

## **Geographic information - Spatial schema**

Geographic information - Spatial schema

## EESTI STANDARDI EESSÕNA

## NATIONAL FOREWORD

<p>Käesolev Eesti standard EVS-EN ISO 19107:2005 sisaldab Euroopa standardi EN ISO 19107:2005 ingliskeelset teksti.</p> <p>Käesolev dokument on jõustatud 22.02.2005 ja selle kohta on avaldatud teade Eesti standardiorganisatsiooni ametlikus väljaandes.</p> <p>Standard on kättesaadav Eesti standardiorganisatsioonist.</p>	<p>This Estonian standard EVS-EN ISO 19107:2005 consists of the English text of the European standard EN ISO 19107:2005.</p> <p>This document is endorsed on 22.02.2005 with the notification being published in the official publication of the Estonian national standardisation organisation.</p> <p>The standard is available from Estonian standardisation organisation.</p>
--	---

<p><b>Käsitlusala:</b> This International Standard specifies conceptual schemas for describing the spatial characteristics of geographic features, and a set of spatial operations consistent with these schemas. It treats vector geometry and topology up to three dimensions. It defines standard spatial operations for use in access, query, management, processing, and data exchange of geographic information for spatial (geometric and topological) objects of up to three topological dimensions embedded in coordinate spaces of up to three axes.</p>	<p><b>Scope:</b> This International Standard specifies conceptual schemas for describing the spatial characteristics of geographic features, and a set of spatial operations consistent with these schemas. It treats vector geometry and topology up to three dimensions. It defines standard spatial operations for use in access, query, management, processing, and data exchange of geographic information for spatial (geometric and topological) objects of up to three topological dimensions embedded in coordinate spaces of up to three axes.</p>
--	--

ICS 35.240.70

Võtmesõnad:

ICS 35.240.70

English version

## Geographic information - Spatial schema (ISO 19107:2003)

Information géographique - Schéma spatial (ISO  
19107:2003)

Geoinformation - Raumbezugsschema

This European Standard was approved by CEN on 24 December 2004.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the Central Secretariat has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.



EUROPEAN COMMITTEE FOR STANDARDIZATION  
COMITÉ EUROPÉEN DE NORMALISATION  
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: rue de Stassart, 36 B-1050 Brussels

# Contents

Page

Foreword .....	7
Introduction .....	8
1 Scope .....	9
2 Conformance .....	9
2.1 Overview .....	9
2.2 Conformance classes .....	11
3 Normative references .....	12
4 Terms and definitions .....	12
5 Symbols, notation and abbreviated terms .....	22
5.1 Presentation and notation .....	22
5.1.1 Unified Modeling Language (UML) concepts .....	22
5.1.2 Attributes, operations, and associations .....	22
5.1.3 Stereotypes .....	25
5.1.4 Data types and collection types .....	26
5.1.5 Strong substitutability .....	27
5.2 Organization .....	28
5.3 Abbreviated terms .....	30
6 Geometry packages .....	30
6.1 Semantics .....	30
6.2 Geometry root package .....	32
6.2.1 Semantics .....	32
6.2.2 GM_Object .....	33
6.3 Geometric primitive package .....	40
6.3.1 Semantics .....	40
6.3.2 GM_Boundary .....	41
6.3.3 GM_ComplexBoundary .....	42
6.3.4 GM_PrimitiveBoundary .....	42
6.3.5 GM_CurveBoundary .....	42
6.3.6 GM_Ring .....	42
6.3.7 GM_SurfaceBoundary .....	42
6.3.8 GM_Shell .....	43
6.3.9 GM_SolidBoundary .....	43
6.3.10 GM_Primitive .....	43
6.3.11 GM_Point .....	46
6.3.12 Bearing .....	47
6.3.13 GM_OrientablePrimitive .....	48
6.3.14 GM_OrientableCurve .....	50
6.3.15 GM_OrientableSurface .....	50
6.3.16 GM_Curve .....	51
6.3.17 GM_Surface .....	52
6.3.18 GM_Solid .....	54
6.4 Coordinate geometry package .....	55
6.4.1 DirectPosition .....	55
6.4.2 GM_PointRef .....	56
6.4.3 GM_Envelope .....	56
6.4.4 TransfiniteSet<DirectPosition> .....	57
6.4.5 GM_Position .....	57
6.4.6 GM_PointArray, GMPointGrid .....	57
6.4.7 GM_GenericCurve .....	57
6.4.8 GM_CurveInterpolation .....	61

