

Geographic information — Data description — Spatial schema

ICS 07.040; 35.240.70

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW



Document Sharing Hub

National foreword

This Draft for Development is the English language version of ENV 12160:1997.

This publication is not to be regarded as a British Standard.

It is being issued in the Draft for Development series of publications and is of a provisional nature due to the limited duration of the European prestandard. It should be applied on this provisional basis, so that information and experience of its practical application may be obtained.

Comments arising from the use of this Draft for Development are requested so that UK experience can be reported to the European organisation responsible for its conversion into a European Standard. A review of this publication will be initiated 2 years after its publication by the European organization so that a decision can be taken on its status at the end of its three-year life. The commencement of the review period will be notified by an announcement in the *Update Standards*.

According to the replies received by the end of the review period, the responsible BSI Committee will decide whether to support the conversion into a European Standard, to extend the life of the prestandard or to withdraw it. Comments should be sent in writing to the Secretary of BSI Technical Committee IST/36, Geographic information, at 389 Chiswick High Road, London, W4 4AL, giving the document reference and clause number and proposing, where possible, an appropriate revision of the text.

A list of organizations represented on this committee can be obtained on request to its secretary.

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the BSI Standards Catalogue under the section entitled “International Standards Correspondence Index”, or by using the “Find” facility of the BSI Standards Electronic Catalogue.

Summary of pages

This document comprises a front cover, an inside front cover, pages i and ii, the ENV title page, pages 2 to 76 and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

This Draft for Development, having been prepared under the direction of the DISC Board, was published under the authority of the Standards Board and comes into effect on 15 October 1998

© BSI 05-1999

ISBN 0 580 30526 0

Amendments issued since publication

Amd. No.	Date	Comments

Contents

	Page
National foreword	Inside front cover
Foreword	2
Text of ENV 12160	5

ICS 07.040; 35.240.70

Descriptors: Geographic information, languages, data, geometry, description

English version

Geographic information — Data description — Spatial schema

Information géographique — Description des
données — Schéma spatial

Geoinformation — Datenbeschreibung —
Raumbezugsschema

This European Prestandard (ENV) was approved by CEN on 13 September 1997 as a prospective standard for provisional application.

The period of validity of this ENV is limited initially to three years. After two years the members of CEN will be requested to submit their comments, particularly on the question whether the ENV can be converted into a European Standard.

CEN members are required to announce the existence of this ENV in the same way as for an EN and to make the ENV available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force (in parallel to the ENV) until the final decision about the possible conversion of the ENV into an EN is reached.

CEN members are the national standards bodies of Austria, Belgium, Czech Republic, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and United Kingdom.

CEN

European Committee for Standardization
Comité Européen de Normalisation
Europäisches Komitee für Normung

Central Secretariat: rue de Stassart 36, B-1050 Brussels

© 1997 CEN All rights of exploitation in any form and by any means reserved worldwide for CEN national Members.

Foreword

This European Prestandard has been prepared by the Technical Committee CEN/TC 287 “Geographic Information” the secretariat of which is held by AFNOR.

According to the CEN/CENELEC Internal Regulations, the national standards organizations of the following countries are bound to announce this European Prestandard: Austria, Belgium, Czech Republic, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and the United Kingdom.

Contents

	Page
Foreword	2
Introduction	5
1 Scope	5
1.1 Geometry	5
1.2 Topology	5
2 Normative reference	5
3 Definitions	5
4 Symbols and abbreviations	7
5 Context	8
5.1 Spatial schemas as parts of conceptual schemas	8
5.2 Description of the geometry and the topology of objects	8
5.3 Spatial schemas and spatial view	9
6 Components of spatial schemas	10
6.1 Geometric primitives	11
6.1.1 Geometric primitives and direct position	11
6.1.2 Geometric primitive Point	12
6.1.3 Geometric primitive Curve	12
6.1.4 Geometric primitive Surface	13
6.1.5 Geometric primitives Grid, Pixel, Raster band, Voxel and Raster block	16
6.1.5.1 Frame	16
6.1.5.2 Geometric primitive Grid	19
6.1.5.3 Geometric primitives Pixel, Raster band, Voxel and Raster block	20
6.2 Topological primitives	23
6.2.1 General — Introduction	23
6.2.2 Topological primitive Node	23
6.2.2.1 Description	23
6.2.2.2 Topological relations	25
6.2.3 Topological primitive Edge	25
6.2.3.1 Description	25
6.2.3.2 Topological relations	26
6.2.4 Topological primitive Face	27
6.2.4.1 Description	27
6.2.4.2 Topological relations	28
6.3 Spatial view	28
6.3.1 General — Introduction	28
6.3.2 Examples	29
6.3.2.1 Example 1	29
6.3.2.2 Example 2	31
6.3.2.3 Example 3	33
6.3.2.4 Example 4	35

	Page	Page
7 Spatial schemas	38	Figure 16 — Voxel and Raster block 22
7.1 Basic spatial schema G0	38	Figure 17 — Node 23
7.2 Conformance requirements for spatial schemas		Figure 18 — Connected node 24
7.2.1 Conformance requirement 1	45	Figure 19 — Example of intermediate node 24
7.2.2 Conformance requirement 2	45	Figure 20 — Topological relation between Isolated node and Face 25
7.2.3 Conformance requirement 3	46	Figure 21 — Edge and Face 26
7.2.4 Conformance requirement 4	46	Figure 22 — Example of topological relations between Edges 26
7.2.5 Conformance requirement 5	47	Figure 23 — The road network for the examples 29
7.3 Pre-defined spatial schemas	48	Figure 24 — A planar graph with terminating nodes and edges 30
7.3.1 Full planar graph topology spatial schema G1	48	Figure 25 — Application schema of example 1 31
7.3.2 Planar graph linear network spatial schema G2	51	Figure 26 — A non planar graph with Terminating nodes and Edges 32
7.3.3 Non planar graph linear network spatial schema G3	55	Figure 27 — Application schema of example 2 33
7.3.4 Non planar graph linear network with surfaces spatial schema G4	58	Figure 28 — A non-planar graph with Connected nodes and Edges 34
7.3.5 Spaghetti spatial schema G5	61	Figure 29 — Application schema of example 3 35
7.3.6 Bounded triangulated irregular network spatial schema G6	63	Figure 30 — Example 4, geometry 36
7.3.7 Raster image spatial schema G7	66	Figure 31 — Example 4, topology 37
7.3.8 Grid spatial schema G8	67	Figure 32 — Application schema of example 4 38
7.4 Example	69	Figure 33 — Basic spatial schema G0 44
Figure 1 — Example of an application schema with representation of geometry and topology for objects of type user-defined object type	9	Figure 34 — Raster part of basic spatial schema G0 45
Figure 2 — Example of an application schema with spatial view	10	Figure 35 — Full planar graph topology spatial schema G1 51
Figure 3 — Main components of spatial schemas	11	Figure 36 — Planar graph linear network spatial schema G2 54
Figure 4 — Point	12	Figure 37 — Non planar graph linear network spatial schema G3 57
Figure 5 — Curve	12	Figure 38 — Non planar graph linear network with surfaces spatial schema G4 61
Figure 6 — Examples of boundaries	14	Figure 39 — Spaghetti spatial schema G5 63
Figure 7 — Surface, boundary and curve	14	Figure 40 — Bounded triangulated irregular network spatial schema G6 65
Figure 8 — Example of Surface	15	Figure 41 — Raster image spatial schema G7 67
Figure 9 — Example of outer boundary of a Surface	15	Figure 42 — Grid spatial schema G8 69
Figure 10 — Example of outer and inner boundaries of an Surface	15	Figure 43 — The objects and their instances in this example 69
Figure 11 — Example of an object composed of several Surfaces	16	Figure 44 — The object instances of this example 70
Figure 12 — Rectangular frames, geographical or plane reference systems	17	Figure 45 — The two spatial views of this example 70
Figure 13 — Frame	19	Figure 46 — The topological primitives of the example 71
Figure 14 — Grid	20	
Figure 15 — Pixel and Raster band	21	

	Page
Figure 47 — The geometric primitives of the example	71
Figure 48 — Data representation of the example	72
Figure 49 — Application schema	73
Figure 50 — Extended G5 spatial schema	75

Introduction

This European Prestandard specifies the spatial schema which defines and describes the spatial aspects of the geographic data. The spatial aspects includes specialised structures, properties and associations for geometry and topology.

