontology describes the parameters diagrammatically and specifies them by providing the spatial query that can be used to execute the spatial relation in a GIS or SDI. In some cases, this is straightforward (for example, with topology), but in others, requires more manipulation to convert essentially qualitative parameters into quantitative queries. The ontology presented here brings together much of the QSR work, and specifies methods for converting it into quantitative queries. In many cases, we adopt existing approaches to do this (for example, methods for the quantification of the qualitative notion of proximity have been developed already), while in others, we define a new method.

The GCO provides approaches that could be used in a GIS or SDI, in which data sets may be modelled using points, lines, polygons and complex geometries. However, our work does not extend to 3 dimensional geometries. Finally, for this version, we confine our attention to binary relations.

2 Related Work

A number of linguistically motivated typologies and ontologies have been developed, with the goal of describing a range of spatial relations in terms of their linguistic representations. These are usually focussed around prepositions and explore spatial relations from a linguistic perspective, but do not provide spatially explicit, computational semantics for the terms included, many of which can encompass more than one spatial sense. For example, Coventry and Garrod's [10] typology of relational prepositions includes *in* and *on*, both of which have multiple possible spatial interpretations. Zwarts [35] provides another such typology, based on telicity, and the algebraic properties of different spatial relations. GUM-Space is a very detailed, linguistically motivated spatial ontology based on the General Upper Model (GUM), a task and domain independent linguistically motivated ontology [21]. GUM-Space includes a range of concepts that describe the pertinent content from a natural language spatial expression, but they do not specify a precise, concrete interpretation, and mappings are required [22].

At the other end of the scale from these linguistically motivated schemes, are schemes that focus on the mathematical interpretation of spatial relations. However, these are usual partial in addressing one or two parameters, with a particular focus on topology. For example, the Ordnance Survey Spatial Relations Ontology[1] includes topological operators, in addition to properties for describing metric location (easting and northing), while the NeoGeo spatial ontology[2] is restricted to topological relations.

Some other typologies and ontologies occupy positions in between these two extremes, including SUMO and OpenCyc[3], both of which provide partial specification, but full, executable semantics are not given [2, 14]. Ontologies that are combined with particular applications techniques include Bucher et al [5], who separate a geometric level ontology from an application level in their topology of spatial relations, and Bitters [4] who proposes to assign weighted probabilities to each relation in his ontology for a given pair of geographic features.

More general typologies are provided by Habel and Eschenback [19], who devise a three dimensional classification of spatial concepts, and Egenhofer and Franzosa [13], who divide spatial relations into topological, metric and ordered relations. In both these cases, full semantics are not provided. Upper level ontologies like DOLCE [16] and BFO's SNAP and SPAN [18] provide foundational concepts for the description of spatial concepts, but do not provide the level of detail required here. Finally, Kemmerer [24] highlights the cross-linguistic differences in spatial relations.

Figure 1: Geospatial Ontology Representation Layers

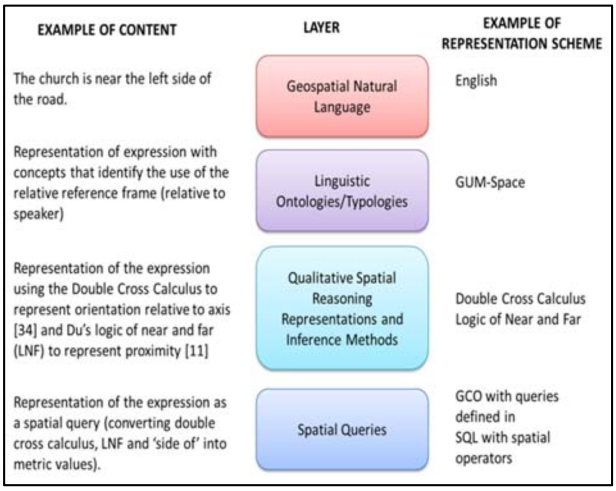


Figure 1 depicts the distinction between the different layers of knowledge representation that are exhibited by the previous work, and the role that the current work plays relative to it. Some of the ontologies described herein cross more than one layer, and to date, the bottom layer has not been separated from the layers above. In some cases the spatial query that is formulated is dependent on the representation used in the QSR layer, but the actual geometric configuration is independent of the QSR representation in our scheme. Thus alternative spatial queries may be developed for different QSR methods. Bennett's [3] work is closely related to the work described herein, and we draw on this work where possible.

