
**Geographic information — Schema for
coverage geometry and functions —**

**Part 1:
Fundamentals**

*Information géographique — Schéma de la géométrie et des fonctions
de couverture —*

Partie 1: Principes de base

STANDARDSISO.COM : Click to view the full PDF of ISO 19123-1:2023



STANDARDSISO.COM : Click to view the full PDF of ISO 19123-1:2023



COPYRIGHT PROTECTED DOCUMENT

© ISO 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword	v
Introduction	vii
1 Scope	1
2 Normative references	1
3 Terms, definitions, abbreviated terms and notation	1
3.1 Terms and definitions	1
3.2 Abbreviated terms	8
4 Conformance	8
4.1 Notation	8
4.2 Interoperability and conformance testing	9
4.3 Organization	9
5 Coverages	10
5.1 Overview	10
5.2 Coverage packages	11
5.3 Probing coverages: evaluate() function	12
5.4 Domain of a coverage	12
5.4.1 Concept	12
5.4.2 Coordinates	13
5.4.3 Mathematical versus physical coordinates	14
5.4.4 Coordinate reference systems and axes	14
5.4.5 Coverage classification along topological dimensions	16
5.5 Range of a coverage	16
5.6 Interpolation	17
5.6.1 Concept	17
5.6.2 Discrete and continuous coverages	18
5.7 Common point rule	19
5.8 Realization variants	19
5.8.1 Overview	19
5.8.2 Geometry/value pair view	19
5.8.3 Domain/range view	20
5.8.4 Partitioned view	21
5.8.5 Functional view	21
5.9 Envelope	21
6 Multi-point coverages	22
7 Grid coverages	23
7.1 Overview	23
7.2 Grids	24
7.2.1 Grid definition	24
7.2.2 Grid axis types	27
7.3 Rectified and referenceable grid coverages	30
7.4 Grid cells	31
7.4.1 Grid cell concept	31
7.4.2 Pixel-in-centre, pixel-in-corner	32
7.5 Grid coverage	32
7.6 Further grid coverage types	33
8 Multi-curve coverages	34
8.1 Overview	34
8.2 General multi-curve coverages	35
8.3 Segmented curve coverages	35
9 Multi-surface coverages	36
9.1 Overview	36

9.2	General multi-surface coverages.....	36
9.3	Further surface coverages.....	37
9.3.1	General.....	37
9.3.2	Thiessen polygon coverages.....	37
9.3.3	Triangulated irregular networks (TINs).....	38
10	Multi-solid coverages.....	39
Annex A	(normative) Conformance tests.....	41
Annex B	(informative) Interpolation methods.....	45
Annex C	(informative) Sequential enumeration.....	49
Annex D	(normative) Legacy data-centric coverage specification.....	60
Bibliography	77

STANDARDSISO.COM : Click to view the full PDF of ISO 19123-1:2023

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 211, *Geographic information/Geomatics*, in collaboration with the European Committee for Standardization (CEN) Technical Committee CEN/TC 287, *Geographic Information*, in accordance with the Agreement on technical cooperation between ISO and CEN (Vienna Agreement), in collaboration with the Open Geospatial Consortium (OGC), and in collaboration with the IEEE GRSS Earth Science Informatics Technical Committee.

This first edition cancels and replaces ISO 19123:2005, which has been technically revised.

The main changes are as follows:

- the document has been renamed as “Part 1: Fundamentals”, since a new “Part 2: Coverage Implementation Schema” has been published;
- the text has been simplified for better understanding;
- concepts, while in principle unchanged, have been defined more rigorously and some errors have been corrected;
- the approach to standardization taken in this document has been changed. This edition of the document defines a high-level, generic concept of coverages with an interface definition from which many different (not necessarily interoperable) implementation structures can be derived. The previous edition of this document, ISO 19123:2005, defined a single generic data structure for coverages. The previous model remains valid as one of the many possible data structures that can implement the ISO 19123-1 interface. This data structure, which is defined in [Annex D](#), supports backward compatibility. Standardization targets that referenced ISO 19123 can continue referencing those same classes, although new realizations are not encouraged to do so. It is noted, however, that the coverage definition terms in [Clause 3](#) which are owned by other documents have been updated to refer to newer editions of the documents (including their definitions) where such newer editions are available;

- all operations except *evaluate()* have been removed, for simplification purposes. ISO 19123-3 now specifies the operations requirements;
- the Scope has been extended to include Mesh;
- the concept of discrete and continuous coverages has been generalized to achieve an improved conceptual basis and to allow for coverages which are discrete along some domain axes and continuous along other domain axes. This is achieved by using the coordinate reference system axes as the basis for the definitions so that any axis individually can be discrete or continuous. Since this is a generalization of the previous concept it is backward compatible. As a side effect, this reworking has greatly simplified the structure of this document;
- updates in ISO 19103 have been reflected, and corresponding adjustments have been made where necessary. The informative Annex on “UML notation” has been deleted since UML notation is now described in ISO 19103;
- all coordinate-related definitions are based on ISO 19111, and corresponding adjustments have been made to this document as necessary;
- the definition of image CRS has been moved from ISO 19111 to this document;
- the definition of interpolation is based on the interpolation definition of ISO 19107 in order to avoid duplicate and diverging definitions;
- the UML diagrams have been redrawn for clarity, in order to correct errors, and to follow the new conventions established in ISO/TC 211;
- the bibliography has been revised to include additional references and has been reformatted.

A list of all parts in the ISO 19123 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

This document defines, at a high, implementation-independent level, the notion of coverages as digital representations of space-time varying phenomena, corresponding to the notion of a field in physics. Such coverages can be discrete or continuous.

Historically, geographic information has been treated in terms of two fundamental data types called "vector data" and "raster data".

"Vector data" deals with discrete phenomena, each of which is conceived of as a feature. The spatial characteristics of a discrete real-world phenomenon are represented by a set of one or more geometric primitives (points, curves, surfaces or solids). Other characteristics of the phenomenon are recorded as feature attributes. Usually, a single feature is associated with a single set of attribute values. ISO 19107 provides a schema for describing features in terms of geometric and topological primitives.

"Raster data", on the other hand, deals with phenomena that vary over space and time, mathematically described by "fields". It contains a set of values, each associated with one of the elements in an array of points or cells. Raster data is often associated with a method for interpolating values at spatial positions between the points or within the cells.

The coverage concept, originally adopted from the Abstract Specification of the Open Geospatial Consortium (OGC),^[15] generalizes these and further data structures, such as point clouds, into a model for representing phenomena that vary continuously over space and time, and possibly over further dimensions such as spectral bands. Formally, a coverage is a function from a spatial (such as horizontal x and y and vertical height or depth), temporal, other (in ISO 19111:2019 nomenclature: parametric) domain or any combination thereof to values of some data type.

A coverage consists of a set of spatio-temporally extended geometric (often geographic) objects, each with associated attribute values. The spatio-temporal locations with which attribute values are associated are called "direct positions".

Formally, a coverage itself is a subtype of a feature as defined in ISO 19101-1. The coverage feature is a set of features all sharing some key properties, such as the same attribute definition and coordinate reference system.

NOTE Direct positions can be of different dimensions. For example, in a raster image modelled as a coverage, the direct positions will be the grid points; in a multi-solid coverage a direct position is given by the interior of a 3D solid.

In practice, coverages encompass regular and irregular grids, point clouds and general meshes. Examples include raster data, point clouds, meshes such as triangulated irregular networks and polygon sets. Coverages are multi-dimensional, including examples such as 1D sensor timeseries, 2D satellite images, 3D $x/y/t$ image timeseries and $x/y/z$ geophysical voxel data, and 4D $x/y/z/t$ climate and ocean data. Coordinate axes of such coverages can have spatial, temporal, or any other meaning, and they can be combined freely for n -dimensional coverages.

EXAMPLE The electromagnetic spectrum is an example of an axis with neither spatial nor temporal semantics. Such a spectral axis can be defined as a "parametric CRS" as established in ISO 19111.

A coverage which provides values only at the direct positions is called "a discrete coverage". If interpolation information is added so that values can be obtained also between the coverage's direct positions, such a coverage is called "a continuous coverage".

Just as the concepts of discrete and continuous phenomena are not mutually exclusive, their representations as discrete coverages are not mutually exclusive. The same phenomenon can be represented as either a discrete feature or a coverage, depending on the particular context and requirements. A city can be viewed as a discrete coverage that returns a single value for each attribute, such as its name, area and total population, but it can also be represented as a continuous coverage that returns values such as population density, land value or air quality index for each location in the city.

A coverage, moreover, can be derived by bundling a collection of discrete features sharing a common attribute definition, the values of the coverage at each position being the values of the attributes of the feature located at that position. Conversely, a collection of discrete features can be derived from a coverage by extracting all direct positions with their associated attribute values.

The previous edition of this document, ISO 19123:2005, addressed coverage modelling on both a conceptual and (to some extent) an implementation level, effectively mixing both. Coverage modelling has now been split into two separate, but related documents: ISO 19123-1 (this document), which establishes an abstract, high-level coverage model, and ISO 19123-2, which establishes an implementation-level model ensuring interoperability, based on the concepts of ISO 19123-1. A corresponding high-level processing model for coverages is defined in ISO 19123-3.

STANDARDSISO.COM : Click to view the full PDF of ISO 19123-1:2023

Geographic information — Schema for coverage geometry and functions —

Part 1: Fundamentals

1 Scope

This document defines a conceptual schema for coverages. A coverage is a mapping from a spatial, temporal or spatiotemporal domain to attribute values sharing the same attribute type. A coverage domain consists of a collection of direct positions in a coordinate space that can be defined in terms of spatial and/or temporal dimensions, as well as non-spatiotemporal (in ISO 19111:2019, “parametric”) dimensions. Examples of coverages include point clouds, grids, meshes, triangulated irregular networks, and polygon sets. Coverages are the prevailing data structures in a number of application areas, such as remote sensing, meteorology and mapping of depth, elevation, soil and vegetation. This document defines the coverage concept including the relationship between the domain of a coverage and its associated attribute range. This document defines the characteristics of the domain. The characteristics of the attribute range are not defined in this document, but are defined in implementation standards. Consequently, the standardization target of this document consists of implementation standards, not concrete implementations themselves.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 19103, *Geographic information — Conceptual schema language*

ISO 19107, *Geographic information — Spatial schema*

ISO 19111, *Geographic information — Referencing by coordinates*

3 Terms, definitions, abbreviated terms and notation

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1.1

analytical coverage

coverage where the mapping function from domain to range is given by an analytical mathematical function

3.1.2

axis

<coordinate geometry> tuple of axis name, axis abbreviation, axis direction, axis unit and further information

Note 1 to entry: This definition is established in accordance with ISO 19111:2019, Table 26 and subclause 10.4.

Note 2 to entry: Inside a coordinate reference system (CRS) containing several axes the axis names are required to be pairwise different.