The components of the spatial schema are described in clause 6.

A basic spatial schema G0 is defined in 7.1 and predefined standard spatial schemas derived from G0 are defined in 7.3. Each schema is described using EXPRESS and represented using EXPRESS-G (the graphical notation of EXPRESS). The EXPRESS descriptions take precedence on the EXPRESS-G figures.

Clause 7.2 specifies the requirements to be met by the user defined spatial schemas (application schemas) to conform to this European Prestandard.

1 Scope

This European Prestandard establishes principles for describing spatial schemas which can be part of application schemas. These schemas can be used for defining, structuring, querying, updating, encoding, transforming and transferring geographic data.

The spatial schemas specify the constructs for the representation of geometry and topology of geographic objects. The constructs are defined in spatial schemas according to requirements from the applications.

1.1 Geometry

Within the scope of this European Prestandard are:

- definition of vector geometry, e.g. points, curves and surfaces;
- definition of circular arc, B-spline and clothoid;
- definition of raster geometry: grid, pixel, raster band, voxel, raster block.

The geometry can be defined in a plane cartesian reference system, a 3-dimensional cartesian reference system, geodetic coordinate systems or an astronomic reference.

Outside the scope of the first version of this European Prestandard are:

- full 3-dimensional vector surfaces and volumes, that can be considered in further extensions;
- other types of linear functions than mentioned above.

1.2 Topology

Topology is represented by means of topological primitives and the relationships between them.

The definition of the topological primitives Node, Edge and Face is within the scope of this European Prestandard.

The 3-dimensional topological primitives are outside the scope of this European Prestandard.

2 Normative reference

This European Prestandard incorporates by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to This European Prestandard only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

ENV 12009:1997, *Geographic information — Reference Model*.

prEN 12762:1997, *Geographic information — Referencing — Position*.

ISO 10303-11:1994, *Information automation systems — Product data representation and exchange — Part 11: Description methods: The EXPRESS language reference manual*.

3 Definitions

For the purpose of This European Prestandard, the following definitions apply:

3.1

boundary

closed 1-dimensional non-intersecting, either implicitly or explicitly, element

3.2

connected node

node related to one or more edges

3.3

coordinates

ordered set of two or three components of a reference system describing the spatial position of a point

3.4

curve

bounded continuous 1-dimensional geometric primitive

3.5

edge

1-dimensional topological primitive being an oriented connection between two terminating nodes, which may be the same

3.6

explicit intersection

two primitives explicitly intersect one each other when the two respective lists of coordinates have one or more repeated values between them. One primitive intersects explicitly itself when in its list of coordinates there are one or more repeated values

3.7

face

2-dimensional topological primitive

3.8

frame

2 or 3-dimensional regular tessellation of the space according to a specific local reference system

3.9

geometric primitive

description, partial or total, of the spatial aspects of an object by means of coordinates and mathematical functions

3.10

geometry

metric spatial aspects of Geographic Information

3.11

grid

point distribution defining a regular pattern, derived from the corners of the tessellation of a specific frame or a limited part of it

3.12

implicit intersection

two primitives implicitly intersect one each other when their respective interpolation methods applied to its respective lists of coordinates pass two or more times for exactly the same spatial position. One primitive intersects implicitly itself when its interpolation method is applied to its list of coordinates and passes two or more times through exactly the same spatial position

3.13

intermediate node

connected node coincident with an edge without terminating it

3.14

isolated node

node not related to any edge

3.15**node**

0-dimensional topological primitive

3.16**object**

single phenomenon existing in the real world

3.17**pixel**

2-dimensional geometric primitive which is the unit in a specific 2-dimensional frame

3.18**point**

0-dimensional geometric primitive

3.19**raster band**

2-dimensional geometric primitive which is a limited rectangular part of a specific 2-dimensional frame

3.20**raster block**

3-dimensional geometric primitive which is a limited part of a specific 3-dimensional frame

3.21**ring**

ordered set of connected edges forming a 1-dimensional closed not intersecting, either implicitly or explicitly, element

3.22**spatial view**

collection of geometric primitives and/or topological primitives conforming to a pre-defined spatial schema

3.23**surface**

bounded continuous 2-dimensional geometric primitive

3.24**terminating node**

connected node terminating an edge

3.25**topological primitive**

description, partial or total, of the topological aspects of an object

3.26**topology**

non-metric discrete spatial aspects of Geographic Information

3.27**voxel**

3-dimensional geometric primitive which is the unit in a specific 3-dimensional frame

4 Symbols and abbreviations

This European Prestandard does not define symbols or abbreviations.

NOTE This European Prestandard uses the symbols defined in ISO 10303-11, Annex D.

5 Context

5.1 Spatial schemas as parts of conceptual schemas

An application schema is a conceptual schema for a certain application. Whenever object types defined in an application schema have spatial characteristics, the representation of those characteristics is defined by a spatial schema. Such a spatial schema is an integrated part of the application schema.

This European Prestandard defines the constructs in spatial schemas.

NOTE An application schema integrates other schemas in addition to the spatial schema: semantic schema, quality schema, direct positioning schema,...

This European Prestandard defines the possibilities which may be used for a given application area.

5.2 Description of the geometry and the topology of objects

Geometry covers the quantitative description, by means of coordinates and mathematical functions, of the spatial aspects of Geographic Information. Therefore, geometry is the only aspect of Geographic Information changing when a transformation of Geodetic Reference System or Coordinate System is performed.

The geometry is described by geometric primitives. Geometric primitives are based on coordinates and mathematical functions in a well-defined reference system.

EXAMPLE 1: Point, curve and surface are types of geometric primitives.

Topology covers the qualitative discrete description of the spatial aspects of Geographic Information. Topology remains invariant if the space is deformed elastically and continuously, for example, when a transformation of Geodetic Reference System or Coordinate System is performed.

The topology of objects is described by topological primitives and the relationships established between them.

EXAMPLE 2: Node, edge and face are types of topological primitives.

References between geometric primitives and topological primitives can be set up.

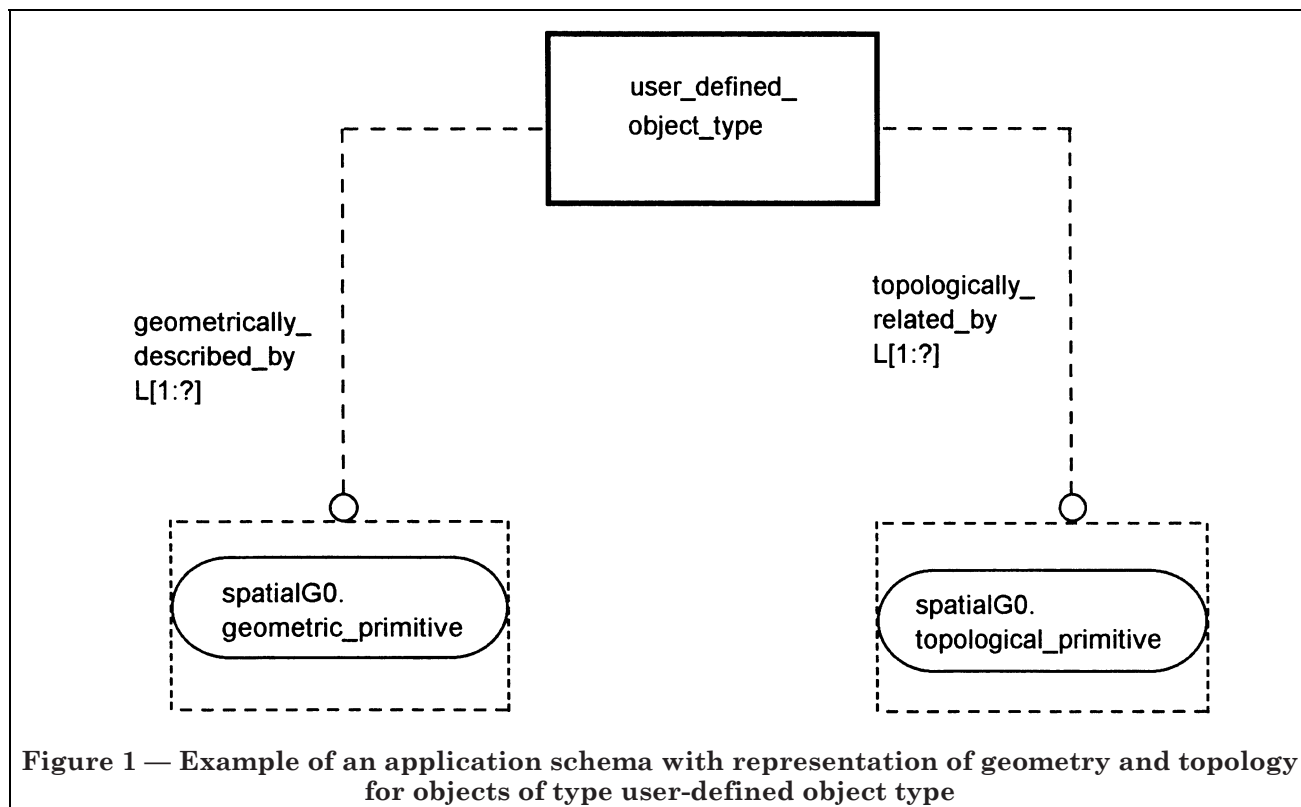
EXAMPLE 3: A node might be located by referring to a point.