3 The Ontology

3.1 Goals

The creation of the GCO was driven by three goals:

1. To create a simple ontology that could express the most common geospatial natural language expressions, rather than every possible permutation of geospatial relations.
2. To focus on the requirements of **geospatial** information. Many of the existing ontologies and typologies include levels of detail that are rarely relevant in the geospatial context. For example, OpenCyc includes spatial relations *hangs from* and *suspended in liquid*, which are not commonly used with geospatial objects.
3. To create concrete, rather than abstract, ontology concepts and properties, that could be specified using a geospatial query that can be executed in a GIS or SDI.

3.2 Three Level Structure

The ontology has three levels. The top level is a division between binary relational parameters (describing the geometric relation between pairs of geometries) and extensional parameters (describing the geometric extension of a single geometry). This accords roughly with the literature, which identifies the importance of relational parameters like topology, orientation and distance [7, 20, 26, 27], and extensional parameters like size and shape [6]. Although the top level of the ontology consists of two branches, the remainder of the work presented here covers the relational parameters only.

The second level of the ontology consists of a series of parameters, being characteristics that may be used to describe the relation or geometry. These include the most commonly discussed relational parameters in the literature (like topology, distance and orientation), as well as other parameters that are relevant for the colloquial description of spatial location. Many of the parameters that have been studied in detail in QSR are accommodated.

The third level of the ontology contains parameter values. In many cases, these are derived from existing literature in QSR and related areas, but they are also considered in terms of the range of ways in a which a parameter may be described, whether qualitative or quantitative. Although the parameter values are given simple language labels for convenience, it must be stressed that the ontology does not aim to describe linguistic spatial relations directly, as have many other ontologies and typologies (for example, SUMO and OpenCyc). We do not describe spatial relation words or phrases, but actual spatial relations that may be encountered in the world. Rather than approaching the problem from the language point of view, we approach from the geometric configuration point of view. The reason for this is that there are multiple ways of describing spatial relations in language

[1] http://data.ordnancesurvey.co.uk/ontology/spatialrelations/
[2] http://socop.oor.net/ontologies/1021
[3] http://www.cyc.com/platform/opencyc

(often the same spatial relation may be described in many different ways), and they differ depending on the language concerned and the context. In this way, multiple language constructs (individual words or more complex phrases) may be mapped to the same geometric configuration in the ontology. It is also likely that in some cases, language descriptions will map to multiple parameter values, which must all be true in order for the natural language expression to be fully realised. Finally, language based ontologies are often much more complex than this ontology, because they encompass the myriad different ways of describing a spatial relation. We aim to maintain the simple three level structure for this ontology, as much of the complexity is in the way language expresses relations, rather than the relations themselves.

The spatial relations are defined using spatial queries, which are provided using standard SQL syntax, according to ISO 13249-3 [23]. This standard was used because the goal of the work is to map geometric concepts to spatial queries, and ISO 13249-3 is widely used (sometimes with minor syntactic variations) by most GIS systems, along with relevant SDI standards. The queries included are designed to accommodate point, line and polygon geometries. We use previous work in the QSR literature to define these in some cases, as follows, but other schemes could easily be substituted to suit the required purposes.

Topology: We adopt a simple set of five parameter values, including the touches spatial relation (excluded from RCC5), but excluding the distinction between tangential and non-tangential proper parts [8,9]. This is because we have not found any evidence that the distinction is commonly made in natural language descriptions of spatial relations. We also exclude inverse relations (contains in addition to within), as the same relation can be expressed by reversing the geographic features concerned.

Distance: We adopt the simple logic of near and far (LNF) of Du et al [11], but other more complex schemes (and particularly, more advanced methods for calculating nearness) could be substituted [6, 15, 33]. LNF determines nearness using buffer zones and a fixed sigma value that is selected manually for the activity concerned, and the queries we define reflect this. We also include a quantitative representation of distance that is commonly encountered in natural language. It involves description of distance using a simple quantity and unit, the latter being either spatial (for example, 300 metres, 1 mile) or temporal, in which case it is includes an explicit or implicit mode of travel (for example, 3 minutes' walk, 5 hours' drive).

Linear Orientation: We adopt Dugat et al's [12] set of orientations, as it is the most comprehensive set, of several similar alternatives [1, 6, 26].

Horizontal Projective Orientation: We adopt the simple scheme of Clementini et al [6], defining left/right and back/front as semi-circles, rather than quadrants. This is because these simple relations more closely reflect the most commonly used linguistic expressions and because combined expressions like '*left and in front of*' may be determined by combining their individual components (left, in front of). We do not adopt the between relation proposed by more recent work by Clementini et al [7], which is similar in principle to relations provided in the Double Cross Calculus [34] and the Dipole Calculus [26], since these are also not commonly used in natural language (except for examples that include three objects, and are thus ternary relations and out of our scope).