Note 3 to entry: The axis unit (of measure) defines the set of values which can be used as a coordinate along this axis. These can be numbers (such as in Latitude and Longitude) or general strings (such as in timestamps or special identifiers like “FL100” in aviation).

3.1.3

cell

<coverage> neighbourhood around a direct position in a coverage grid, not overlapping with any other direct position neighbourhood in the coverage grid

Note 1 to entry: Coverage cell is synonymous to grid cell.

Note 2 to entry: All cells of a grid coverage together establish a tessellation (i.e. complete, overlap-free cover) of the grid space.

3.1.4

continuous coverage

coverage that returns values for both direct positions and positions between direct positions

3.1.5

coordinate

one of a sequence of measures designating the position of a point

Note 1 to entry: In a coordinate reference system, the coordinate numbers are usually qualified by units. Some coordinates may use a unit representation, for example date/time conformant with ISO 8601-1. When coordinates are an index (ordinal coordinates) they are unitless (which possibly can be represented by a unit of 1).

[SOURCE: ISO 19111:2019, 3.1.5, modified — Original note 1 to entry has been amended.]

3.1.6

coordinate reference system

coordinate system that is related to an object by a datum

[SOURCE: ISO 19111:2019, 3.1.9, modified — Notes 1 and 2 to entry have been deleted.]

3.1.7

coordinate system

set of mathematical rules for specifying how coordinates are to be assigned to points

Note 1 to entry: A coordinate system contains an ordered sequence of one or more axes; their names are required to be pairwise different.

[SOURCE: ISO 19111:2019, 3.1.11, modified — Note 1 to entry has been added.]

3.1.8

coordinate tuple

tuple composed of coordinates

Note 1 to entry: The number of coordinates in the coordinate tuple equals the dimension of the coordinate system; the order of coordinates in the coordinate tuple is identical to the order of the axes of the coordinate system.

[SOURCE: ISO 19111:2019, 3.1.13]

3.1.9**coverage**

function which returns values from its range for any direct position within its domain

3.1.10**coverage coordinate reference system****coverage CRS**

coordinate reference system (CRS) in which all coordinates in a coverage domain are expressed

Note 1 to entry: Sometimes a coverage's CRS is also referred to as the coverage's native CRS to express that this is the CRS to which all the coverage's location data refer.

3.1.11**coverage dimension**

<coordinate geometry> number of separate decisions needed to describe a direct position in a coverage domain

Note 1 to entry: This is equivalent to "the number of axes in the coverage domain CRS".

Note 2 to entry: This definition is based on the term "coordinate dimension" defined in ISO 19107:2019, 3.17.

3.1.12**coverage geometry**

domain of a coverage described in terms of geometric objects

3.1.13**Delaunay triangulation**

network of triangles such that the circle passing through the vertices of any triangle does not contain, in its interior, the vertex of any other triangle

3.1.14**direct position**

<coverage> position inside one of the geometric objects in a coverage described by a coordinate tuple within the coverage coordinate reference system

Note 1 to entry: A direct position is described by an ordered sequence of coordinates. The number of elements in a direct position is established by the number of axes of the coverage CRS.

Note 2 to entry: This is consistent with the definition in ISO 19136-1:2020, 3.1.20.

3.1.15**discrete coverage**

coverage that returns value only for the direct positions within its domain

Note 1 to entry: Discrete coverages have values only for their direct positions, whereas continuous coverages can be interpolated, thereby providing values between direct positions in addition.

3.1.16**domain**

<coverage> set of geometric objects

Note 1 to entry: Examples of such geometric objects are points, lines, faces and solids. All elements within a domain (set) are of a single given type.

3.1.17**external coordinate reference system**

coordinate reference system whose datum is independent of the object that is located by it

Note 1 to entry: This term is kept only for backwards compatibility and is not used in nor fundamental to the coverage definition of this document.

[SOURCE: ISO 19130-1:2018, 3.25, modified — Note 1 to entry has been added.]

3.1.18
evaluation

<coverage> determination of the values of a coverage at a direct position within the domain of the coverage

3.1.19
feature

abstraction of real world phenomena

[SOURCE: ISO 19101-1:2014, 4.1.11, modified — Note 1 to entry has been removed.]

3.1.20
feature attribute
characteristic of a feature

Note 1 to entry: The value associated with a direct position. Also known as “feature property” and may support potential attribute, quality, or characteristic of a feature.

[SOURCE: ISO 19101-1:2014, 4.1.12, modified — Original notes to entry and examples have been deleted and a new Note 1 to entry added.]

3.1.21
function

<mathematics, programming> rule that associates each element from a domain (“source domain”, or “domain” of the function) to a unique element in another domain (“target domain”, “co-domain” or “range” of the function)

[SOURCE: ISO 19107:2019, 3.41]

3.1.22
geometric dimension

<geometry, topology> largest number n such that each point in a set of points can be associated with a subset that has that point in its interior and is topologically isomorphic to \mathbb{E}^n , Euclidean n -space

[SOURCE: ISO 19107:2019, 3.48 modified — Original notes to entry have been deleted.]

3.1.23
geometric object

<geometry> spatial object representing a geometric set

Note 1 to entry: A geometric object consists of a geometric primitive, a collection of geometric primitives, or a geometric complex treated as a single entity. A geometric object may be the spatial representation of a feature object.

[SOURCE: ISO 19107:2019, 3.49]

3.1.24
geometric set
set of direct positions

Note 1 to entry: A geometric set describes a single geometric object. The domain of a coverage consists of a set of such geometric objects. In the case of point clouds and grid data where each geometric set consists of only one single point, the domain is a set of these direct positions. For higher-dimensional geometric sets, such as curves, surfaces and solids, the set can be described through other means than enumeration, such as Boundary Representation or constructive solid geometry (CSG).

[SOURCE: ISO 19136-1:2020, 3.1.32, modified — Original note to entry has been deleted and a new Note 1 to entry has been added.]

3.1.25**georectified**

corrected for positional displacement with respect to the surface of the Earth

Note 1 to entry: This term is kept only for backwards compatibility and is not fundamental to the coverage definition of this document.

[SOURCE: ISO 19115-2:2019, 3.11, modified — Note 1 to entry has been added.]

3.1.26**georeferenceable**

associated with a geopositioning information that can be used to convert grid coordinate values to values of coordinates referenced to an external coordinate reference system related to the Earth by a datum

Note 1 to entry: This term is kept only for backwards compatibility and is not fundamental to the coverage definition of this document.

[SOURCE: ISO/TS 19163-1:2016, 4.9, modified — Note 1 has been added.]

3.1.27**georeferencing**

<coverage> geopositioning an object using a correspondence model mapping coverage domain elements to ground coordinates

3.1.28**grid**

<coverage> covering of a multi-dimensional region using quadrilateral shapes (in the 2D case) or their n -dimensional generation (in the n D case) with no overlaps and gaps

Note 1 to entry: The term "grid" originates historically from a 2D view: in ISO 19123 a grid consists of a network composed of one or more sets of curves in which the members of each set intersect the members of the other sets. Meantime, n D grids (including 1D) are known and in use. The "covering" of a region is also known as a "tessellation" in mathematics.

Note 2 to entry: The ISO 19123 definition is equivalent to the revised definition of this document.

3.1.29**grid coordinate reference system****grid CRS**

<coverage> coordinate reference system of a grid coverage

Note 1 to entry: This is consistent with the definition in ISO 19136-2:2015, 4.2.1.

3.1.30**grid coordinates**

<coverage> sequence of coordinates specifying a position on a grid

Note 1 to entry: This is consistent with the definition in ISO 19115-2:2019, 3.15.

3.1.31**grid coverage**

coverage whose domain is described by a grid

3.1.32**grid point**

point of a grid

3.1.33

image coordinate reference system

image CRS

engineering grid coverage coordinate reference system (CRS)

Note 1 to entry: The CRS of a raster image (without georeferencing) is a 2D grid with Cartesian axes; this special case of an index CRS is commonly referred to as "image CRS".

3.1.34

index coordinate reference system

index CRS

coverage coordinate reference system (CRS) in which all axes are Cartesian

3.1.35

mesh

geometry with associated topology of dimension greater than zero

Note 1 to entry: Geometry and topology are defined in ISO 19107. Mesh examples include curves, TINs and solids. Points (and point clouds) resemble geometries with dimension zero.

3.1.36

pixel

smallest element of a digital image to which attributes are assigned

Note 1 to entry: A pixel is the smallest unit of display for a visible image.

Note 2 to entry: This term originated as a contraction of "picture element".

[SOURCE: ISO 19101-2:2018, 3.28, modified — Note 1 to entry has been moved to Note 2 to entry and a new Note 1 to entry has been added.]

3.1.37

point cloud

collection of data points in 3D space

Note 1 to entry: The distance between points is generally non-uniform and hence all three coordinates (Cartesian or spherical) for each point must be specifically encoded.

[SOURCE: ISO/TS 19130-2:2014, 4.51]

3.1.38

point coverage

coverage that has a domain composed of points

3.1.39

polygon coverage

coverage that has a domain composed of polygons

3.1.40

range

<coverage> set of values associated by a function, the coverage, with the domain of a coverage

Note 1 to entry: This is consistent with the more generic definition of range in ISO 19107.

Note 2 to entry: Coverage range types and values correspond to the notion of feature attribute types and values.

3.1.41

raster

rectilinear grid

Note 1 to entry: The term is also used as an imprecise generic term for imagery and gridded coverage data.

Note 2 to entry: Historically, the term derives from the scanning lines display pattern on a cathode ray tube.

3.1.42**rectified grid**

grid for which there is an affine transformation between the grid coordinates and the coordinates of an external coordinate reference system

Note 1 to entry: If the coordinate reference system is related to the Earth by a datum, the grid is a georectified grid.

Note 2 to entry: This term is kept only for backwards compatibility and is not used in nor fundamental to the coverage definition of this document.

3.1.43**referenceable grid**

grid with an external coordinate reference system whose type is either geodetic or projected

Note 1 to entry: If the coordinate reference system is related to the Earth by a datum, the grid is a georeferenceable grid.

Note 2 to entry: This term is kept only for backwards compatibility and is not used in nor fundamental to the coverage definition of this document.

3.1.44**solid**

geometric set with three spatial dimensions

Note 1 to entry: This is consistent with the definition in ISO 19107.

Note 2 to entry: A solid may have further dimensions, such as time.

3.1.45**spatial object**

<topology, geometry> object used for representing a spatial characteristic of a feature

[SOURCE: ISO 19107:2019, 3.87]

3.1.46**spatio-temporal object**

object representing a set of direct positions in space and time

3.1.47**temporal coordinate reference system**

coordinate reference system based on a temporal datum

[SOURCE: ISO 19111:2019, 3.1.63]

3.1.48**tessellation**

partitioning of a space into a set of conterminous subspaces having the same dimension as the space being partitioned

3.1.49**Thiessen polygon**

polygon that encloses one of a set of points on a plane so as to include all direct positions that are closer to that point than to any other point in the set

3.1.50**triangulated irregular network****TIN**

tessellation composed of triangles

3.1.51

vector

quantity having direction as well as magnitude

Note 1 to entry: A directed line segment represents a vector if the length and direction of the line segment are equal to the magnitude and direction of the vector. The term vector data refers to data that represents the spatial configuration of features as a set of directed line segments.