A certain object can be described by geometric primitives and/or topological primitives.

EXAMPLE 4: An object can be described by geometric primitives only — e.g. spaghetti data — by topological primitives only — e.g. the air routes of an airline company — or by a combination.

One object can be described by zero to many geometric primitives and/or zero to many topological primitives. The primitives need not to be of the same type. An object need not be described by geometric primitives or by topological primitives.

NOTE 1 The application schema with spatial schema describes the geometry and topology of objects at type level, as illustrated in Figure 1.



NOTE 2 The application schema may define a relationship between an user-defined object type and any geometric primitive(s) and/or topological primitive(s). The EN Rules for application schemas will describe the composition of application schemas, using the data description techniques and integrating components for geometry, topology, quality, and direct positioning.

5.3 Spatial schemas and spatial view

A great number of combinations of geometric and topological primitives can exist. Each combination can be defined in a spatial schema.

EXAMPLE: Some of the combinations are particularly characteristic of geographic information, such as spaghetti, network, full topology, etc.

A collection of geometric primitives and/or topological primitives conforming to a spatial schema is called a spatial view. An example of an application schema with spatial view is shown in Figure 2.

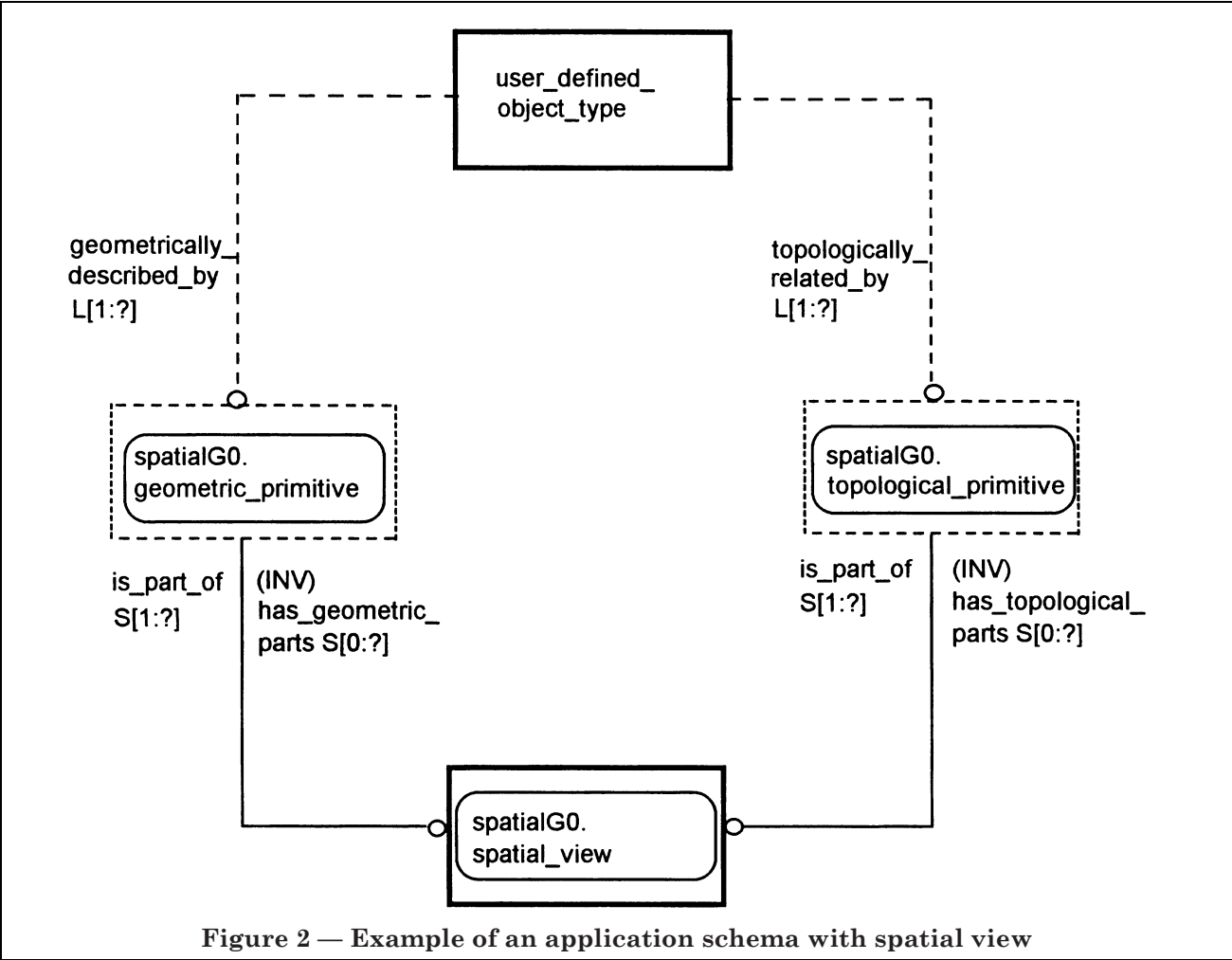


Figure 2 — Example of an application schema with spatial view

6 Components of spatial schemas

The main components of a spatial schema are illustrated in Figure 3. The definition of spatial schemas are in clause 7.

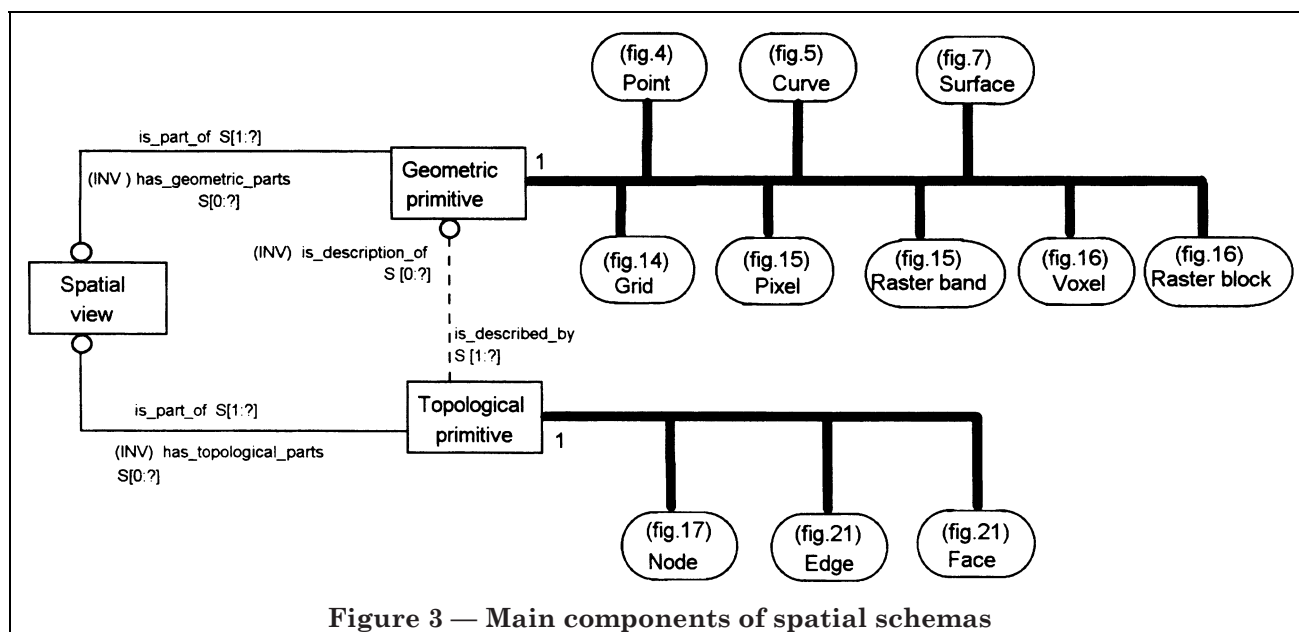


Figure 3 — Main components of spatial schemas

6.1 Geometric primitives

6.1.1 Geometric primitives and direct position

A **geometric primitive** describes, partially or totally, the spatial representation of an object directly by coordinates and mathematical functions.