Direction: We adopt Goyal and Egenhofer's [17] 9 cell model for cardinal direction relations, based on the minimum bounding rectangle of the reference object. We chose this basic model due to its simplicity and the absence of empirical evidence that the subsequent extensions more closely reflect natural language.

Adjacency: We refine the Wordnet definition of adjacency used by Klien and Lutz [25] to require that objects must be either disjoint or touching (as overlapping objects are unlikely to be next to each other). This is thought to be only partially expressive of the adjacency notion, which also includes a consideration of intervening objects of the same type, and is also in many cases part of more specific specialisations of adjacency like alongside (both approximately parallel and adjacent). However, these aspects are handled at the level of contextual analysis of a natural language expression, rather than the geometric configuration ontology.

Collocation: The notion of objects being 'in the same place' is commonly encountered in natural language, and described varying degrees of proximity. It is expressed often using the 'on' (*situated on the*) or 'at' (*located at the junction*) and 'in' (*in the area*) prepositions with varying degrees of precision. Previous work in this area is limited, so we adopt a simple scheme based on topological relations.

Object Parthood: A significant amount of work has addressed the notion of object parthood, and the language used to discuss it [29, 30, 32]. This previous work mainly focuses on identifying different types of parthood, depending on the functional relationship between parts and wholes, type similarity and separability. However, language descriptions of parthood commonly address specific parts of an object (start, end, middle), and our interest is in providing a mechanism for defining which part of a whole is referred to by particular specifications. For this purpose, we define a range of different specific parts. Our interest is in words that specify particular parts of wholes that apply across a range of feature types, rather than parthood relations that are specific to a particular feature type (for example, desert-oasis). In addition to the use of both orientation and direction parameter values as spatial relations (*x is north of y*), they may also be used as adjectives to define some part of an object. We treat these as spatial relations between some part of the object and its entirety.

Figure 2 shows the ontology parameters and parameter values, along with axioms and the queries that define them.

4 Examples

The following examples illustrate the mapping from natural language to queries via the GCO, and are based on results from a recently conducted questionnaire (to be reported in a future publication).

1. 'A train station in Nottingham' maps to the *contain* GCO concept, which maps to a simple query using the contains spatial relation, with the geometry for Nottingham, to show the area in which the desired train station might exist.

2. 'The street next to Jasmine Cottage' maps to the *adjacent* GCO concept, which uses the Jasmine Cottage geometry with the simple within distance method to create a query that selects streets in the appropriate area.
3. 'The monument outside the Town Hall' maps to both the *adjacent* and *disjoint* GCO concepts, and would require the two queries to be combined conjunctively so that only areas that are within a specified distance but not touching are included.
4. 'Development along the Trent River network' maps to the *parallel*, *alongside* and *side (part)* GCO concepts. As with the previous example, combined conjunctively.

These last two examples illustrate cases in which query composition is required. The default approach is to combine the queries conjunctively, so that all clauses must be fulfilled to define the area of interest. Future work will explore query composition methods in more detail.

Table 1: Counts for Each Relational Parameter

RELATIONAL PARAMETERS	**Qty**	**%**	**GCO**
path direction	128	**27.2%**	×
collocation	68	**14.4%**	✓
topology	44	**9.3%**	✓
direction	32	**6.8%**	✓
object parthood	32	**6.8%**	✓
distance	30	**6.4%**	✓
adjacency	19	**4.0%**	✓
horizontal projective orientation	17	**3.6%**	✓
traversal	16	**3.4%**	✓
throughness	13	**2.8%**	✓
alignment	11	**2.3%**	✓
joining	11	**2.3%**	Partial
vertical projective orientation	9	**1.9%**	Out of scope (3D)
distribution	9	**1.9%**	Partial
possession	7	**1.5%**	✓(topology)
betweenness	5	**1.1%**	Out of scope (tertiary)
surroundedness	5	**1.1%**	×
splitting	3	0.6%	Out of scope (tertiary)
aroundness	3	0.6%	×
sidedness	2	0.4%	Out of scope (tertiary)
boundedness	2	0.4%	Partial
protrusion	2	0.4%	×
linear orientation	1	0.2%	✓
oppositeness	1	0.2%	Out of scope (tertiary)
trajectory	1	0.2%	Out of scope (3D)
TOTALS	471	100%	

5 Evaluation

By way of partial evaluation of the focus and completeness of the ontology, we now present an analysis of the spatial parameters and parameter values used in a random selection of 200 spatial clauses from the Nottingham Corpus of Geospatial Language (NCGL) [28]. The NCGL is a publicly available[4] corpus containing only expressions that describe spatial location harvested from web sites, and is thus uniquely placed to evaluate the coverage of the GCO, unlike most other corpora that contain a wide selection of non-spatial language as well, making an evaluation of this kind very time-consuming.