6.4.9	GM_CurveSegment .....	62
6.4.10	GM_LineString .....	63
6.4.11	GM_LineSegment .....	64
6.4.12	GM_GeodesicString .....	65
6.4.13	GM_Geodesic .....	66
6.4.14	GM_ArcString .....	66
6.4.15	GM_Arc .....	68
6.4.16	GM_Circle .....	70
6.4.17	GM_ArcStringByBulge .....	70
6.4.18	GM_ArcByBulge .....	71
6.4.19	GM_Conic .....	72
6.4.20	GM_Placement .....	74
6.4.21	GM_AffinePlacement .....	75
6.4.22	GM_Clothoid .....	75
6.4.23	GM_OffsetCurve .....	76
6.4.24	GM_Knot .....	78
6.4.25	GM_KnotType .....	79
6.4.26	GM_SplineCurve .....	79
6.4.27	GM_PolynomialSpline .....	79
6.4.28	GM_CubicSpline .....	80
6.4.29	GM_SplineCurveForm .....	81
6.4.30	GM_BSplineCurve .....	81
6.4.31	GM_Bezier .....	82
6.4.32	GM_SurfaceInterpolation .....	83
6.4.33	GM_GenericSurface .....	83
6.4.34	GM_SurfacePatch .....	85
6.4.35	GM_PolyhedralSurface .....	86
6.4.36	GM_Polygon .....	86
6.4.37	GM_TriangulatedSurface .....	88
6.4.38	GM_Triangle .....	88
6.4.39	GM_Tin .....	89
6.4.40	GM_ParametricCurveSurface .....	90
6.4.41	GM_GriddedSurface .....	93
6.4.42	GM_Cone .....	94
6.4.43	GM_Cylinder .....	94
6.4.44	GM_Sphere .....	94
6.4.45	GM_BilinearGrid .....	95
6.4.46	GM_BicubicGrid .....	95
6.4.47	GM_BSplineSurfaceForm .....	95
6.4.48	GM_BSplineSurface .....	96
6.5	Geometric aggregate package .....	97
6.5.7	Semantics .....	97
6.5.8	GM_Aggregate .....	97
6.5.9	GM_MultiPrimitive .....	97
6.5.10	GM_MultiPoint .....	98
6.5.11	GM_MultiCurve .....	99
6.5.12	GM_MultiSurface .....	99
6.5.13	GM_MultiSolid .....	99
6.6	Geometric complex package .....	100
6.6.7	Semantics .....	100
6.6.8	GM_Complex .....	101
6.6.9	GM_Composite .....	102
6.6.10	GM_CompositePoint .....	103
6.6.11	GM_CompositeCurve .....	104
6.6.12	GM_CompositeSurface .....	105
6.6.13	GM_CompositeSolid .....	105
7	Topology packages .....	106
7.4	Semantics .....	106
7.5	Topology root package .....	108