3.2 Abbreviated terms

1D	one-dimensional
2D	two-dimensional
3D	three-dimensional
4D	four-dimensional
ASCII	American Standard Code for Information Interchange
CCD	charge-coupled device
CRS	coordinate reference system
CSG	constructive solid geometry
EPSG	European Petroleum Survey Group
GIS	geographic information system
OLAP	online analytical processing
TIN	triangulated irregular network
UML	unified modelling language

4 Conformance

4.1 Notation

In this document, conceptual schemas are presented in the Unified Modelling Language (UML). ISO 19103 presents the specific profile of UML used in this document.

Several model elements used in this schema are defined in other International Standards developed by ISO/TC 211. [Table 1](#) lists the other International Standards and packages in which UML classes used in this document have been defined.

Table 1 — Sources of externally defined UML classes

Prefix	International Standard	Package
EX	ISO 19115-1	Extent
GF	ISO 19109	General Feature Model
GM	ISO 19107	Geometry
	ISO 19111	Reference Systems

4.2 Interoperability and conformance testing

As an abstract standard, this document enables multiple different implementation patterns and architectures and does not define a standardized interoperable implementation. The abstract concepts described herein can be implemented in a variety of ways which are not necessarily interoperable. The same abstract coverage represented through two different implementation models will not necessarily be identical in their structure, and services following two different implementation models will not necessarily deliver equivalent results on equivalent queries or other operations. The purpose of the abstract description specified in this document is to provide an underlying consistency at the data model level that makes it possible to establish concretizing, interoperable standards.

Conformance testing is accomplished by manually validating a candidate concretization against all requirements by exercising the tests set out in [Annex A](#).

In an implementation standard based on this abstract specification, the semantics defined in this document will normally be cast into a concrete data model (describing data structures to be stored, transferred, ingested or extracted) and a concrete service model (describing the functionality of a service operating on coverages). Such derived models should be designed in an interoperable manner, i.e. allow concise conformance testing.

ISO 19123-2 is a companion standard. Based on this document (ISO 19123-1), ISO 19123-2 defines a concrete coverage model in the sense that interoperability can be guaranteed and interoperability tests (such as the OGC compliance tests on coverages^[12]) can be established.

[Annex D](#) defines an additional generic coverage data structure. This structure represents one possible data structure that is compatible with the interface defined in this document. This is the structure that was defined in ISO 19123:2005. The purpose of [Annex D](#) is to retain backward compatibility. This is because there exist other standards, from both ISO and other organizations, which make direct reference to the generic data classes that were defined in the previous edition of ISO 19123 and which now potentially reference the same classes as defined in [Annex D](#). The interface approach is more flexible, but the classes defined in [Annex D](#) are one valid structure that may be supported through the interface. A different standard referencing this document shall indicate whether it is compliant with either the clauses below or [Annex D](#).

4.3 Organization

The coverage schema is organized into the packages shown in, [Figure 1](#) and [Table 2](#). Each package establishes one requirements class. Grouping into these requirements classes has been done with a practical perspective in mind: realizations of this document can potentially be focused on particular structures, such as grid coverages or point clouds, while ignoring all the other options.

The name and contact information of the maintenance agency for this document can be found at www.iso.org/maintenance_agencies.

Table 2 — Conformance classes

Conformance class	Clause	URI
Coverage (Core)	5	https://standards.iso211.org/19123/-1/1/conf/core
Multi-Point Coverage	6	https://standards.iso211.org/19123/-1/1/conf/multi-point
Grid Coverage	7	https://standards.iso211.org/19123/-1/1/conf/grid
Multi-Curve Coverage	8	https://standards.iso211.org/19123/-1/1/conf/multi-curve
Multi-Surface Coverage	9	https://standards.iso211.org/19123/-1/1/conf/multi-surface
Multi-Solid Coverage	10	https://standards.iso211.org/19123/-1/1/conf/multi-solid

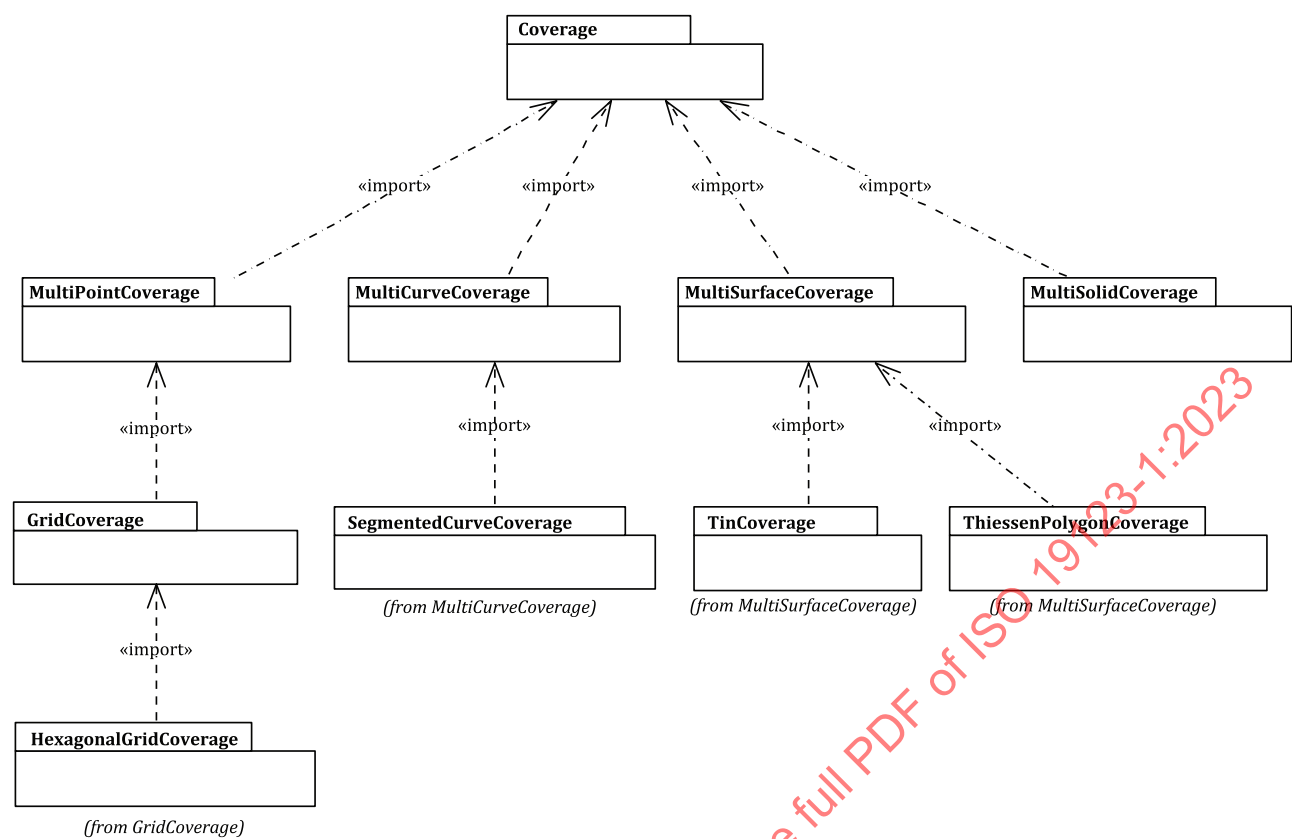


Figure 1 — Packages of the coverage schema

5 Coverages

5.1 Overview

This clause defines the conformance class Core of the package Coverage.

This document defines the term “coverage”, adopted from the Abstract Specification of the Open Geospatial Consortium, [15] to refer to any data representation that assigns values to positions. As such, a coverage can be viewed conceptually as a function which, for every value of its domain, provides a particular value taken from its range; in fact, a coverage may provide a set of values for a particular position.

An equivalent view of a coverage is that of a set of geometric objects, located in a given space and equipped with a given attribute value. Both views are supported by this document.

Coverages are multi-dimensional by nature, such as 1D sensor timeseries, 2D x/y images, 3D x/y/t image timeseries and x/y/z geophysical information, and 4D x/y/z/t atmospheric and oceanographic information. Note that “multi-dimensional” here means “1 or more dimensions” following science and engineering nomenclature, rather than the common English understanding of “multi” being “more than one”. The dimension axes spanning the coverage can be of spatial, temporal or “abstract” nature (where abstract is understood in the sense of being neither spatial nor temporal), or any combination of these. As such, a coverage is particularly suitable for the digital representation of a given space/time varying phenomenon.

EXAMPLE 1 Point clouds can be modelled as multi-point coverages. Several observation values can be acquired for a particular location, and reading such a location will result in the set of all values observed. On a grid, on the other hand, it is ensured by definition that every grid location will have exactly one value assigned.

EXAMPLE 2 A satellite image timeseries has two spatial and one temporal axis. A geo-statistical datacube can have a spatial and temporal axis and a population density axis.

The set of locations (in a general sense) where a coverage has objects sitting and, hence, has values to offer is called the coverage’s “domain” and the locations are called “direct positions”. The set of all these values is the coverage’s “range”, described by the “range type”.

Coordinates in a coverage are all expressed in the same coordinate reference system (CRS). Based on ISO 19111:2019, such a CRS may either be defined directly (such as in the EPSG collection of CRSs^[21]) or it may be composed from several CRS definitions.

EXAMPLE 3 The CRS of a satellite image timeseries consists of two spatial and one temporal axis, in a given order. Coordinates along each axis are expressed accordingly: horizontal spatial coordinates can be expressed in degrees Latitude and Longitude, as laid down in the EPSG:4326 CRS, while time can be expressed in seconds since epoch or in a given calendar, like Proleptic Gregorian.

In terms of common data structures, coverages encompass regular and irregular grids (requirements class “Grid Coverage”), point clouds (requirements class “Multi-Point Coverage”), and general meshes (requirements classes “Multi-Curve Coverage”, “Multi-Surface Coverage”, and “Multi-Solid Coverage”).

Different views on this coverage concept exist. Therefore, several practically relevant and compatible views are explicitly supported.