In the 200 clauses from the NCGL that were examined, 471 distinct spatial relational parameter values were identified. Table 1 presents the quantities of each parameter value, listing all parameter values that were encountered, whether or not they appeared in the ontology. While 200 is not a very large quantity, the percentages were stable with the addition of the last 50 clauses in terms of the broad colour coded categories that indicate the percentage of spatial location expressions that the parameter in question includes (in the percent column, >10% shown in red; 3-9% inclusive shown in blue; 1-2% inclusive shown in brown).

While path location is the most frequent of the parameters, it was not included in this version of the GCO, because it is a grouping of a number of concepts that describe both geometric configuration and direction of movement, including expressions like *from, to, onto, away from, leave, upwards* and *uphill.* These deserve special treatment and are therefore beyond the scope of this version of the GCO.

The GCO covers all of the most frequently occurring relational parameters other than path direction, and 66% of the total set of relational parameters found in the 200 expressions. As can be seen from the right-most column in Table 1, future extensions to include tertiary relations and 3D would be useful to accommodate a more complete range of concepts, along with the addition of path direction and the development of the extensional parameters branch of the GCO.

6 Conclusions

This paper has presented a Geometric Configurations Ontology that is designed to support natural language querying, but has wider applications, and could be implemented as an interface to make a range of new spatial query operators available through standard GIS and SDIs. In future work we are conducted more extensive evaluations with empirical studies, and plan to extend the scope of the current ontology.

[4] http://geospatiallanguage.nottingham.ac.uk/

Acknowledgements

The work described in this paper was funded by Ordnance Survey under the NaturalGeo Project.

References

[1] M. Aurnague and L. Vieu. Towards a Formal Representation of Space in Language: A Commonsense Reasoning Approach. *IJCAI-93 Workshop on Spatial and Temporal Reasoning*, pages 123-158, 1993.

[2] J. Bateman and S. Farrar. Towards a generic foundation for spatial ontology. In Achille C. Varzi and Laure Vieu. *Formal Ontology in Information Systems: Proceedings of the Third International Conference on Formal Ontology in Information Systems (FOIS-2004)*. Amsterdam, pages 237-248, 2004.

[3] B. Bennett. Spatial Relations in the AURA Knowledge Base. Version 1.0. Unpublished Report, School of Computing, University of Leeds, 2012.

[4] B. Bitters. Spatial relationship networks: Network theory applied to GIS data. *Cartography and Geographic Information Science*, 36(1):81-93, 2009.

[5] B. Bucher, G. Falquet, E. Clementini, and M. Sester. Towards a typology of spatial relations and properties for urban applications, Usage, Usability, and Utility of 3D City Models – European COST Action TU0801, 2012.

[6] E. Clementini, P. Di Felice, and D. Hernández. Qualitative Representation of Positional Information, *Artificial Intelligence*, 95, 317-356, 1997.

[7] E. Clementini, S. Skiadopoulos, R. Billen and F. Tarquini. A Reasoning System of Ternary Projective Relations. *IEEE Trans. Knowl. Data Eng*. 22(2): 161-178, 2010.

[8] A. Cohn, B. Bennett, J. Gooday and N. Gotts. Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. *Geoinformatica*, 1, 275-316, 1997.

[9] A. Cohn and S. Hazarika. Qualitative Spatial Representation and Reasoning: An Overview. *Fundamenta Informaticae*, 43: 2-32, 2001.

[10] K.R. Coventry and S.C. Garrod. *Saying, Seeing and Acting: The Psychological Semantics of Spatial Prepositions*. Psychology Press, East Sussex, 2004.

[11] H. Du, N. Alechina, K. Stock, and M. Jackson. The logic of NEAR and FAR. *COSIT 2013: Conference on Spatial Information Theory*, Scarborough, UK, 2013.

[12] V. Dugat, P. Gambarotto and Y. Larvor. Qualitative Theory of Shape and Orientation, *Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI '99)*, pages 45-53, 1999.

[13] M.J. Egenhofer and R. Franzosa. Point-Set Topological Spatial Relations. *International Journal of Geographical Information Systems*, 5: 161-174, 1991.