The mathematical functions used for describing the geometry of an object depend on the type of Geodetic Reference System and Coordinate System defining the used Direct Position.

All the direct positions used in the description of a single geometric primitive shall be defined in the same Geodetic Reference System and Coordinate System.

As illustrated in the Figure 3, different types of geometric primitives can be used for the spatial representation of objects.

EXPRESS Specification:

ENTITY **geometric_primitive**

ABSTRACT SUPERTYPE OF

(ONEOF (point, curve, surface, grid, pixel, raster_band, voxel, raster_block));

is_part_of: SET [1:?] OF spatial_view;

INVERSE

is_description_of: SET [0:?] OF topological_primitive
FOR is_described_by;

END_ENTITY;

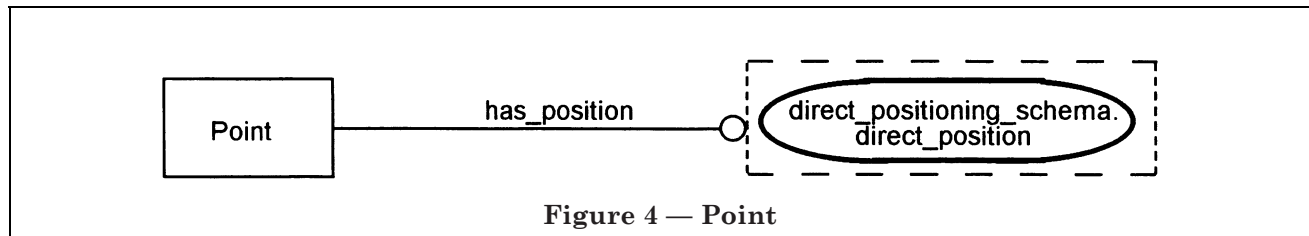
Attribute definition:

is_part_of: The set of spatial view to which the geometric primitive belongs (see 6.3).

is_description_of: The set of topological primitive (see 6.2) whose spatial position is described by the geometric primitive.

6.1.2 Geometric primitive Point

A **Point** is a 0-dimensional geometric primitive. The spatial position of a Point is described by coordinates as illustrated in Figure 4.



EXPRESS Specification:

ENTITY **point**

SUBTYPE OF (geometric_primitive);

has_position: direct_position;

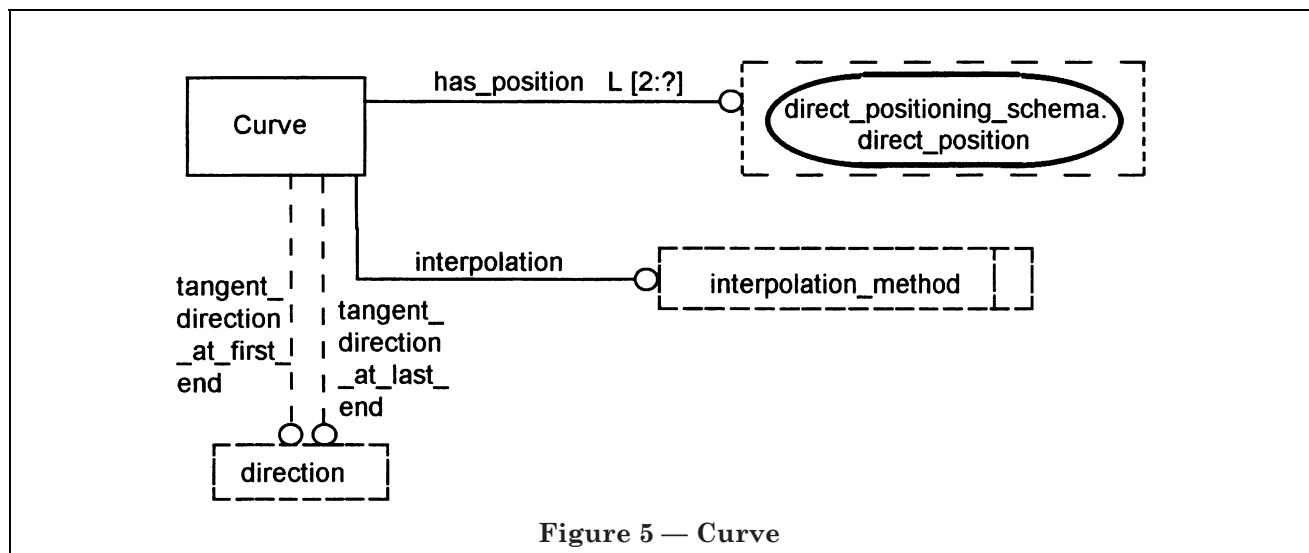
END_ENTITY;

Attribute definition:

has_position: The spatial position of the Point.

6.1.3 Geometric primitive Curve

A **Curve** is a bounded, continuous 1-dimensional geometric primitive. A Curve can be closed or not. The spatial position of a Curve is described by an interpolation method applied to a list of two or more direct positions as illustrated in Figure 5. A Curve may intersect itself explicitly or implicitly.



EXPRESS Specification:

TYPE **interpolation_method** = ENUMERATION OF (shortest_way, circular_arc, B_spline, clothoid);

END_TYPE;

TYPE **direction** = real;

WHERE

(SELF > — PI) AND (SELF <= PI);

(* Angles are counted from the first axis and counterclockwise *)

END_TYPE;

ENTITY curve

SUBTYPE OF (geometric_primitive);
has_position: LIST [2:?] OF direct_position;
interpolation: interpolation method;
tangent_direction_at_first_endOPTIONAL direction;
 :
tangent_direction_at_last_end:OPTIONAL direction;

INVERSE

is_a_component_of: SET [0:?] OF boundary
 FOR is_composed of;

(* WHERE

IR1: Two subsequent direct_position shall not be equal

IR2: All the direct positions of a curve shall have the same values of geodetic_reference_system and coordinate_system *)

END_ENTITY;Type definition:

interpolation_method: Defines the different allowed interpolation methods
 direction: An angle defining the direction of a straight curve.

Attribute definition:

has_position: The list of direct positions defining the spatial position of a Curve.

interpolation The interpolation method to be applied to the list of direct positions.

tangent_direction_at_first_end/tangent_direction_at_last_end:

The direction of tangent to the Curve used as parameter by some of the allowed interpolation methods.

is_a_component_of: The single boundary to which the curve belongs.

6.1.4 Geometric primitive Surface

A **Surface** is a bounded, continuous 2-dimensional geometric primitive, delimited by one outer non-intersecting boundary and zero or more non-nested non-intersecting inner boundaries.

A distinction is made between the geometric primitives Curve and Surface, although the Surface is described by its boundaries as illustrated in Figure 7. The main reasons why this distinction is made are:

- 1) in the case of a Curve, the Curve itself (even if it is closed) is the carrier of the information. The surface inside the closed curve is of no importance;
- 2) in the case of a Surface, its boundaries are only of secondary importance. The boundaries are only there to delimit the Surface. This is also the main reason why the Surface has only one outer boundary but may have more than one inner boundary. Inner boundaries are necessary to describe the intended Surface.