7.5.1	Semantics .....	108
7.5.2	TP_Object.....	109
7.6	Topological primitive package .....	113
7.6.1	Semantics .....	113
7.6.2	TP_Boundary.....	113
7.6.3	TP_ComplexBoundary.....	113
7.6.4	TP_PrimitiveBoundary.....	113
7.6.5	TP_EdgeBoundary .....	114
7.6.6	TP_FaceBoundary.....	115
7.6.7	TP_SolidBoundary .....	115
7.6.8	TP_Ring.....	115
7.6.9	TP_Shell .....	115
7.6.10	TP_Primitive .....	116
7.6.11	TP_DirectedTopo .....	117
7.6.12	TP_Node.....	120
7.6.13	TP_DirectedNode .....	121
7.6.14	TP_Edge .....	122
7.6.15	TP_DirectedEdge .....	123
7.6.16	TP_Face.....	123
7.6.17	TP_DirectedFace .....	125
7.6.18	TP_Solid .....	125
7.6.19	TP_DirectedSolid .....	126
7.6.20	TP_Expression .....	126
7.7	Topological complex package.....	129
7.7.1	Semantics .....	129
7.7.2	TP_Complex .....	129
8	Derived topological relations.....	131
8.1	Introduction .....	131
8.2	Boolean or set operators.....	132
8.2.1	Form of the Boolean operators .....	132
8.2.2	Boolean Relate .....	132
8.2.3	Relation to set operations .....	133
8.3	Egenhofer operators.....	133
8.3.1	Form of the Egenhofer operators .....	133
8.3.2	Egenhofer relate.....	133
8.3.3	Relation to set operations .....	134
8.4	Full topological operators .....	134
8.4.1	Form of the full topological operators .....	134
8.4.2	Full topological relate .....	134
8.5	Combinations .....	134
Annex A	(normative) Abstract test suite.....	135
A.1	Geometric primitives .....	135
A.2	Geometric complexes.....	138
A.3	Topological complexes .....	140
A.4	Topological complexes with geometric realization .....	142
A.5	Boolean operators .....	144
Annex B	(informative) Conceptual organization of terms and definitions .....	146
B.1	Introduction .....	146
B.2	General terms .....	146
B.3	Collections and related terms.....	147
B.4	Modelling terms.....	147
B.5	Positioning terms.....	148
B.6	Geometric terms.....	148
B.7	Topological terms .....	151
B.8	Relationship of geometric and topological complexes .....	154
Annex C	(informative) Examples of spatial schema concepts .....	156
C.1	Geometry.....	156

<b>Annex D (informative) Examples for application schemata</b> .....	<b>162</b>
<b>D.1 Introduction</b> .....	<b>162</b>
<b>D.2 Simple Topology</b> .....	<b>162</b>
<b>D.3 Feature Topology</b> .....	<b>166</b>
<b>D.4 MiniTopo</b> .....	<b>167</b>
<b>Bibliography</b> .....	<b>173</b>

## Figures

Figure 1 — UML example association .....	24
Figure 2 — UML example package dependency .....	28
Figure 3 — Normative clause as UML package dependencies .....	29
Figure 4 — Geometry package: Class content and internal dependencies .....	31
Figure 5 — Geometry basic classes with specialization relations .....	32
Figure 6 — GM_Object.....	34
Figure 7 — GM_Boundary .....	41
Figure 8 — GM_Primitive .....	44
Figure 9 — GM_Point.....	46
Figure 10 — GM_OrientablePrimitive .....	49
Figure 11 — GM_Curve .....	51
Figure 12 — GM_Surface.....	53
Figure 13 — GM_Solid.....	54
Figure 14 — DirectPosition .....	56
Figure 15 — Curve segment classes .....	58
Figure 16 — Linear, arc and geodesic interpolation .....	64
Figure 17 — Arcs.....	67
Figure 18 — Conics and placements .....	73
Figure 19 — Spline and specialty curves .....	77
Figure 20 — Surface patches.....	84
Figure 21 — Polygonal surface .....	87
Figure 22 — TIN construction .....	89
Figure 23 — GM_ParametricCurveSurface and its subtypes .....	91
Figure 24 — GM_Aggregate .....	98
Figure 25 — GM_Complex.....	102
Figure 26 — GM_Composite.....	103
Figure 27 — GM_CompositePoint .....	104
Figure 28 — GM_CompositeCurve .....	104
Figure 29 — GM_CompositeSurface .....	105
Figure 30 — GM_CompositeSolid.....	106
Figure 31 — Topology packages, class content and internal dependencies.....	107
Figure 32 — Topological class diagram .....	108
Figure 33 — Relation between geometry and topology .....	109
Figure 34 — TP_Object.....	110
Figure 35 — Boundary and coboundary operation represented as associations .....	111
Figure 36 — Important classes in topology .....	112
Figure 37 — Boundary relation data types.....	114
Figure 38 — TP_Primitive .....	116
Figure 39 — TP_DirectedTopo subclasses.....	118
Figure 40 — TP_DirectedTopo .....	118