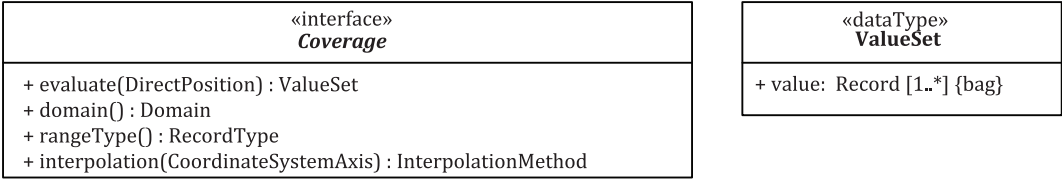
- In a mathematical view, a coverage is defined as a function $C: D \rightarrow R$ with domain D and range R which delivers some value for each element from D . This view is realized as one variant of coverage modelling in subclass CoverageByDomainAndRange (see 5.8.3).
- Geometrically, a coverage can be seen as a set of geometric objects. This view is provided as another variant in subclass CoverageByValuePair (see 5.8.2).
- In standardization terms, a coverage can be described as a set of “features” (as per ISO 19101-2). Each feature has a location and attributes. When considered together, the set of all these pairs defines the coverage’s mapping from location to values. “Geometry” here is understood in the widest possible sense, including all spatial and temporal dimensions and beyond. Features in such a collection forming a coverage must have locations expressed in the same CRS (the coverage’s CRS), and all attribute values must share the same data type (the coverage’s range type). See 5.8 for details.

5.2 Coverage packages

This clause describes the package Coverage in terms of interfaces (as per ISO 19103) of its central class, Coverage. A set of requirements for compliance with this interface description is defined that are based on the concepts of UML modelling from ISO 19103, coordinates and collections from ISO 19107, and coordinate systems from ISO 19111.

Requirement 1: <https://standards.iso211.org/19123-1/1/req/core/packages>

An instantiation of the package Coverage shall have all instances and properties specified for this package, its contents, and its dependencies for Coverage as per Figure 1 and Figure 2.



NOTE A concretization can potentially add further features. For example, ISO 19123-2 adds a metadata component.

Figure 2 — Coverage central classes

The UML interface modelling approach allows multiple different data structures to be defined which exhibit the same behaviour through the interface. Standardization targets are implementation standards based on the abstract model defined in this document. One such example is ISO 19123-2 which has concrete implementations as its standardization target. Note that in general different realizations of this document are not necessarily interoperable.

The original (now obsolete) model from ISO 19123:2005 is included as [Annex D](#), outside the Core conformance class. Any implementation standard referring to [Annex D](#) instead of [Clauses 5](#) to [10](#) (the conformance classes listed in [Table 2](#)) shall indicate this information clearly in its conformance statement.

Requirement 2: <https://standards.iso211.org/19123/-1/1/req/core/model>

An instantiation of the package Coverage shall indicate whether it realizes either [Clauses 5](#) to [10](#) or [Annex D](#).

5.3 Probing coverages: evaluate() function

One way to define the semantics of a coverage is via a probing function which, for some direct position coordinate expressed in the coverage's CRS, returns the set of values associated with it. This function can be defined, for a coverage C with domain D and value set V of that coverage, as:

$$evaluate_C: D \rightarrow V, evaluate_C(p) = \bigcup_{f \in C} f.contains(p)$$

based on the ISO 19107 *contains()* predicate.

NOTE This probing function serves for definition purposes only. Implementation is not required. In practice, another retrieval functionality is desirable, such as the OGC Web Coverage Service (WCS)^[14] bounding box subsetting and geo datacube analytics as per ISO 19123-3.

While, in general, more than one value can potentially be returned for a particular direct position, sometimes exactly one value will be delivered. This can occur in two cases, explained in the following examples.

EXAMPLE 1 In the case of a grid coverage, no two direct positions can share the same coordinates by definition, so there is always one value available via *evaluate()*.

EXAMPLE 2 More than one value per direct position can occur in several cases, such as in point clouds with two points incidentally sharing the same coordinates but having different values, or in MultiSolid coverages that overlap.

5.4 Domain of a coverage

5.4.1 Concept

The coverage domain describes the positions in the coverage's multi-dimensional space for which values are available. Within this multi-dimensional space defined by the domain's CRS, the coverage domain contains a set of geometric objects which together determine the direct positions, i.e. the locations in this space where the coverage offers a value. This description can be given through direct enumeration of the direct positions (for example, point clouds) or through containment descriptions (for example, areas and volumes), or another mechanism (for example, ground control points in sensor models). The coverage's "extent" gives a bounding box (i.e. lower and upper bounds along every coordinate axis) within which all its direct positions are located so that a quick overview on the footprint of the coverage can be obtained.

All coordinates and coordinate reference systems in this document are based on ISO 19111.

Requirement 3: <https://standards.iso211.org/19123/-1/1/req/core/coordinates>

An instantiation of the package Coverage shall make use of the ISO 19111 package RS_ReferenceSystem.

EXAMPLE 1 The space spanned in coverages representing 1D temperature measurement timeseries, 2D x/y images of the Earth surface, 3D x/y/t image timeseries and x/y/z subsurface voxel data, and 4D x/y/z/t atmospheric and ocean data, all are described through an appropriate CRS with the respective dimension.

The geometric objects in a coverage domain are not strictly confined through 3-dimensional physical space. They can be m -dimensional objects where $m \leq n$ for an n -dimensional coverage. Coverage subtypes are defined in terms of their domain types in [Clause 6](#) onwards.

EXAMPLE 2 A 3-dimensional x/y/t image timeseries datacube can be composed of 0-dimensional points (pixels). A coverage domain of dimension 3 can be composed of points, curves, surfaces or solids, while a domain of dimension 2 can be composed only of points, curves or surfaces.

5.4.2 Coordinates

For an n -dimensional coverage, $n > 0$, points in the coverage domain are addressed through n -tuples of coordinates based on the n -dimensional coverage's CRS (see [Figure 3](#)).

Often the unit of measure of an axis is numeric, such as degrees or metres or seconds. However, there can be several representations of coordinate measurements, such as degree/minute/seconds or Proleptic Gregorian calendar time/date strings. Finally, axes without any spatial nor temporal semantics, can be added (such as geo-statistics measures). Therefore, in this document, DirectPosition defining coverage coordinates takes a broader approach to coordinate values than ISO 19111 class Coordinate and ISO 19107 class DirectPosition by allowing not only numbers, but general strings.

Requirement 4: <https://standards.iso211.org/19123/1/1/req/core/directpositions>

Domain coordinates in a coverage shall be expressed as DirectPositions as described in [Figure 3](#).

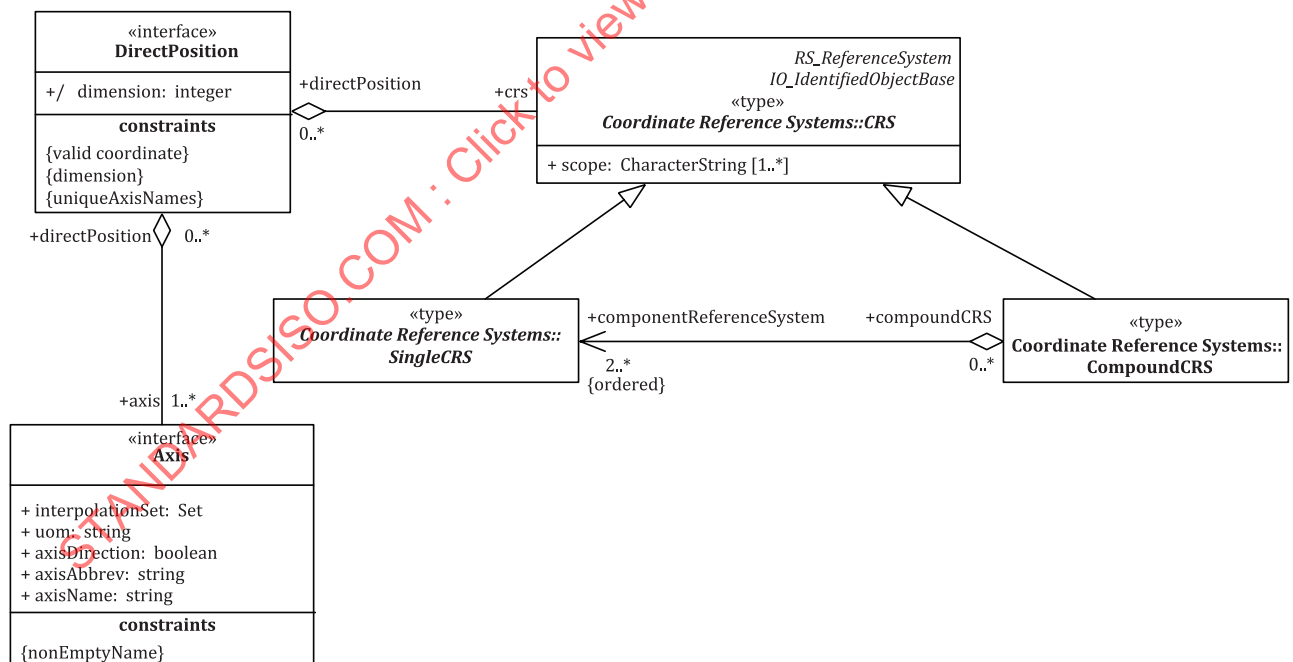


Figure 3 — Class DirectPosition

EXAMPLE The following 4D coordinate tuple illustrates different coordinate types which can possibly describe Latitude, Longitude, time, and height:

(43.001, 25, "2022-07-24", "FL100")

5.4.3 Mathematical versus physical coordinates

A coverage domain can represent physical as well as imaginary nD spaces. For example, geographic data typically have two horizontal axes, possibly one height axis (expressing altitude or depth), and possibly a time axis. Climate modelling adds a second time axis for differentiating the model run time from time modelled. By combining one or more of these axes, multi-dimensional spatio-temporal objects can be built. Further, axes can represent alternate representations of coordinate measurements. “Abstract” (in the sense of non-spatio-temporal) axes may occur as well, as in Online Analytical Processing (OLAP) where, for example, time/product/subsidiary axes are common. Additionally, bands in hyperspectral imagery often appear as a numbered sequence when there are hundreds of such bands, such as the Hyperion instrument on board of EO-1 with its 220 bands.^[25] With natural numbers, addressing becomes less wieldy.

NOTE 1 In ISO 19111, non-spatiotemporal CRSs are referred to as “parametric”. CRSs can be composed from other CRSs which can have one or more axes embedded.

NOTE 2 Spatial dimensions can potentially become non-spatial at a certain level of generalization. For example, cities in Europe can be expressed through coordinates in the first place, but then at a higher level they can be abstracted to be in Bavaria, in the Alps region, etc. which is at a symbolic level. This can be represented through dimension hierarchies like in online analytical processing (OLAP).

5.4.4 Coordinate reference systems and axes

The n -dimensional space in which a coverage resides is described by its CRS. Such CRSs often readily exist (such as in the EPSG catalogue) or they can be built by combining existing CRSs into a new, higher-dimensional one ([Figure 3](#)). Mechanisms to do so are out of scope of this document. They are defined in ISO 19111:2019.

Requirement 5: <https://standards.iso.org/19123/1/1/req/core/crs>

The coverage CRS shall be expressed in accordance with ISO 19111.

NOTE Sometimes there are constraints excluding some combinations. For example, Latitude and Longitude in ISO 19111:2019 always come in pairs and building CRSs that contain only one of these pairs is not permitted.