EXPRESS Specification:**ENTITY surface**

SUBTYPE OF (geometric_primitive);
has_outer_boundary: boundary;
has_inner_boundary: SET [0:?] OF boundary;

(* WHERE

IR1: Each inner boundary shall not intersect, either implicitly or explicitly outer boundary

IR2: Each inner boundary shall not intersect, either implicitly or explicitly other inner boundary

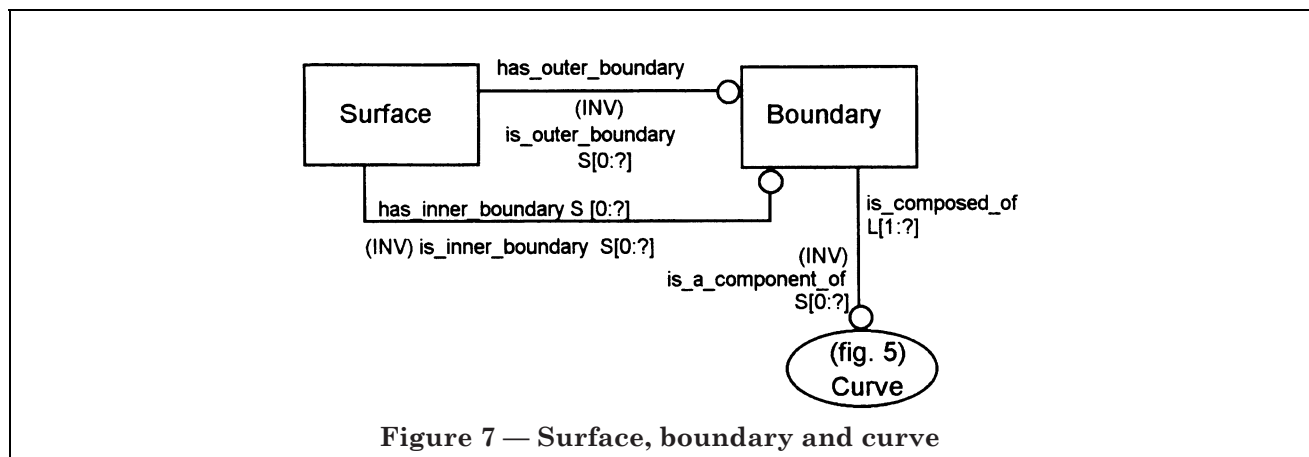
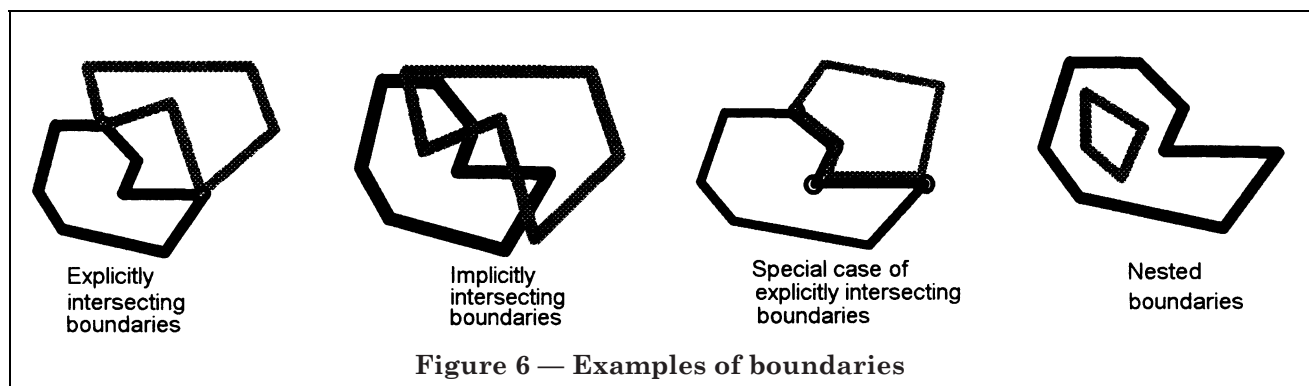
IR3: Inner boundaries shall not be nested between them

IR4: Inner boundaries shall be nested into outer boundary *)

END_ENTITY;Attribute definition:

has_outer_boundary: The outer boundary which bounds the Surface.

has_inner_boundary: The zero or more boundary which describe holes within the Surface.



A **Boundary** is a closed 1-dimensional non intersecting, either implicitly or explicitly, element. A boundary is composed of one or more curves.

EXPRESS Specification:

ENTITY **boundary**;

is_composed_of: LIST [1:?] OF UNIQUE curve;
INVERSE

is_outer_boundary: SET [0:?] OF surface
FOR has_outer_boundary;

is_inner_boundary: SET [0:?] OF surface
FOR has_inner_boundary;

(* WHERE

IR1: The boundary shall not intersect itself, either implicitly or explicitly

IR2: First_end of each curve shall be equal to previous curve last_end

IR3: First_end of the first curve shall be equal to last_end of last curve

IR4: The curves composing a boundary shall not have the same attribute values *)

END_ENTITY;

Attribute definition:

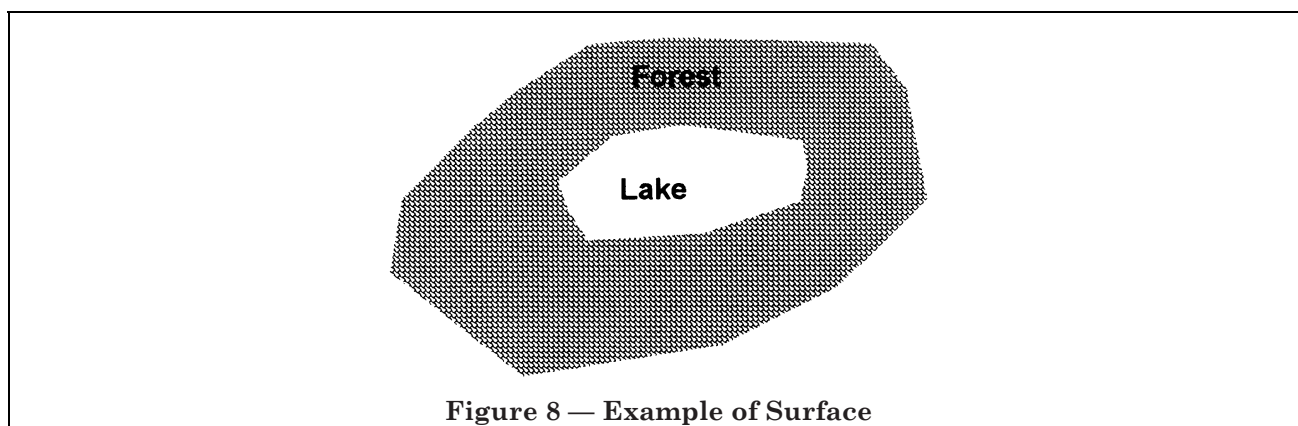
is_composed_of: The list of curve(s) defining the spatial position of the boundary.

is_outer_boundary: The Surface bounded by the boundary.

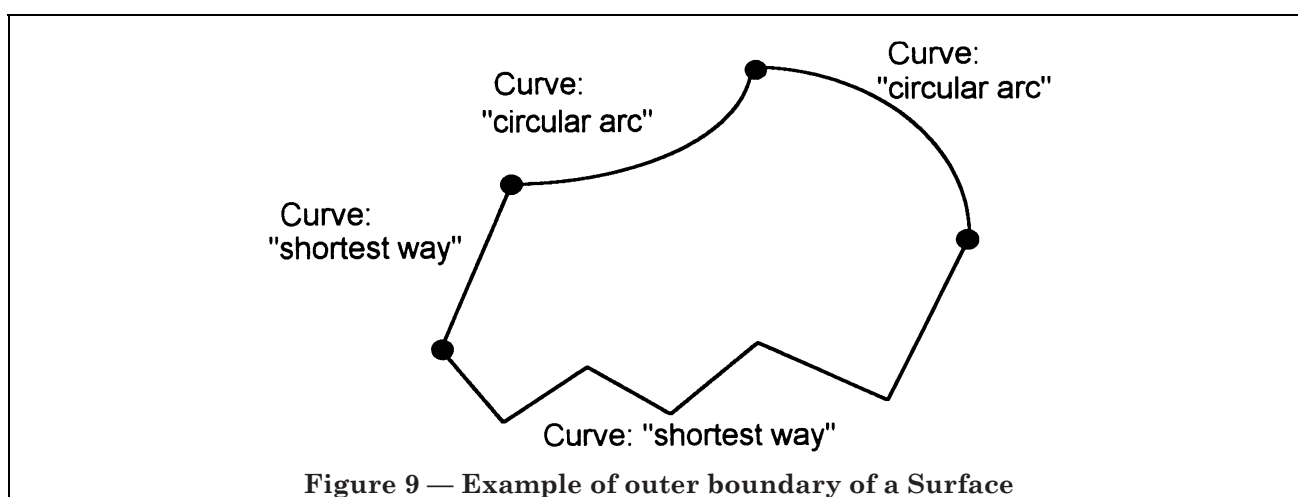
is_inner_boundary: The Surface bounded by the boundary.