Figure 41 — TP_Node .....	121
Figure 42 — TP_Edge .....	122
Figure 43 — TP_Face .....	124
Figure 44 — TP_Solid .....	125
Figure 45 — TP_Expression .....	127
Figure 46 — TP_Complex .....	130
Figure C.1 — A data set composed of the GM_Primitives .....	157
Figure C.2 — Simple cartographic representation of sample data .....	158
Figure C.3 — A 3D Geometric object with labeled coordinates .....	160
Figure C.4 — Surface example .....	161
Figure D.1 — Packages and classes for simple topology .....	163
Figure D.2 — Topology and geometry classes in simple topology .....	164
Figure D.3 — Feature components in simple topology .....	165
Figure D.4 — Theme based feature topology .....	167
Figure D.5 — Geometric example of MiniTopo topology structure .....	168
Figure D.6 — MiniTopo .....	169
Figure D.7 — Classic MiniTopo record illustration .....	171
<b>Tables</b>	
Table 1 — Conformance classes for geometric primitives .....	11
Table 2 — Conformance classes for geometric complexes .....	11
Table 3 — Conformance classes for topological complexes .....	11
Table 4 — Conformance classes for topological complexes with geometric realizations .....	11
Table 5 — Conformance classes for Boolean operators .....	11
Table 6 — Package and classes .....	29
Table 7 — Various types of parametric curve surfaces .....	92
Table 8 — Meaning of Boolean intersection pattern matrix .....	132
Table 9 — Meaning of Egenhofer intersection pattern matrix .....	133
Table 10 — Meaning of full topological intersection pattern matrix .....	134
Table D.1 — Correspondence between original MiniTopo pointers and the current model .....	172

## Foreword

The text of ISO 19107:2003 has been prepared by Technical Committee ISO/TC 211 „Geographic information/Geomatics“ of the International Organization for Standardization (ISO) and has been taken over as EN ISO 19107:2005 by Technical Committee CEN/TC 287 „Geographic Information“, the secretariat of which is held by NEN.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by July 2005, and conflicting national standards shall be withdrawn at the latest by July 2005.

According to the CEN/CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

## Endorsement notice

The text of ISO 19107:2003 has been approved by CEN as EN ISO 19107:2005 without any modifications.

## Introduction

This International Standard provides conceptual schemas for describing and manipulating the spatial characteristics of geographic features. Standardization in this area will be the cornerstone for other geographic information standards.

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. Raster data is based on the division of the extent covered into small units according to a tessellation of the space and the assignment to each unit of an attribute value. This International Standard deals only with vector data.

In the model defined in this International Standard, spatial characteristics are described by one or more spatial attributes whose value is given by a geometric object (GM\_Object) or a topological object (TP\_Object). Geometry provides the means for the quantitative description, by means of coordinates and mathematical functions, of the spatial characteristics of features, including dimension, position, size, shape, and orientation. The mathematical functions used for describing the geometry of an object depend on the type of coordinate reference system used to define the spatial position. Geometry is the only aspect of geographic information that changes when the information is transformed from one geodetic reference system or coordinate system to another.

Topology deals with the characteristics of geometric figures that remain invariant if the space is deformed elastically and continuously — for example, when geographic data is transformed from one coordinate system to another. Within the context of geographic information, topology is commonly used to describe the connectivity of an  $n$ -dimensional graph, a property that is invariant under continuous transformation of the graph. Computational topology provides information about the connectivity of geometric primitives that can be derived from the underlying geometry.

Spatial operators are functions and procedures that use, query, create, modify, or delete spatial objects. This International Standard defines the taxonomy of these operators in order to create a standard for their definition and implementation. The goals are to:

- a) Define spatial operators unambiguously, so that diverse implementations can be assured to yield comparable results within known limitations of accuracy and resolution.
- b) Use these definitions to define a set of standard operations that will form the basis of compliant systems, and, thus act as a test-bed for implementers and a benchmark set for validation of compliance.
- c) Define an operator algebra that will allow combinations of the base operators to be used predictably in the query and manipulation of geographic data.