[14] S. Farrar and J. Bateman. General Ontology Baseline. Collaborative Research Center for Spatial Cognition, University of Bremen, Germany. I1-[Main.OntoSpace]: D1. SFB/TR8 internal report http://www.ontospace.uni-bremen.de/pub/FarrarBateman04-i1-d1.pdf. 2005.

[15] A.U. Frank. Qualitative Spatial Reasoning about Distances and Directions in Geographic Space. *Journal of Visual Languages and Computing*, 3: 343-371, 1992.

[16] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari and L. Schneider. Sweetening Ontologies with DOLCE. In A. Gómez-Pérez, V.R. Benjamins, editors, *Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, 13th International Conference, EKAW 2002*, Siguenza, Spain, Springer Verlag, 166-181, 2002.

[17] R. Goyal and M. Egenhofer. The Direction-Relation Matrix: A Representation for Directions Relations between Extended Spatial Objects, *Proc. Univ. Consortium for Geographic Information Systems Science (UCGIS) Ann. Assembly and Summer Retreat*, June 1997.

[18] P. Grenon and B. Smith. SNAP and SPAN: Towards Dynamic Spatial Ontology, *Spatial Cognition and Computation*, 4(1): 69-103, 2004.

[19] C. Habel and C. Eschenbach. Abstract structures in spatial cognition. In C. Freksa, M. Jantzen and R. Valk, editors, *Foundations of Computer Science. Potential – Theory – Cognition*, Springer, Berlin, pages 369–378, 1997.

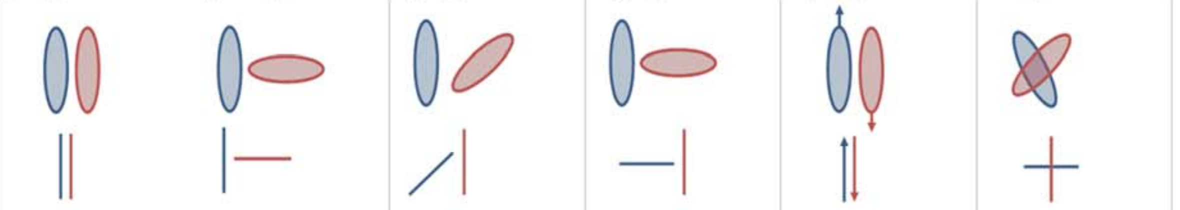
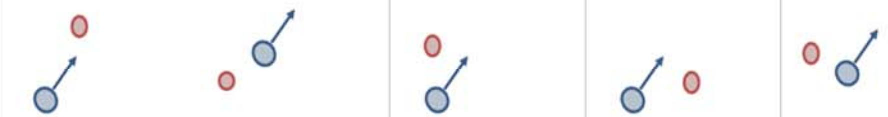
[20] T. Hahmann and M. Grüninger. A naïve theory of dimension for qualitative spatial relations. In *Proc. of the Symposium on Logical Formalizations of Commonsense Reasoning (CommonSense 2011)*, AAAI Spring Symposium, 2011. AAAI Press, 2011.

[21] J. Hois, T. Tenbrink, R. Ross and J. Bateman. GUM-Space: The Generalized Upper Model spatial extension: a linguistically-motivated ontology for the semantics of spatial language Technical Report. Universität Bremen SFB/TR8 Spatial Cognition, http://www.ontospace.uni-bremen.de/ontology/TechnReport09GUMspace.pdf, 2009.

[22] J. Hois and O. Kutz. Natural Language meets Spatial Calculi. *Spatial Cognition VI. Learning, Reasoning, and Talking about Space. 6th International Conference on Spatial Cognition*, pages 266-282, 2008.

[23] International Standards Organisation. Information technology – Database languages – SQL multimedia and application packages, Part 3: Spatial. 13249-3:2011 International Standard, 2011.