The following examples illustrate the meaning of the geometric primitive Surface.

EXAMPLE 1: Consider the case of a forested Surface surrounding a lake as illustrated in Figure 8. The intended Surface is in this case described by one outer boundary and one inner boundary.

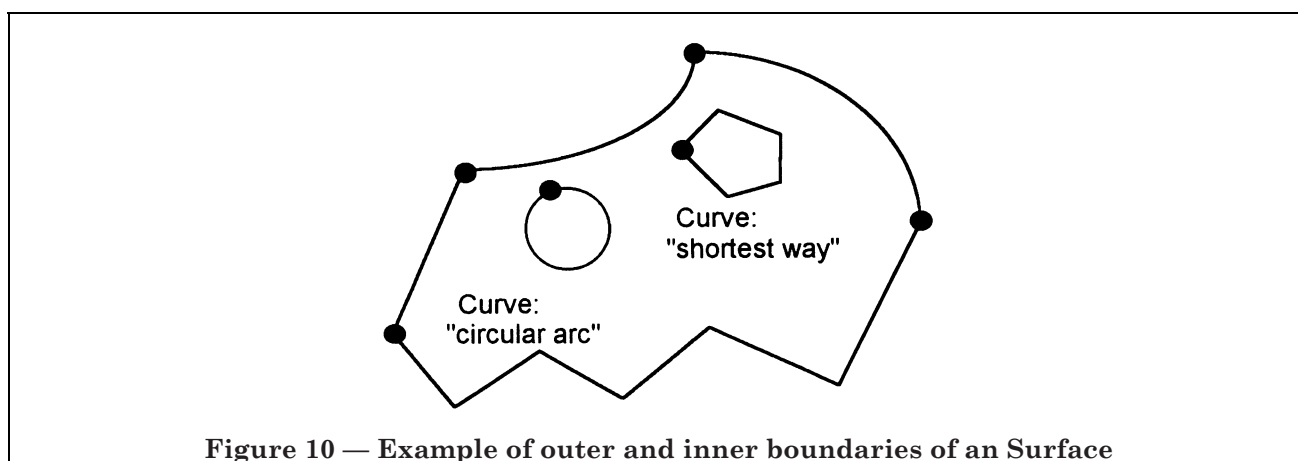


EXAMPLE 2: This example represented by Figure 9 illustrates a Surface with one outer boundary composed of two curves using the “shortest way” method and two boundary curves using the “circular arc” method.



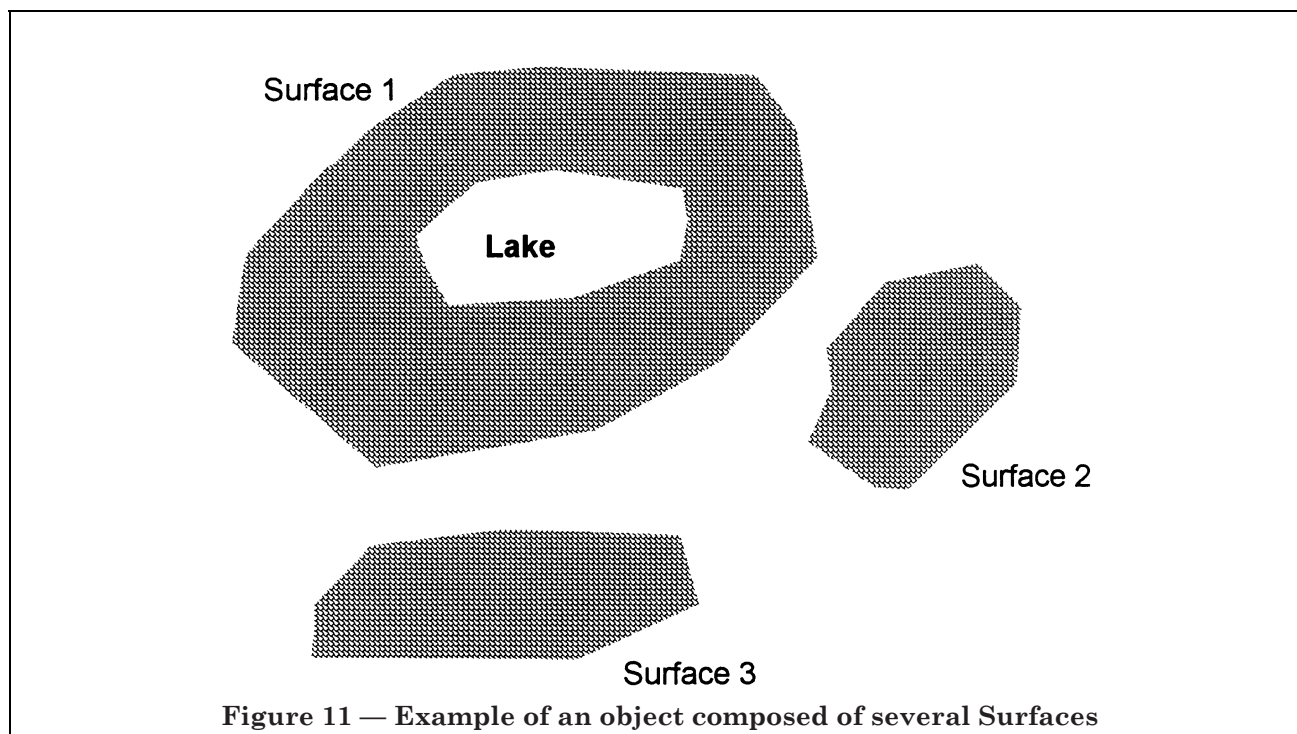
EXAMPLE 3: This example presented in Figure 10 illustrates the same Surface but with two inner boundaries:

- one inner boundary described by a curve using the “circular arc” method;
- one inner boundary described by a curve using the “shortest way” method.



NOTE This does not mean that at the “object level”, exclaves are not acceptable. It may very well happen that a forest is composed of several Surfaces. In that case, the object will be described by three instances of the geometric primitive Surface as illustrated in Figure 11:

- geometric primitive 1: Surface 1 with its inner boundary;
- geometric primitive 2: Surface 2;
- geometric primitive 3: Surface 3.



6.1.5 Geometric primitives Grid, Pixel, Raster band, Voxel and Raster block

6.1.5.1 Frame

The definition of Grid, Pixel, Raster band, Voxel and Raster Block requires the introduction of the concept of **frame**, as a tessellation on a specific local coordinate system, derived from a reference system with coordinates with two or three components.

A **frame** is based on a tessellation of a limited part of the base surface by a predefined pattern of units whose shape (rectangular, triangular or hexagonal) and size along each axis is constant and characterises the frame. A tessellation is made of one or more contiguous planes of contiguous rows and columns of regular units filling a limited part of the space. Given a numbering order along the two or three axes, each unit within a frame will be uniquely identified by two positive integers in the 2-dimensional case: a row number and a column number; and by three positive integers in the 3-dimensional case: a row number, a column number and a plane number.

The direct position of the origin of a frame is defined in a specific Geodetic Reference System and according to a Coordinate System. When it is derived from a Plane reference system, a rotation angle may be defined between the Coordinate System axes and the frame axes. The coordinates describing the spatial position of the centre of each unit may be computed from this origin, the defined size of units along the axis and the row and column number of the unit and its plane number if exists. See Figure 12 and Figure 13.