Standardized conceptual schemas for spatial characteristics will increase the ability to share geographic information among applications. These schemas will be used by geographic information system and software developers and users of geographic information to provide consistently understandable spatial data structures.

## 1 Scope

This International Standard specifies conceptual schemas for describing the spatial characteristics of geographic features, and a set of spatial operations consistent with these schemas. It treats vector geometry and topology up to three dimensions. It defines standard spatial operations for use in access, query, management, processing, and data exchange of geographic information for spatial (geometric and topological) objects of up to three topological dimensions embedded in coordinate spaces of up to three axes.

## 2 Conformance

### 2.1 Overview

Clauses 6 and 7 of this International Standard use the Unified Modeling Language (UML) to present conceptual schemas for describing the spatial characteristics of geographic features. These schemas define conceptual classes that shall be used in application schemas, profiles and implementation specifications. The document concerns ONLY externally visible interfaces and places no restriction on the underlying implementations other than what is needed to satisfy the interface specifications in the actual situation such as:

- Interfaces to software services using techniques such as COM or CORBA
- Interfaces to databases using techniques such as SQL
- Data interchange using encoding as defined in ISO 19118.

Few applications will require the full range of capabilities described by this conceptual schema. This clause, therefore, defines a set of conformance classes that will support applications whose requirements range from the minimum necessary to define data structures to full object implementation. This flexibility is controlled by a set of UML types that can be implemented in a variety of manners. Implementations that define full object functionality must implement all operations defined by the types of the chosen conformance class, as is common for UML designed object implementations. Implementations that choose to depend on external “free functions” for some or all operations, or forgo them altogether, need not support all operation, but shall always support a data type sufficient to record the state of each of the chosen UML type as defined by its member variables. Common names for “metaphorically identical” but technically different entities are acceptable. The UML model in this International Standard defines abstract types, application schemas define conceptual classes, various software systems define implementation classes or data structures, and the XML from the encoding standard (ISO 19118) defines entity tags. All of these reference the same information content. There is no difficulty in allowing the use of the same name to represent the same information content even though at a deeper level there are significant technical differences in the digital entities being implemented. This “allows” types defined in the UML model to be used directly in application schemas.

There are 39 conformance options for application schemas that define types for the instantiation of geometric or topological objects. They are differentiated on the basis of three criteria.

The first two criteria (complexity and dimensionality) determine the types defined in this schema that shall be implemented according to an application schema that conforms to a given conformance option. In defining the dimensionality of object types to be implemented, the application schema will be required to specify which of

the interpolation types for curves or surfaces they wish to implement. Curve implementations, for those application schemas including 1-dimensional objects, shall always include a “linear” interpolation technique. Application schema including 1-dimensional objects should always include a mechanism to approximate any curve as a line string to allow for transfer of data into simpler schema where needed. Surface implementations, for those application schemas including 2-dimensional objects, shall always include a “planar” interpolation technique. Application schema should always include a mechanism to approximate any surface as collections of planar surface patches to allow for transfer of data into simpler schema where needed. Additional curve and surface interpolation mechanism are optional, but if implemented, they shall follow the definition included in this International Standard.

The third criterion (functional complexity) determines the member elements (attributes, association roles and operations) of those types that shall be implemented. The most limited of such schema would define only data types, and may be used in the transfer of data or the passing of operational parameters to service providers.

The first criterion is level of data complexity. Four levels are identified:

- Geometric primitives
- Geometric complexes
- Topological complexes
- Topological complexes with geometric realization

NOTE Schemas for what is commonly called “spaghetti” data use only unstructured collections of geometric primitives. If single definitions of each component of geometry are required, then geometric complexes are introduced into the schema. Primitives within the same geometric complex share only boundaries. If the schema requires explicit topological information then the geometric complex is expanded to include the structure of a topological complex. The types of object included in a complex are controlled by the dimension of that complex. What is commonly called “chain-node” topology is a 1-dimensional topological complex. What is commonly called “full topology” in a cartographic 2D environment is a 2-dimensional topological complex realized by geometric objects in a 2D coordinate system.