Additionally, this document establishes CRS definitions for unitless abstract spaces. The general term “index CRS” refers to a non-georeferenced CRS where all axes are numbers. This induces a family of nD CRSs which, by convention, are named index1D, index2D, index3D, etc.

For backwards compatibility, image CRS exists for the non-georeferenced case (class imageCRS in [Figure 4](#)).

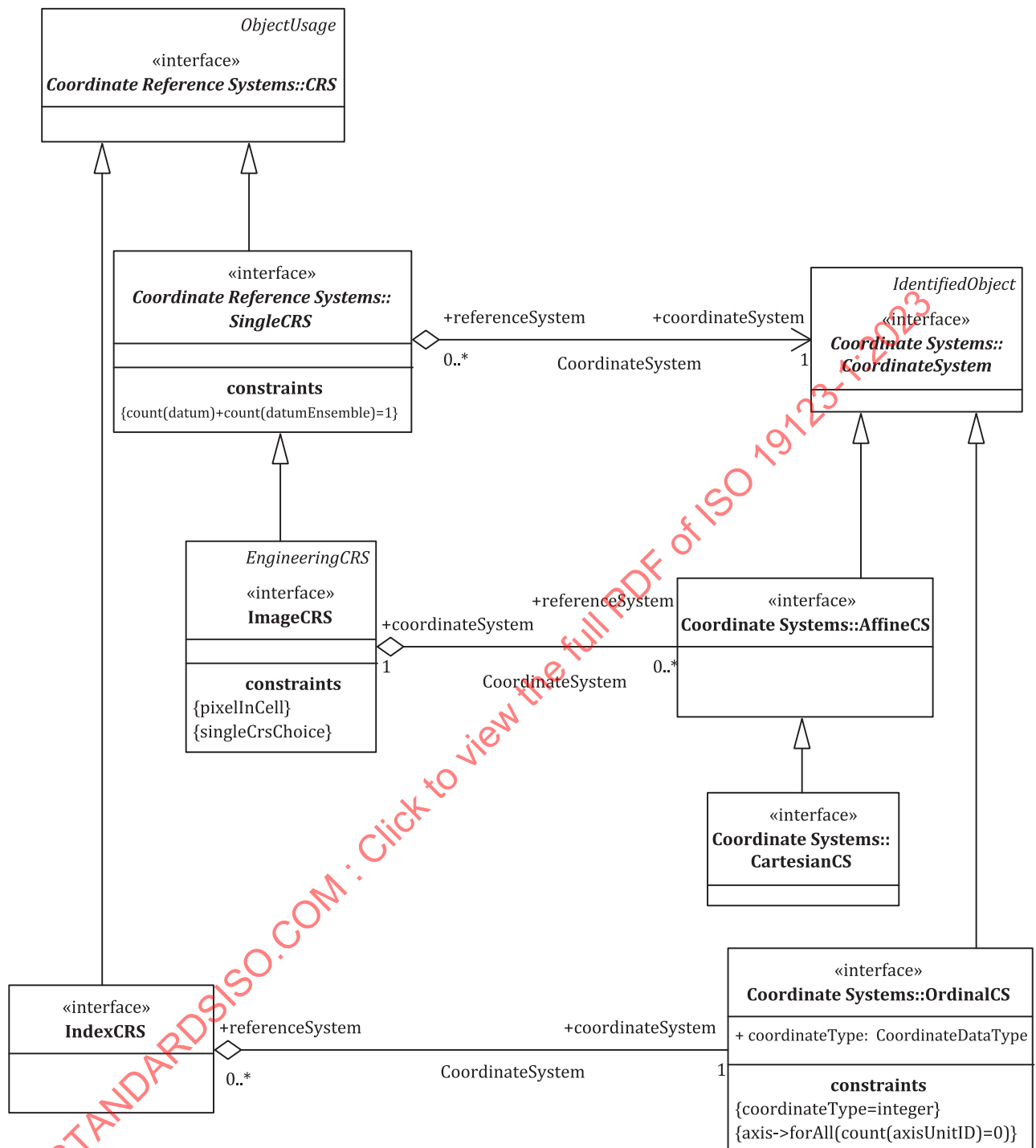


Figure 4 — IndexCRS and ImageCRS classes

EXAMPLE 1 A 4D domain can be described through axes Latitude and Longitude (given by EPSG:4326) combined with a time axis as per ISO 8601-1 combined with an indexCRS2D for spectral bands.

EXAMPLE 2 The direct positions in a coverage's domain can be enumerated (example: point sets in a multi-point coverage) or they can be given implicitly (example: corner points indicating the set of all raster positions in a Rectified grid coverage).

In defining multi-dimensional coverage spaces, this document takes an approach where axes are first-class citizens, as opposed to ISO 19111 which operates mainly on a CRS level. Both views are consistent and compatible as CRSs describe ordered lists of axes.

For the purpose of coverages, CRS axes are enriched with further properties required for proper modelling of various coverage types, in particular, grid coverages (see 7.2).

EXAMPLE 3 The following is a non-exhaustive list of possible axes in coverage Domains:

- a CRS consisting of a single time axis can model a timeseries of measurements. Time coordinates can be expressed, among others, in seconds since epoch (often: January 1, 1970) or as time/date strings of the Proleptic Gregorian Calendar defined in ISO 8601-1, such as “2021-05-12”;
- pressure altitudes, measured in hectopascal (hPa), as well as Flight Levels, expressed in 100 feet steps (e.g. “FL150” for 15000 feet above Mean Sea Level), represent “proxies” for altitude used in aviation. In ISO 19111 this can be modelled through parametric CRSs;
- spectral frequencies can define a coverage axis. In ISO 19111 this can be modelled through parametric CRSs.

5.4.5 Coverage classification along topological dimensions

The coverage concept in this document supports a wide range of diverse coverage types. One suitable criterion for classification is along the type of feature bundled in a coverage. As feature types are commonly sorted along their topological dimensions (thereby defining points, lines, surfaces and solids in natural space) this is an appropriate criterion for classifying coverage types. Additionally, there can be further constraints, such as grid coverages where the direct positions are supposed to sit on a grid. In [Clause 6](#) onwards, coverage types sorted along their topological dimension are as defined in ISO 19107.

5.5 Range of a coverage

At every direct position a coverage holds a single value or a non-empty set of values. All of these values of a coverage together make up the coverage's range.

EXAMPLE 1 A coverage can assign to each direct position in a county the temperature, pressure, humidity and wind speed and direction vector, at a specific time, at that point. The coverage maps every direct position in the county to a record of these fields. In this case, the common type of the coverage range values is a record of components, each of its individual type.

The range type component describes the type of the set of values available for the range. Structures such described can be atomic or a record of named components.

Requirement 6: <https://standards.iso211.org/19123/-1/1/req/core/record>

The range type component shall be of type Record Type as defined in ISO 19103.

Requirement 7: <https://standards.iso211.org/19123/-1/1/req/core/mininfo>

The range type of a coverage shall provide at least the information necessary to decode and process the range values:

- a) data type information in some available typing system,
- b) null values, and
- c) unit of measure.

EXAMPLE 2 RGB images, when modelled as a coverage, have as their range type a record consisting of three components red, green and blue (in that order), each of them of type unsigned 8-bit integer.

The structure of a range type definition is not specified further in this document because no commonly accepted consensus exists. However, for interoperability reasons it is highly recommended that some description be adopted, for example to differentiate height in metres from height in feet in a digital elevation model.

Recommendation 1: <https://standards.iso211.org/19123/-1/1/req/core/measurement>

Range type definitions should rely on some commonly agreed system of description and measurement.

NOTE One way of providing this information is via a link pointing to a location that provides this information in a standardized manner. Another way is to provide this information directly as part of the coverage.

5.6 Interpolation

5.6.1 Concept

Sensors observing natural phenomena (mathematically: fields) do not collect data only at the exact point of a direct position. Rather, they integrate data over a given neighbourhood of the direct position and present the result as the value at this position. Generally speaking, observations of the field are aggregated over a given region and time. Therefore, it can make sense to invert this quantization to discrete points and derive values for in-between positions from the direct positions available in the coverage.

Through interpolation, range values can be obtained for coordinates within the domain of a coverage which are not direct positions. Basically, a coverage provides values only at its direct positions. Interpolation means applying a given algorithm to obtain a range value for locations inside a coverage's domain which are not direct positions, usually by combining the values of several direct positions in the neighbourhood of the coordinate location under inspection.

Interpolation requires the domain to be in a CRS which allows for expressing "in-between" coordinates.

EXAMPLE 1 Index coordinates, representing integer numbers, do not allow the expression of any in-between value as this would not be an integer. Conversely, Latitude, Longitude, height and time, allow the expression of values between any two given coordinates as they conceptually map to real numbers. The in-between values can be continuous (such as temperature) or discrete (such as land classification).

Depending on the interpolation method (the algorithm applied), the range type can require provision of "in between" values, too. In general, for interpolation it can be necessary for the range type to allow for arbitrary values to be generated, requiring a continuous range type. Therefore, applicability of a particular interpolation method in general also depends on the range type.

EXAMPLE 2 "Nearest neighbour" interpolation does not require new values to be generated as the resulting value is chosen from the available ones taken from the surrounding direct positions. "Linear interpolation", on the other hand, will usually generate values different from the surrounding ones, based on the arithmetic means computation.

ISO 19107 defines a set of interpolation methods, individually for the particular geometry types, upon which this document relies.

Requirement 8: <https://standards.iso/2011.org/19123/-1/1/req/core/iso19107-interpolation>

Interpolation in a coverage shall conform with the interpolation schemes specified in ISO 19107. A reserved value shall indicate that no interpolation is applicable.

NOTE 1 Typical examples of such values include "none" or "undefined" or "nil". However, these are not defined normatively in this document (and neither are they defined in ISO 19107). Instead, they are left to concrete instantiations of coverage data structures.

NOTE 2 Additionally, an exemplary, non-normative set of interpolation methods is presented in [Annex B](#).

Interpolation can be applied along a single axis or along several axes simultaneously. Therefore, interpolation methods on a high, conceptual level are defined on axis level. An implementation standard based on this document can potentially bundle interpolation methods to allow only one and the same interpolation method across all (or a subset of) axes.

EXAMPLE 3 In horizontal geographic CRSs, interpolation will normally be coupled so that Latitude and Longitude always share the same interpolation. Time and height, on the other hand, will normally be interpolated individually.

Recommendation 2: <https://standards.iso/211.org/19123/-1/1/req/core/interpolation-method>

An application should apply one interpolation method collectively along all axes of the coverage, or it may allow the indication of individual interpolation methods applied to different axes of the coverage.

EXAMPLE 4 In an image timeseries having Latitude, Longitude and time axes, several ways of interpolation are relevant:

- linear interpolation along Latitude and Longitude (“bilinear interpolation”), with temporal resolution unchanged (i.e. all existing timeslices get extracted) and, hence, no interpolation occurring along time;
- interpolation of each pixel’s history (i.e. along time) using linear interpolation, without any spatial interpolation;
- linear interpolation along Latitude, Longitude, and time simultaneously (“trilinear interpolation”).