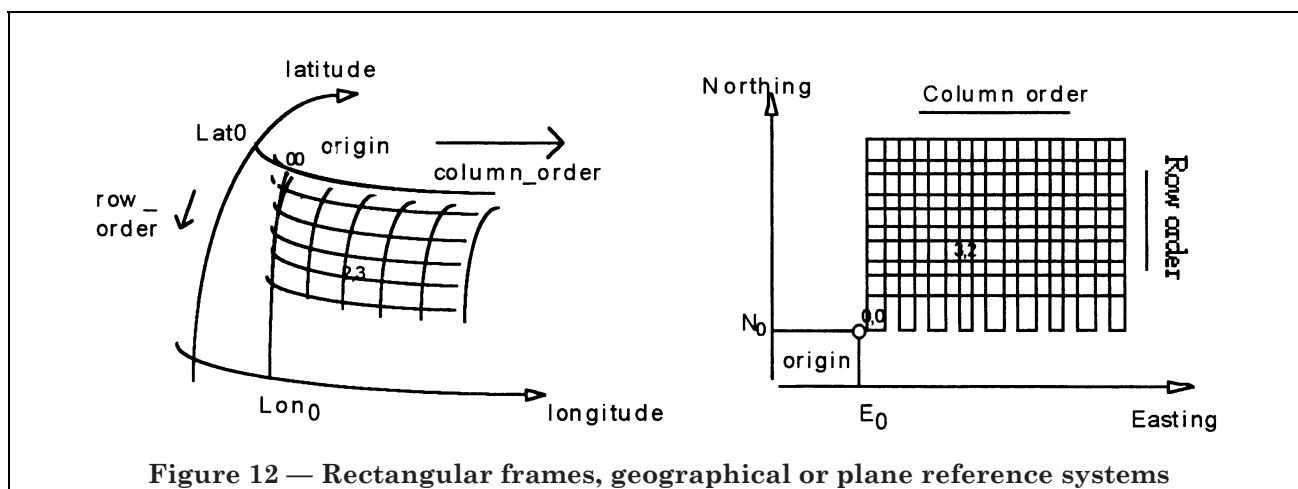


Figure 12 — Rectangular frames, geographical or plane reference systems

EXPRESS Specification:

TYPE **shape** = ENUMERATION OF (rectangular, triangular, hexagonal);
END_TYPE;

TYPE **numbering_order** = ENUMERATION OF (along_axis, reverse);
END_TYPE;

TYPE **size** = real;
END_TYPE;

TYPE **rotation_angle** = real;
WHERE

(SELF >= 0) AND (SELF < PI/2);

(* Angles are counted from the first axis and counterclockwise *)

END_TYPE;

TYPE **frame_coordinate** = integer;
WHERE

SELF >= 0;

END_TYPE;

ENTITY **frame**;

has_origin:	direct_position;
row_order:	numbering_order;
column_order:	numbering_order;
plane_order:	OPTIONAL numbering_order;
rotation:	rotation_angle;
unit_shape:	shape;
unit_size_1:	size;
unit_size_2:	OPTIONAL size;
unit_size_3:	OPTIONAL size;

INVERSE

supports_grid:	SET [0:?] OF grid FOR defined_according_to;
contains_pixel:	SET [0:?] OF pixel FOR is_the_unit_of;
supports_raster_band:	SET [0:?] OF raster_band FOR defined_according_to;
contains_voxel:	SET [0:?] OF voxel FOR is_the_unit_of;
supports_raster_block:	SET [0:?] OF raster_block FOR defined_according_to;

WHERE

DR1: (* A frame shall be used by at least one primitive *)

SIZEOF (contains_pixel) +
 SIZEOF (supports_raster_band) +
 SIZEOF (contains_voxel) +
 SIZEOF (supports_raster_block) +
 SIZEOF (supports_grid) > 0;

(* **IR1:** If unit shape is triangular or hexagonal, only unit size 1 is needed; if unit shape is rectangular, unit size 2 is needed and if unit shape is rectangular and the frame is 3-dimensional unit size 3 is also needed *)

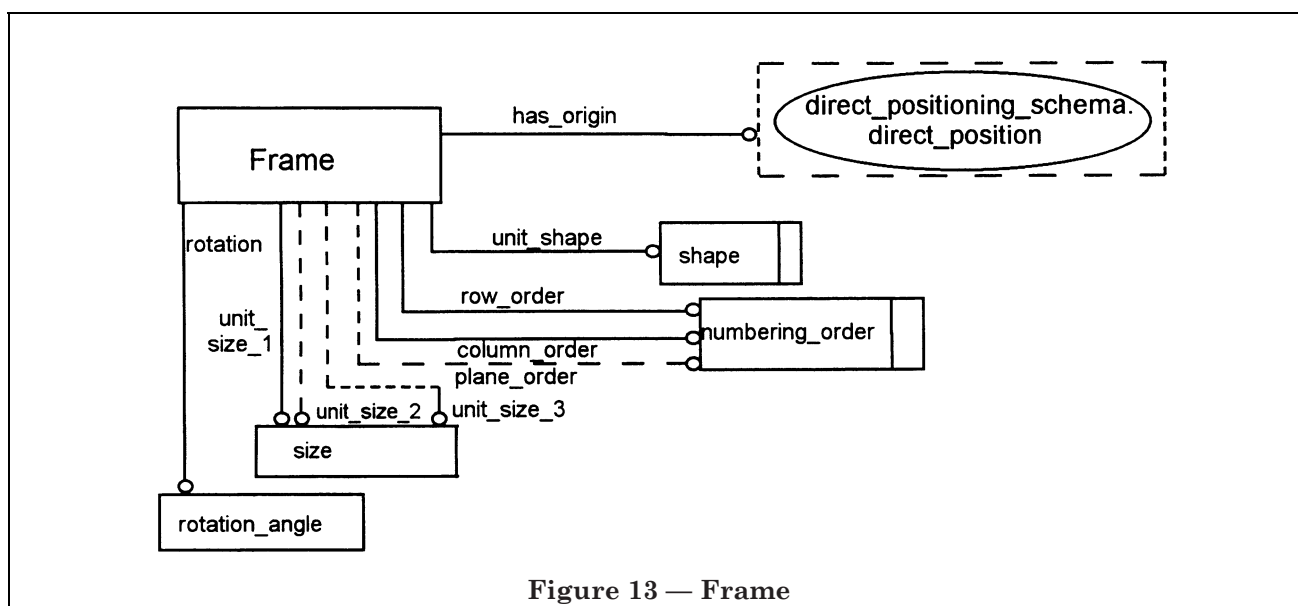
END_ENTITY;

Type definition:

shape	The allowed shapes for frame units.
numbering_order	The numbering order along a frame axis.
size	The length measure of a unit size along one axis using the measure unit defined in the Coordinate System of the origin direct_position.
rotation_angle	The measure in radians of a rotation angle.
frame_coordinate	The row, column or plane number identifying a unit.

Attribute definition:

has_origin	The direct_position of the origin frame.
row_order	The numbering order along the second axis.
column_order	The numbering order along the first axis.
plane_order	The numbering order along the third axis.
rotation	The rotation angle from the Coordinate System axes to the frame axes.
unit_shape	The shape of the frame units.
unit_size_1	When the unit shape is rectangular it is the unit size along the first axis. When the unit shape is triangular or hexagonal, it is the size of the triangle or hexagon side.
unit_size_2	The size of a unit along the second axis. It is considered only when the unit shape is rectangular.
unit_size_3	The size of a unit along the third axis. It is considered only when the unit shape is rectangular and the frame is 3-dimensional.
supports_grid	The set of Grid based on the frame.
contains_pixel	The set of Pixels contained by the frame.
supports_raster_band	The set of Raster_band based on the frame.
contains_voxel	The set of Voxels contained by the frame.
supports_raster_block	The set of Raster_block based on the frame.



6.1.5.2 Geometric primitive Grid

A **Grid** is a point distribution defining a regular pattern, derived from the corners of the tessellation of a specific frame, on a limited part of it. Consequently, if the frame is 2-dimensional the Grid will be 2-dimensional and if the Frame is 3-dimensional the Grid will be also 3-dimensional.

Therefore, a Grid is defined according to a specific frame which gives the spatial position of each point in the Grid as illustrated in Figure 14.

EXPRESS Specification:

ENTITY grid

```

SUBTYPE OF (geometric_primitive);
defined_according_to: frame;
first_row: frame_coordinate;
first_column: frame_coordinate;
first_plane: OPTIONAL frame_coordinate;
last_row: frame_coordinate;
last_column: frame_coordinate;
last_plane: OPTIONAL frame_coordinate;

```

DERIVE

```

nb_rows: integer:= last_row – first_row + 1;
nb_columns: integer:= last_column – first_column + 1;
nb_planes: integer:= NVL (last_plane, 1) – NVL (first_plane, 1) + 1;

```

END_ENTITY;

Attributes definition:

defined_according_to: The frame according to which the Grid is defined.
first_row: The frame_coordinate of the first row of the Grid.
first_column: The frame_coordinate of the first column of the Grid.
first_plane: The frame_coordinate of the first plane of the Grid.
last_row: The frame_coordinate of the last row of the Grid.
last_column: The frame_coordinate of the last column of the Grid.
last_plane: The frame_coordinate of the last plane of the Grid.
nb_rows: The number of rows in the Grid.
nb_columns: The number of columns in the Grid.
nb_planes: The number of planes in the Grid.