The second criterion is dimensionality. There are four levels for simple geometry:

- 0-dimensional objects
- 0- and 1-dimensional objects
- 0-, 1-, and 2-dimensional objects
- 0-, 1-, 2- and 3-dimensional objects

However, 0-dimensional complexes provide no useful information beyond that provided by 0-dimensional geometric primitives, so conformance classes are only defined for complexes of 1-, 2-, and 3-dimensions.

The third criterion is level of functional complexity. There are three levels.

- Data types only
- Simple operations
- Complete operations

Clause 8 of this International Standard defines three groups of Boolean operators that may be used to derive topological relations between geometric and topological objects. This International Standard defines four conformance classes for application schemas that implement these operators.

**2.2 Conformance classes**

To conform to this International Standard, an implementation shall satisfy the requirements of the Abstract test suite (ATS) in Annex A for a specified conformance class. Table 1 through Table 5 identify the clauses of the ATS that apply for each conformance class.

**Table 1 — Conformance classes for geometric primitives**

Dimension	Data Types	Simple Operations	Complete Operations
0	A.1.1.1	A.1.2.1	A.1.3.1
1	A.1.1.2	A.1.2.2	A.1.3.2
2	A.1.1.3	A.1.2.3	A.1.3.3
3	A.1.1.4	A.1.2.4	A.1.3.4

**Table 2 — Conformance classes for geometric complexes**

Dimension	Data Types	Simple Operations	Complete Operations
1	A.2.1.1	A.2.2.1	A.2.3.1
2	A.2.1.2	A.2.2.2	A.2.3.2
3	A.2.1.3	A.2.2.3	A.2.3.3

**Table 3 — Conformance classes for topological complexes**

Dimension	Data Types	Simple Operations	Complete Operations
1	A.3.1.1	A.3.2.1	A.3.3.1
2	A.3.1.2	A.3.2.2	A.3.3.2
3	A.3.1.3	A.3.2.3	A.3.3.3

**Table 4 — Conformance classes for topological complexes with geometric realizations**

Dimension	Data Types	Simple Operations	Complete Operations
1	A.4.1.1	A.4.2.1	A.4.3.1
2	A.4.1.2	A.4.2.2	A.4.3.2
3	A.4.1.3	A.4.2.3	A.4.3.3

**Table 5 — Conformance classes for Boolean operators**

Set operators	A.5.1
Egenhofer operators	A.5.2
Full topological operators	A.5.3
All operators	A.5.4

### 3 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 19109:—<sup>1)</sup>, *Geographic information — Rules for application schema*

ISO 19111:—<sup>1)</sup>, *Geographic information — Spatial referencing by coordinates*

ISO/IEC 11404:1996, *Information technology — Programming languages, their environments and system software interfaces — Language-independent datatypes*

### 4 Terms and definitions

For the purposes of this document, the following terms and definitions apply. The terms are listed alphabetically in this clause. In Annex B they are organized by their conceptual relationships.

#### 4.1

##### **application**

manipulation and processing of data in support of user requirements [ISO 19101]

#### 4.2

##### **application schema**

conceptual schema for data required by one or more **applications** [ISO 19101]

#### 4.3

##### **bag**

finite, unordered collection of related items (**objects** or values) that may be repeated

NOTE Logically, a bag is a set of pairs <item, count>.

#### 4.4

##### **boundary**

**set** that represents the limit of an entity

NOTE **Boundary** is most commonly used in the context of geometry, where the set is a collection of points or a collection of objects that represent those points. In other arenas, the term is used metaphorically to describe the transition between an entity and the rest of its domain of discourse.

#### 4.5

##### **buffer**

**geometric object** that contains all **direct positions** whose distance from a specified **geometric object** is less than or equal to a given distance

#### 4.6

##### **circular sequence**

**sequence** which has no logical beginning and is therefore equivalent to any circular shift of itself; hence the last item in the sequence is considered to precede the first item in the sequence

#### 4.7

##### **class**

description of a **set** of **objects** that share the same attributes, operations, methods, relationships, and semantics [ISO/TS 19103]

---

1) To be published.