NOTE 3 As this document is a data model and not a processing or service model, use of a particular interpolation method on a given coverage cannot be enforced. Rather, a coverage can provide information on the set of interpolation methods that is to be applied whenever, in the course of performing some general processing, interpolation needs to be performed (such as in rescaling a coverage along one or more axes).

NOTE 4 Interpolation methods can potentially require additional control parameters. These are not considered in this document.

5.6.2 Discrete and continuous coverages

The domain characteristics decide in the first place whether interpolation is applicable. The CRS, which defines the set of possible coordinate values, can possibly allow addressing of coordinate values beyond the direct positions. Hence, coverages are possible which allow interpolation along one axis, but not along another.

EXAMPLE 1 A grid coverage with Latitude, Longitude and an index axis can be interpolated in Latitude and Longitude, but not along the index axis.

An axis is called discrete if every possible interval with finite bounds describes a finite set of values. Otherwise such an axis is called continuous.

A coverage is called discrete if its axis list contains only discrete axes. A coverage is called continuous if its axis list contains at least one continuous axis.

Coverages can be discrete (or continuous) in their range, in their domain, or in both.

EXAMPLE 2 A map of postal code zones is a coverage which is discrete in its range. The postal code zones cover an entire country and at every location in the country one can evaluate the coverage function and get a value that represents the postal code for that location. Within a postal code zone, the value is constant. Such a discrete coverage cannot be interpolated.

EXAMPLE 3 A coverage that maps a set of polygons to the soil type found within each polygon is a coverage which is discrete in its range. More examples of this type of coverage are given in ISO 19144-2.

EXAMPLE 4 A point set representing a set of measurements that are only valid at the position of each point, and which cannot be interpolated, is a discrete coverage (discrete in domain).

EXAMPLE 5 An image, sampled by a sensor, can be represented as a grid coverage consisting of a set of pixels corresponding to grid cells in the domain of the coverage. A value is associated with each grid cell. However, since the coverage is continuous, an interpolation function (such as linear, quadratic or cubic) can be applied so that a continuously variable value can be determined at any location within the domain extent of the image.

EXAMPLE 6 In a coverage that maps direct positions in San Diego County to their temperature at noon on a specific day, both domain and range can take an infinite number of different values. This continuous coverage would be associated with a discrete coverage that holds the temperature values observed at a set of weather stations (discrete in domain). That is, the measured values correspond to a point set coverage. This point set coverage is discrete because each point can only have one value. The associated continuous coverage uses the point values as driving values for the coverage function and allows interpolation between the points.

EXAMPLE 7 A set of bathymetric soundings is a discrete point set coverage with a single measured water depth value at each point location (discrete in domain). An associated continuous coverage supports interpolation between the measured depth soundings to determine the bottom surface of a body of water.

EXAMPLE 8 Evaluation of a triangulated irregular network involves interpolation of values within a triangle composed of three neighbouring point value pairs.

5.7 Common point rule

The optional Coverage attribute `commonPointRule` of type `CommonPointRule`, carried over from ISO 19123:2005 as deprecated legacy, identifies the procedure to be used for evaluating the Coverage at a position for which more than one range values exist. Its behaviour is defined in [Annex D](#).

NOTE In grid coverages, there is by definition only one value per direct position so the common point rule is irrelevant.

5.8 Realization variants

5.8.1 Overview

In a coverage, domain and range may be organized in different ways, driven by practical considerations. Possible organizations include the following.

- Separate representation of domain and range in some serialization. If both use the same serialization scheme, then corresponding location/value pairs can be identified through their position in the sequence. This is useful, for example, in image processing when the domain is not needed for the image operation on hand.
- Having an implicit description (rather than an explicit representation) of the domain. This reduces a coverage's size and, again, allows convenient range processing while avoiding the sometimes-unwanted overhead of addressing each value one by one through its direct position.
- A set of position/value pairs. This is a natural representation, for example, for point clouds.
- Partitioning a coverage into smaller units. This is commonly known as tiling. In fact, this concept can be extended to recursively nested coverages.
- A functional, analytic description where domain and range are defined through mathematical expressions.
- Combinations of the above, where feasible.

NOTE One interoperable concretization supporting various such organizations is given by ISO 19123-2 which is identical to OGC CIS.^[13]

In the following subclauses, some of the coverage structures listed are detailed further. They are summarized in [Figure 5](#).

5.8.2 Geometry/value pair view

The collection of feature objects constituting a coverage can be seen as establishing a mapping from the features' space to the set of values associated with the features. If these features and their values get enumerated, this naturally leads to a set representation where each set element is a (geometry, value) pair. This is the view supported by the ISO 19107 Collection paradigm. It is captured in CoverageByPartitioning containing a Partition set consisting of geometry/value pairs ([Figure 5](#)). The constraints relevant for this coverage variant are established in [5.8.3](#).

However, other logical organisations of a coverage, with identical information content, can be constructed. Two of them are established below. [Figure 5](#) gives a synoptic view of all variants discussed.

Requirement 9: <https://standards.iso/211.org/19123/-1/1/req/core/gv-pair>

A CoverageByGeometryValuePair shall be structured as described in Figure 5.

5.8.3 Domain/range view

Another way of viewing a coverage is as a mapping from a set of direct positions (given by the geometry objects) to a set of values (given by the feature's associated value payload). In this view, a coverage is defined as a function $C: D \rightarrow R$ with domain D and range R which delivers some value for each element from D . This variant is realized in subclass CoverageByDomainAndRange (Figure 5).

Requirement 10: <https://standards.iso/211.org/19123/-1/1/req/core/domain-range>

A CoverageByDomainAndRange shall be structured as described in Figure 5.

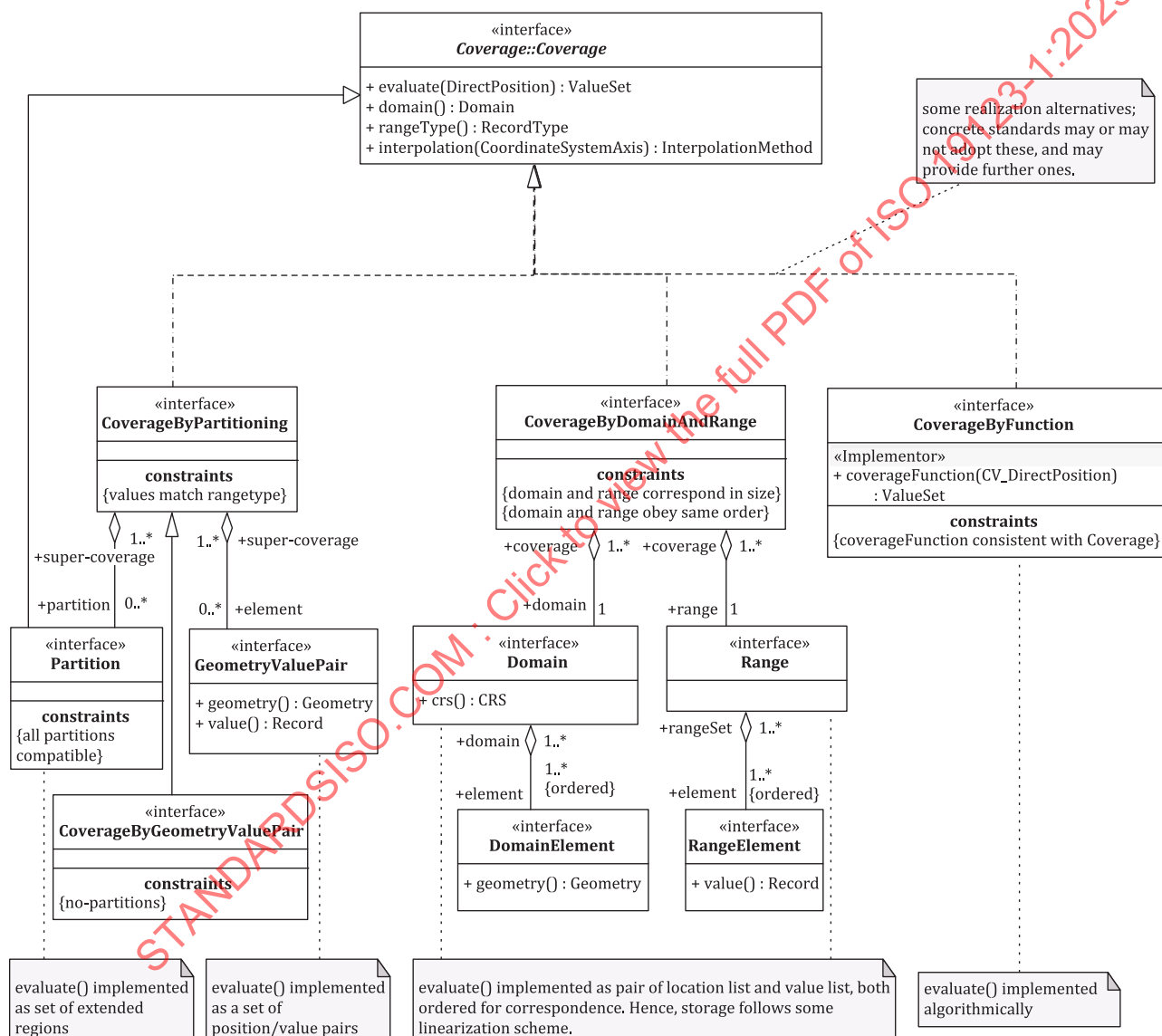


Figure 5 — Class coverage with multiple variants

NOTE Conceptually, the domain of a coverage consists of all direct positions defined for this coverage. This set can be enumerated (for example, point sets in a multi-point coverage). Positions can be given implicitly through various schemes like those listed in Annex D (for example, corner points indicating the set of all raster positions in a rectified grid coverage as per ISO 19123:2005). Hence, the domain/range view on a coverage is particularly relevant for grid coverages.

5.8.4 Partitioned view

The previously-introduced modelling variants resemble two extreme ends of a modelling continuum. In the geometry/value pair representation, the finest possible granularity is adopted by having each single feature object with its value represented together. Conversely, in the domain/range variant, the coarsest possible granularity is adopted by having one set of positions which get mapped to one set of values. In the partitioning approach, organizations in between these extremes are gathered. To this end, a coverage gets split into sub-coverages forming partitions of the original coverage. In the coverage model, this is captured by a PartitionSet containing Partitions.

For a coverage set to be aggregated into a larger coverage, some homogeneity constraints are necessary:

Requirement 11: <https://standards.iso211.org/19123/-1/1/req/core/partitioning>

A CoverageByPartitioning shall be structured as described in [Figure 5](#).

Requirement 12: <https://standards.iso211.org/19123/-1/1/req/core/sub-coverages>

All sub-coverages in a partition shall share the same CRS and range type. Their domains shall fulfil all constraints imposed on the particular coverage type of the super-coverage.