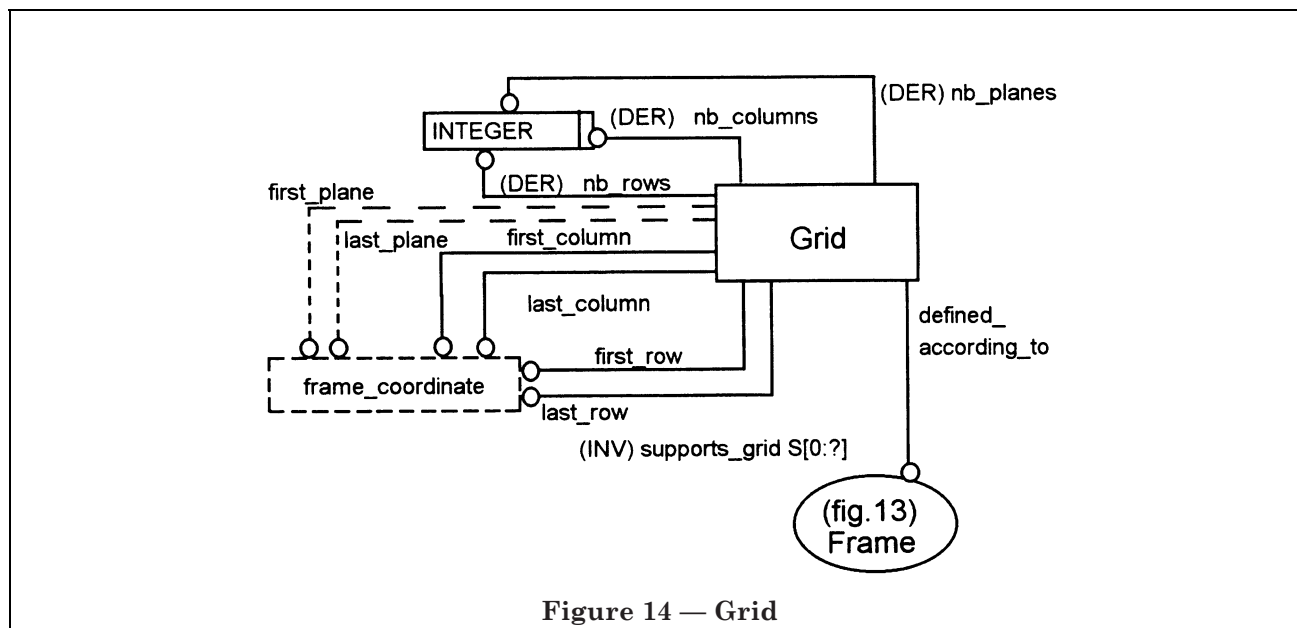


Figure 14 — Grid

6.1.5.3 Geometric primitives Pixel, Raster band, Voxel and Raster block

A **Pixel** is a 2-dimensional geometric primitive which is a unit of a specific 2-dimensional frame. The spatial position of a Pixel may be computed from its row and column number (see Figure 15).

EXPRESS Specification:

ENTITY pixel

SUBTYPE OF (geometric_primitive);

is_the_unit_of: frame;

row_number: frame_coordinate;

column_number: frame_coordinate;

END_ENTITY;

Attributes definition:

is_the_unit_of: The frame of which the Pixel is the unit.

row_number: The frame_coordinate of the row of the Pixel.

column_number: The frame_coordinate of the column of the Pixel.

A **Raster band** is a 2-dimensional geometric primitive which is a limited rectangular part of a specific rectangular 2-dimensional frame.

A Raster band is defined according to a specific frame which gives the spatial position of each unit in the Raster band (see Figure 15).

EXPRESS Specification:

ENTITY raster_band

SUBTYPE OF (geometric_primitive);

defined_according_to: frame;

first_row: frame_coordinate;

first_column: frame_coordinate;

last_row: frame_coordinate;

last_column: frame_coordinate;

DERIVE

nb_rows: integer:= last_row – first_row + 1;

nb_columns: integer:= last_column – first_column + 1;

END_ENTITY;

Attributes definition:

defined_according_to: The frame according to which the Raster_band is defined.
first_row: The frame_coordinate of the first row of the Raster_band.
first_column: The frame_coordinate of the first column of the Raster_band.
last_row: The frame_coordinate of the last row of the Raster_band.
last_column: The frame_coordinate of the last column of the Raster_band.
nb_rows: The number of rows in the Raster_band.
nb_columns: The number of columns in the Raster_band.

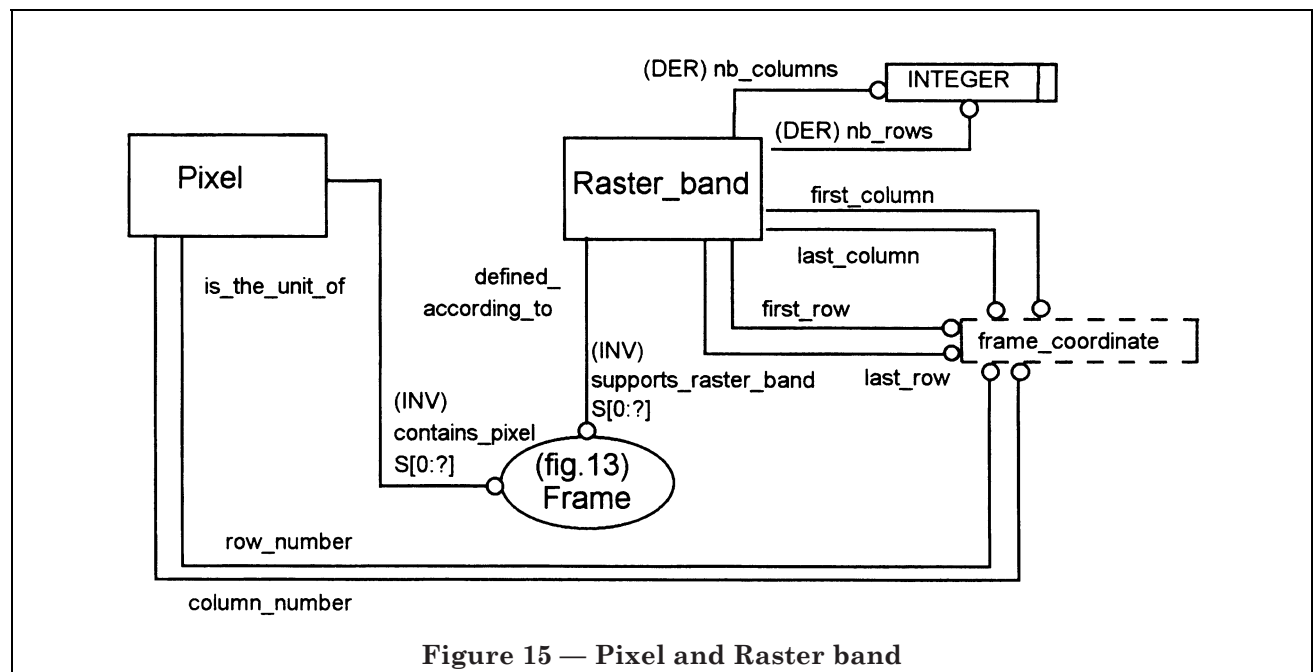


Figure 15 — Pixel and Raster band

A **Voxel** is a 3-dimensional geometric primitive which is a unit of a specific 3-dimensional frame. The spatial position of a Voxel may be computed from its row, column and plane number (see Figure 16).

EXPRESS Specification:ENTITY **voxel**

SUBTYPE OF (geometric_primitive);

is_the_unit_of: frame;

row_number: frame_coordinate;

column_number: frame_coordinate;

plane_number: frame_coordinate;

END_ENTITY;

Attributes definition:

is_the_unit_of: The frame of which the Voxel is the unit.
row_number: The frame_coordinate of the row of the Voxel.
column_number: The frame_coordinate of the column of the Voxel.
plane_number: The frame_coordinate of the plane of the Voxel.

A **Raster block** is a 3-dimensional geometric primitive which is a limited part of a specific 3-dimensional frame.

A Raster block is defined according to a specific frame which gives the spatial position of each unit in the Raster block (see Figure 16).