Figure 2: The Geometric Configuration Ontology

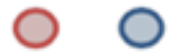
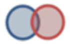
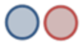
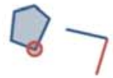
Parameter		Values							
TOPOLOGY (t): Are the objects connected and how?	Label	overlap(a,b)	touch(a,b)	contain(a,b)	disjoint(a,b)	equal(a,b)			
	Illustration								
	Query	ST_Overlaps(a,b) = 1	ST_Touches(a,b) =1	ST_Contains(a,b) = 1	ST_Disjoint(a,b) = 1	ST_Equals(a,b) = 1			
	Axioms								
DISTANCE (ds): How close are the objects to each other?	Label	distance 0 all points(a,b)	distance 0 any point(a,b)	very near(a,b)	near(a,b)	neither near nor far(a,b)	far(a,b)	x spatial units apart (a,b,x)	x temporal units apart by travel at y velocity[1](a,b,x,y)
	Illustration								
	Query (WHERE clause)	ST_Equals(a,b) = 1	ST_Touches(a,b) =1	ST_DWithin(a,b,σ)	ST_DWithin(a,b,2σ)	(ST_Distance(a,b) > 2σ) AND (ST_Distance(a,b) < 4σ)	NOT ST_DWithin(a,b,4σ)	ST_Distance(a,b) = x ST_Distance(ST_Centroid(a), ST_Centroid(b)) = x	ST_Distance(a,b) = xy ST_Distance(ST_Centroid(a), ST_Centroid(b)) = xy
	Axioms	ds.zeroAllPoints(a,b) ≡ t.equal(a,b)	ds.zeroAnyPoint(a,b) ≡ t.touch(a,b)	ds.veryNear(a,b) ≡ (t.disjoint(a,b) ∨ t.touch(a,b))	ds.near ≡ (t.disjoint(a,b) ∨ t.touch(a,b))	ds.neitherNearNorFar(a,b) ≡ t.disjoint(a,b)	ds.far(a,b) ≡ t.disjoint(a,b)	ds.spatialUnitsApart(a,b) ≡ t.disjoint(a,b)	ds.temporalUnitsApart(a,b) ≡ t.disjoint(a,b)
LINEAR ORIENTATION(lo): How are linear objects oriented relative to each other?	Label	parallel(a,b)	perpendicular(a,b)	diagonal(a,b)	orthogonal(a,b)	antiparallel(a,b)	crossed(a,b)		
	Illustration								
	Query (WHERE clause)	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) = 0 ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = 0	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) IN (π/2, 3π/2) ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - (ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = IN (π/2, 3π/2)	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) IN (π/4, 3π/4, 5π/4, 7π/4) ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - (ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = IN (π/4, 3π/4, 5π/4, 7π/4)	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) IN (0, π/2, π, 3π/2) ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - (ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = IN (0, π/2, π, 3π/2)	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) = π ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - (ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = π	MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(a)))) – MaxAzimuth[2](ST_Boundary(SSRectangle[3](ST_ConvexHull(b)))) IN (π/2, 3π/2) AND ST_Overlaps(a,b) ST_Azimuth(ST_StartPoint(a), ST_EndPoint(a)) - (ST_Azimuth(ST_StartPoint(b), ST_EndPoint(b)) = IN (π/2, 3π/2) AND ST_Overlaps(a,b)		
	Axioms	lo.parallel(a,b)⇒lo.orthogonal(a,b)	lo.perpendicular(a,b)⇒lo.orthogonal(a,b)			lo.antiparallel(a,b)⇒lo.orthogonal(a,b)	lo.crosses(a,b)⇒t.overlap(a,b)		
HORIZONTAL PROJECTIVE ORIENTATION(hpo): How are objects oriented to each other relative to a projected axis?	Label	in front of(a,θ[4],b)	behind(a,θ,b)	left(a,θ,b)	right(a,θ,b)	alongside(a,θ,b)			
	Illustration								
	Query	ST_Angle(ST_Azimuth(a,b), θ) < π/2	ST_Angle(ST_Azimuth(a,b), θ) > π/2	(ST_Azimuth(a,b) < θ) AND (ST_Azimuth(a,b) > θ±2 π)	(ST_Azimuth(a,b) > θ) AND (ST_Azimuth(a,b) < θ±2 π)	ST_Angle(ST_Azimuth(a,b), θ) IN (π, 3π/2)			
	Axioms								

[1] Calculated using a lookup table listing average speed for different modes of travel (walking, driving, etc) to relate to natural language expression. The temporal distance must use the same units as the velocity unit and conversion may be required (e.g. seconds and metres/second).

[2] MaxAzimuth is a user defined function that finds the azimuth of the longest side of the boundary of the smallest surrounding rectangle (by testing ST_Length for the first and second sides only), thus representing the direction of the elongated polygon. A value of zero is returned if the side lengths are the same (and thus the polygon is not elongated).