Requirement 13: <https://standards.iso211.org/19123/-1/1/req/core/cyclefree-hierarchies>

A coverage shall not recursively contain itself in a partitioning hierarchy.

NOTE 1 Having only one partition realizes the domain/range variant. Having partitions which contain only one feature object each represents the geometry/value pair variant.

NOTE 2 Partitioned storage organization can massively increase performance of access and processing. In technology, such partitioning is also known as tiling and chunking.

5.8.5 Functional view

Mathematics can be used to describe a coverage. These descriptions must support the *evaluate()* function. This document does not explicitly describe any particular mechanism for expressing such functions.

Requirement 14: <https://standards.iso211.org/19123/-1/1/req/core/by-function>

A CoverageByFunction shall be structured as described in [Figure 5](#).

EXAMPLE In Constructive Solid Geometry (CSG) objects are built by recursive composition of analytical primitives (such as box, sphere, cylinder, torus, etc.) through (regularized) set operations.

5.9 Envelope

For practical purposes, quickly determining where approximately direct positions can be expected is convenient. Therefore, class Coverage contains an optional component envelope of type Extent (as per ISO 19115-1) that provides a simplified summary description of the coverage's domain. This envelope consists of an enclosing bounding shape which should be close to the actual coverage extent but does not have to be minimal. Further, the envelope can be expressed in any CRS which can be transformed to and from the coverage's CRS.

EXAMPLE A satellite swath image can contain a bounding box expressed in WGS 84 ([Figure 6](#)).

Recommendation 3: <https://standards.iso211.org/19123/-1/1/req/core/envelope>

Coverages should contain an Envelope.

Requirement 15: <https://standards.iso211.org/19123/-1/1/req/core/envelope-axes>

The Envelope of a Coverage shall contain all or a subset of the axes of the corresponding domain.

EXAMPLE In a 3D x/y/t image timeseries datacube, the extent can represent only the approximated 2D footprint on the Earth surface, for example, in WGS 84, thus ignoring the time axis.

Requirement 16: <https://standards.iso211.org/19123/-1/1/req/core/envelope-tight>

The Envelope of a Coverage shall approximate its domain as closely as possible, for all axes of the domain present in Extent.

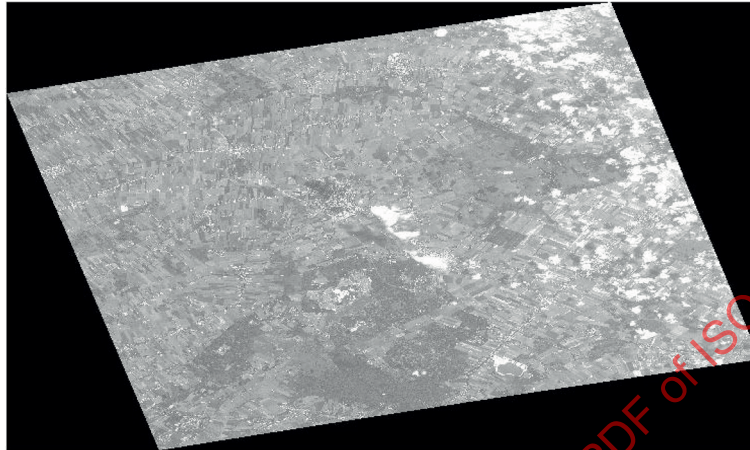


Figure 6 — Satellite image embedded in bounding box

Requirement 17: <https://standards.iso211.org/19123/-1/1/req/core/envelope-domainset>

If a Coverage contains an Envelope then this Envelope shall contain the domain of this Coverage.

Requirement 18: <https://standards.iso211.org/19123/-1/1/req/core/iso19111-crs>

The CRS component of the Coverage shall reference a CRS definition conformant with ISO 19111.

The association coordinate reference system shall link the coverage to the coordinate reference system to which the direct positions in the domain are referenced. Class CRS is specified in ISO 19111.

6 Multi-point coverages

This clause defines the conformance class package MultiPointCoverage.

A multi-point coverage is a coverage consisting of a collection of points. As points may coincide there can be more than one value corresponding to a given direct position. Therefore the evaluation function returns a set of values.

Requirement 19: <https://standards.iso211.org/19123/-1/1/req/multi-point/multi-point>

A multi-point coverage shall contain only geometric objects of type ISO 19107 PointData or a subtype thereof, as described by Figure 7.

Requirement 20: <https://standards.iso211.org/19123/-1/1/req/multi-point/multi-point-evaluate>

Function *evaluate()* shall be defined, for some multi-point coverage *c* and position *p*, as

$evaluate(p) = \{ v \mid \exists \text{ point feature } f \in c: f.contains(p) \}$ where *contains()* is defined in ISO 19107.

NOTE An alternative (and different) realization of MultiPointCoverage is given by the ISO 19107 data type PointCloud. MultiPointCoverage is included here for achieving a complete, coherent coverage framework across all topological and geometric dimensions.

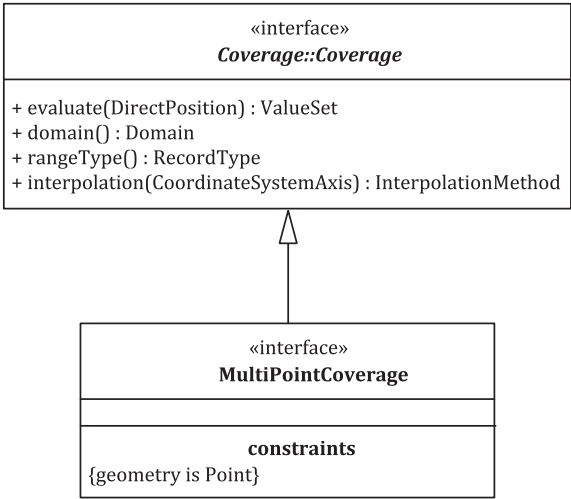


Figure 7 — Class MultiPointCoverage

7 Grid coverages

7.1 Overview

This clause defines the conformance class package GridCoverage.

A grid coverage is a special case of multi-point coverage in that all direct positions are required to sit on a grid. The concept of a multi-dimensional grid is defined in 7.2.

Requirement 21: <https://standards.iso211.org/19123/-1/1/req/grid/grid-subtype-of-point>

A grid coverage shall be defined as in Figure 8.

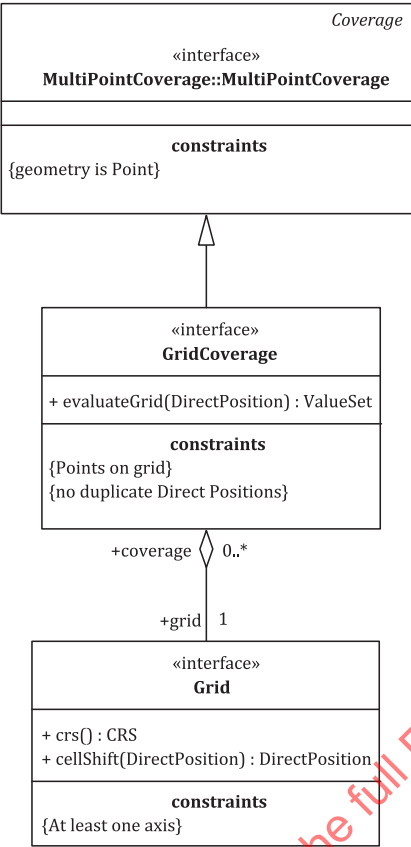


Figure 8 — Class GridCoverage

NOTE Although from an abstract view point grid coverage is a subtype of multi-point coverage, in practice implementation of both types will differ substantially. The regularity of a grid generally supports the lookup of direct positions more easily than in a point cloud. Often, depending on the degree of regularity of the grid, it is not even necessary to materialize the coordinates. This results in efficient storage and processing methods.

7.2 Grids

7.2.1 Grid definition

In a grid coverage, the grid serves to determine the direct positions of the domain by forcing alignment with the specific points given by the grid definition. The grid's CRS is the CRS of the coverage incorporating that grid.

Grids can generally be constructed based on triangles, rectangles or hexagons. In the context of coverages, rectangular grids are modelled through grid coverages, hexagonal grids can be mapped to grid coverages (7.6), and triangular grids are modelled through meshes, i.e. multi-surface (Clause 9) or multi-solid coverages (Clause 10). Therefore, in this document the term "grid" is always understood as a rectangular grid.

NOTE 1 In the nD case grids become n -gons.

NOTE 2 As opposed to a mesh (in Computational Fluid Dynamics also called "unstructured grid") where a vertex (i.e. cell) can be connected with any number of neighbouring vertices, a coverage grid has a regular structure based on a given number of neighbourhood vertices. Therefore, such a grid is sometimes referred to as a "structured grid" for clarity.

Intuitively speaking, in a rectangular grid every direct position (except at the rim of the grid) has exactly one nearest neighbour with a lower coordinate and exactly one nearest neighbour with a higher coordinate along each axis (Figure 9). This neighbourhood establishes the grid topology. the grid

geometry is determined by the concrete coordinate values, which in this document is described by the axis types ([7.2.2](#)).

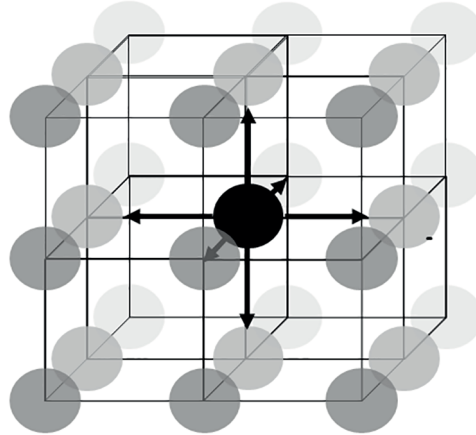


Figure 9 — Multi-dimensional neighbourhood in a grid

Rectangular grids in general do not need to have equidistant spacing between the direct positions. [Figure 11](#) and [Figure 12](#) illustrate some cases of regular and irregular grids.

If the values along each axis are regularly spaced, this induces a regular grid. Otherwise the grid is irregular.

NOTE 3 Often, rectangular grids are illustrated through curve bundles (in case of index, regular and irregular axes: lines), one per axis, whose intersections establish the direct positions. This construction method explains grids in two or more dimensions. However, 1D grids are explicitly covered by this document.

The grid notion can be generalized to the situation that n D grids can be embedded in some $(n+m)$ D space for some $m > 0$.