[3] SSRectangle is a user defined function that implements an algorithm to computer the smallest surrounding rectangle of the convex hull. This differs from the minimum bounding rectangle or envelope (which is available through the ST_Envelope method) in that it is oriented in the direction that makes the smallest rectangle, whereas the envelope is oriented to the x and y axes of the coordinate reference system. The algorithm for calculating the smallest surrounding rectangle is described in http://gis.stackexchange.com/questions/22895/how-to-find-the-minimum-area-rectangle-for-given-points

[4] θ is the azimuth of the direction of the front of a.

Parameter		Values								
DIRECTION (dr): What is the cardinal direction from one object to the other?	Label	north(a,b)	south(a,b)	west(a,b)	east(a,b)	northEast(a,b)	northWest(a,b)	southEast(a,b)	southWest(a,b)	
	Illustration									
	Query (WHERE clause)	MinY[1](ST_Envelope(b)) >= MaxY(ST_Envelope(a)) AND MinX(ST_Envelope(b)) >= MinX(ST_Envelope(a)) AND MaxX(ST_Envelope(b)) <= MaxX(ST_Envelope(a))	MaxY (ST_Envelope(b)) <= MinY(ST_Envelope(a)) AND MinX(ST_Envelope(b)) >= MinX(ST_Envelope(a)) AND MaxX(ST_Envelope(b)) <= MaxX(ST_Envelope(a))	MaxX (ST_Envelope(b)) <= MinX(ST_Envelope(a)) AND MinY(ST_Envelope(b)) >= MinY(ST_Envelope(a)) AND MaxY(ST_Envelope(b)) <= MaxY(ST_Envelope(a))	MinX (ST_Envelope(b)) >= MaxX(ST_Envelope(a)) AND MinY(ST_Envelope(b)) >= MinY(ST_Envelope(a)) AND MaxY(ST_Envelope(b)) <= MaxY(ST_Envelope(a))	MinX (ST_Envelope(b)) >= MaxX(ST_Envelope(a)) AND MinY(ST_Envelope(b)) >= MaxY(ST_Envelope(a))	MaxX (ST_Envelope(b)) <= MinX(ST_Envelope(a)) AND MinY(ST_Envelope(b)) >= MaxY(ST_Envelope(a))	MinX (ST_Envelope(b)) >= MaxX(ST_Envelope(a)) AND MaxY(ST_Envelope(b)) <= MinY(ST_Envelope(a))	MaxX (ST_Envelope(b)) <= MinX(ST_Envelope(a)) AND MaxY (ST_Envelope(b)) <= MinY(ST_Envelope(a))	
	Axioms									
ADJACENCY (a): Are objects adjacent to each other?	Label	adjacent(a,b)								
	Illustration									
	Query (WHERE clause)	(ST_DWithin(a,b,σ)) AND ((ST_Touches(a,b) =1) OR (ST_Disjoint(a,b) = 1))								
	Axioms	a.adjacent(a,b) ⇒ ds.near								
COLLOCATION (cl): Are objects in the same place?	Label	within collocated(a,b)	exactly collocated(a,b)	substantially collocated(a,b)	approximately collocated(a,b)					
	Illustration									
	Query (WHERE clause)	ST_Contains(a,b) = 1	ST_Equals(a,b) = 1	(ST_Overlaps(a,b) = 1) AND (ST_Area(ST_Difference(a,b) > ST_Area(a)/2[2])	ST_DWithin(a,b,σ)					
	Axioms	cl.within collocated (a,b) ≡ t.contain(a,b)	cl.exactly collocated (a,b) ≡ t.equal(a,b)	cl.substantially collocated (a,b) ≡ t.overlap(a,b) cl.exactly collocated (a,b) ⇒ cl.substantially collocated (a,b)	cl.substantially collocated (a,b) cl.exactly collocated (a,b) cl.substantially collocated (a,b) ⇒ cl.approximately collocated (a,b) cl.exactly collocated (a,b) ⇒ cl.approximately collocated (a,b) cl.within collocated (a,b) ⇒ cl.approximately collocated (a,b)					
OBJECT PARTHOOD (op): Which part of the object is of interest?	Label	part(a,b)	whole(a,b)	rest(a,b,c)	front(a,b,D)	back(a,b,D)	left side(a,b,D)	right side(a,b,D)	middle (a,b)	corner (a,b, σ)
	Illustration									
	Query (WHERE clause)	ST_Contains(a,b) = 1	ST_Equals(a,b) = 1	ST_Equals(ST_Union(a,b),c) = 1	FrontGeometry[3](a, D) = b	BackGeometry (a, D) = b	LeftSideGeometry (a, D) = b	RightSideGeometry (a, D) = b	ST_Centroid(a) = b	PolygonAngle[4](ST_Intersection(a,b)) < σ AND PolygonAngle(ST_Intersection(a,b)) > 0 (ST_Touches(a,b) = 1) AND ST_Intersection(ST_Boundary(a), ST_Boundary(b) IS NOT NULL AND (ST_Azimuth(a) <> ST_Azimuth(b))
	Axioms	op.part(a,b) ≡ t.contain(a,b)	op.whole(a,b) ≡ t.equal(a,b)	op.rest(a,b,c) ⇒ op.part(a,c) ∧ op.part(a,b) op.rest(a,b,c) ⇒ ~t.overlap (a,b) op.rest(a,b,c) ⇒ t.touch (a,b)	op.front(a,b,D) ⇒ ~ op.back (a,b,D) ∧ ~ op.left side (a,b,D) ∧ ~ op.right side(a,b,D)	op.back(a,b,D) ⇒ ~ op.front(a,b,D) ∧ ~ op.left side (a,b,D) ∧ ~ op.right side(a,b,D)	op.left side(a,b,D) ⇒ ~ op.back (a,b,D) ∧ ~ op.front (a,b,D) ∧ ~ op.right side(a,b,D)	op.right side(a,b,D) ⇒ ~ op.back (a,b,D) ∧ ~ op.front (a,b,D) ∧ ~ op.left side(a,b,D)	op.middle (a,b,D) ⇒ ~ op.back (a,b,D) ∧ ~ op.front (a,b,D) ∧ ~ op.left side(a,b,D) ∧ ~ op.right side(a,b,D)	op.corner (a,b,D) ⇒ ~ op.junction (a,b,D)

[1] MaxX and similar user defined functions provide the maximum X coordinate of the ST_Envelope.
[2] For an overlap of 50% of a with b. Different ratios could be used as required.
[3] FrontGeometry(x, D, σ) is a user defined function that returns the geometry containing all the line segments in the simplified geometry of x whose centroid is closer to the front of the Relative Minimum Bounding Rectangle (RMBR) around x and orthogonal to D than the back, and for whom the angle between the line segment and the front of the RMBR is less than the angle between the line segment and whichever side of the RMBR is closer to the centroid of the line segment. Equivalent definitions for BackGeometry, LeftSideGeometry and RightSideGeometry exist, as per Stock (*).
[4] PolygonAngle(x) is a user defined function that returns the angle at point x of the sides of the polygon of which x is a part.

[24] D. Kemmerer. The semantics of space: integrating linguistic typology and cognitive neuroscience. *Neuropsychologia*, 44, 1607-1621, 2006.

[25] E. Klien and M. Lutz. The role of spatial relations in automating the semantic annotation of geodata. *Conference on Spatial Information Theory*, 2005.

[26] R. Moratz, J. Renz and D. Wolter. Qualitative Spatial Reasoning about Line Segments. *ECAI 2000*, 234-238, 2000.

[27] C. Schlieder. Representing Visible Locations for Qualitative Navigation. In N. Carrete and M. Singh, M.G., editors, *Qualitative Reasoning and Decision Technologies*, 523-532, CIMNE, 1993.

[28] K. Stock, R.C. Pasley, Z. Gardner, P. Brindley, J. Morley, J. and C. Cialone. Creating a corpus of geospatial language. *COSIT 2013: Conference on Spatial Information Theory*, Scarborough, UK. Lecture Notes in Computer Science (LNCS) 8116, 2013.

[29] A.C. Varzi. Parts, Wholes and Part-Whole Relations, The Prospects of mereotopology. Data and Knowledge Engineering 20, 259–286, 1996.

[30] L. Vieu and M. Aurnague. Part-of relations, functionality, dependence. In M. Aurnague, M. Hickmann and L. Vieu, editors, *The categorization of spatial entities in language and cognition*, John Benjamins (Human Cognitive Processing 20), pages 307-336, 2007.

[31] P. Vretanos. Open Geospatial Consortium Filter Encoding Specification. OGC 04-095, 2005.

[32] X. Wood and A. Galton. A Taxonomy of Collective Phenomena. *Applied Ontology*, 4, 267-292, 2009.

[33] M.F. Worboys. Nearness relations in environmental space. *International Journal of Geographical Information Science*, 15(7): 633–651, 2001.

[34] K. Zimmerman and C. Freksa. Qualitative Spatial Reasoning Using Orientation, Distance and Path Knowledge. *Applied Intelligence*, 6:49-58. 1996.

[35] J. Zwarts. Prepositional aspect and the algebra of paths. *Linguistics and Philosophy*, 28, 739-779, 2005.