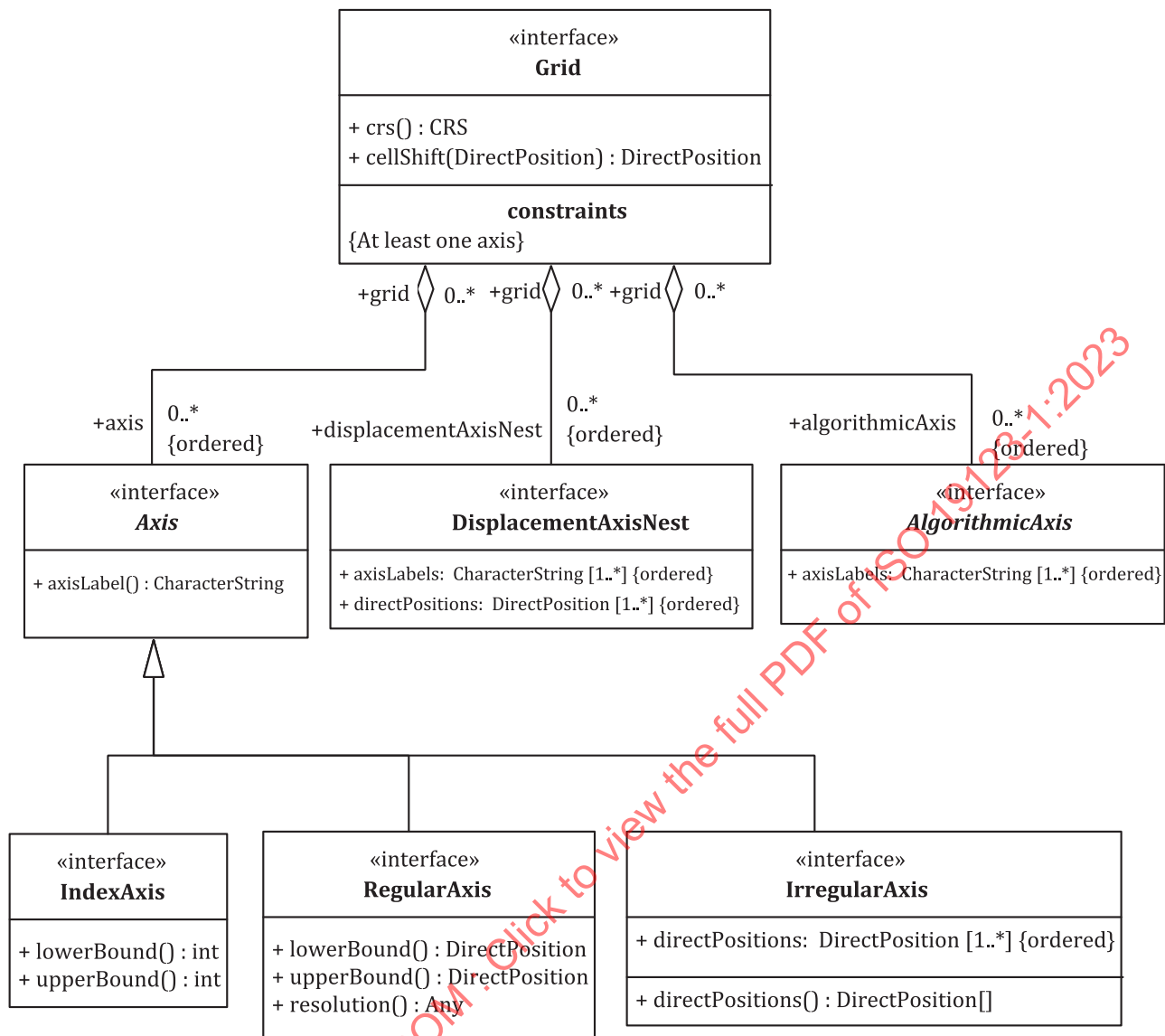


Figure 10 — Class grid

As opposed to a general multi-point coverage, a grid coverage enforces that all grid cells have unique positions. Therefore, the evaluation function will always return exactly one value per direct position.

Requirement 22: <https://standards.iso/2019123-1/1/req/grid/single-point-value>

In a grid coverage, every direct position shall have exactly one value (taken from the coverage's range type) associated, i.e. $| evaluate(p) | = 1$ for all direct positions p .

The direct positions of a grid can be described in several ways, depending on the nature of the grid. This document specifies several alternatives, based on the nature of the coordinate distribution along the grid axes. This is described in [Figure 10](#) and [Subclause 7.2](#).

Requirement 23: <https://standards.iso/2019123-1/1/req/grid/grid-structure>

In a grid coverage, the set of its direct positions shall be given by its grid as described by [Figure 10](#).

7.2.2 Grid axis types

7.2.2.1 General

The axes of a coverage grid can be classified along the degree of regularity of the coordinate distribution along the axis. In this document, the classification is done per axis so that each axis can be characterized individually, allowing combinations such as regular spatial with irregular temporal axes.

Intuitively speaking, in a grid, every direct position (except at the rim) has exactly one nearest neighbour with a lower coordinate and exactly one nearest neighbour with a higher coordinate, in all dimensions. This neighbourhood establishes the topology. The geometry is determined by the concrete coordinate values, which in this document are described by the axis types.

- In the simplest case the direct position coordinates are given by integers, such as in a bare array. This is referred to as “index axis”.
- Often, georeferenced data (such as ortho images) have their coordinates distributed equally along an axis. Such a “regular axis” can be described by the single common distance between adjacent points on the axis, commonly called “resolution”.
- More generally, the direct position coordinates along the axis can be distributed irregularly, leading to the concept of an “irregular axis”.
- Even more generally, each direct position can have coordinates not aligned with any other direct position, leading to a “warped nest” (while still respecting the topological neighbourhood property illustrated in [Figure 9](#)).
- In the extreme case, coordinates are not available directly at all, but rather determined from some opaque data structure fed into some algorithm. Therefore, this is called an “algorithmic axis”. [Figure 11](#) through [Figure 15](#) illustrate several of these cases.

NOTE This axis classification establishes several ways to describe the coordinates of the direct positions, not the grid CRS which contains the axis definitions.

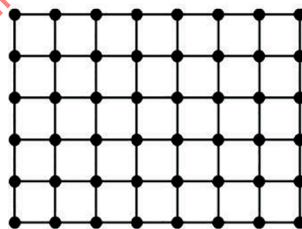


Figure 11 — Sample regular 2D grid

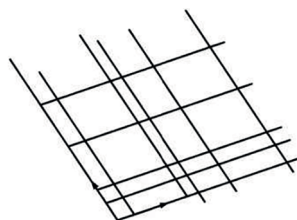


Figure 12 — Sample 2D irregular grid

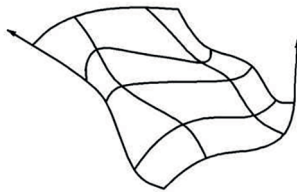


Figure 13 — Sample 2D warped nest grid

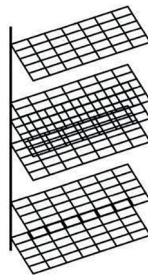


Figure 14 — Sample 3D grid representing the combination of regular Lat/Long with irregular time



Figure 15 — Sample 3D grid representing the combination of a warped nest with irregular time

7.2.2.2 Index, Regular, and Irregular Axes

An index axis is a 1D unit-less axis (in ISO 19111:2019 named “Cartesian axes”). There is no georeference, and admissible coordinates are at discrete, integer positions.

Requirement 24: <https://standards.iso/211.org/19123/-1/1/req/grid/index-axis>

In class Grid, an index axis shall be given by an axis identifier as defined in the grid CRS, lower and upper bounds lo and hi with $lo, hi \in \mathbb{Z}$ and $lo \leq hi$. Direct positions shall be defined for every coordinate tuple where the coordinate value of the index axis on hand is from the closed interval $S = \{x \in \mathbb{Z} \mid lo \leq x \leq hi\}$.

NOTE 1 The unit of measure of an index axis is 1, i.e. coordinate values are unitless.

NOTE 2 A standard concretizing this abstract coverage standard can define further unitless axes using real-valued coordinates, in addition to the integer-based index axis.

NOTE 3 Quadrilateral grids can be represented efficiently as arrays in programming languages, which leads to a preferred storage technique for coverages where the range is modelled as an array based on certain implementation-dependent linearization scheme (such as row-major or column-major arrangement). See [Annex C](#) for sample sequencing rules. This allows, for example, image processing tools to ignore the real-world coordinates and operate efficiently on the array.

A regular axis has an equidistant spacing like an index axis, but is continuous and not constrained to integer positions and distances. It can be georeferenced, i.e. it can have spatial or temporal semantics attached.

Requirement 25: <https://standards.iso211.org/19123/-1/1/req/grid/regular-axis>

In class Grid, a Regular axis shall be given by the axis identifier as defined in the grid CRS, lower and upper bounds lo and hi with $lo, hi \in C$ and $lo \leq hi$, a resolution $r \in C$ where C denotes the coordinate value set defined for this axis in the grid CRS. Direct positions shall be defined for every coordinate tuple where the coordinate value of the Regular axis on hand is from the set $S = \{x \in C \mid lo \leq x * r \leq hi\}$ for some $r \neq 0$.

NOTE 4 Further relevant information of an axis, such as a unit of measure as well as datum are defined in the CRS.

The next level of generalization is an irregular axis. An irregular axis is continuous, possibly georeferenced, and its distances are irregular (but non-zero). Such an axis can be georeferenced, i.e. it can have a spatial or temporal semantics attached.

Requirement 26: <https://standards.iso211.org/19123/-1/1/req/grid/irregular-axis>

In class Grid, an Irregular axis shall be given by an axis identifier as defined in the grid CRS and a set of positions $P = \{p_1, \dots, p_n\} \subseteq C$ where C denotes the coordinate value set defined for this axis in the grid CRS. Direct positions shall be defined for every coordinate tuple where the coordinate value of the Irregular axis on hand is from P .

Mathematically, an n -dimensional grid as discussed in this subclause is a tessellation of the grid coverage's domain defining a set of direct positions through geometric rules as follows. For some $n > 0$ let $A = (a_1, \dots, a_n)$ be a finite ordered set of axes where each axis $a_i = \{v_{i,1}, \dots, v_{i,m_i}\}$ is a totally ordered set of $m_i > 0$ values. This induces a grid $G = a_1 \times \dots \times a_n$ as the cross product. G can be interpreted as a set of coordinates yielding direct positions, $G = \{(x_1, \dots, x_n) \mid x_i \in a_i \text{ for } 1 \leq i \leq n\}$.

7.2.2.3 Displacement axis nest

A displacement axis nest (or warped nest) is a set of continuous, possibly georeferenced axes forming a subset of the CRS's axes. Relative to a regular grid, each direct position is shifted by some individual offset within the CRS space spanned by the axes participating.

Requirement 27: <https://standards.iso211.org/19123/-1/1/req/grid/displacement-axis-nest>

In class Grid, a displacement axis nest shall be given by a list of axis identifiers as defined in the grid CRS forming a non-empty subset of the axes contained in the grid CRS. Direct positions shall be defined for every coordinate tuple where the coordinate value of each axis participating in the displacement axis nest on hand is in the coordinate value set of this axis.

7.2.2.4 Algorithmic axis

In the most general case, coordinates of the direct positions are not stored explicitly, but obtained algorithmically from certain implementation-dependent parameters. Such algorithmic axes are given by a set of discrete or continuous, possibly georeferenced axes, forming a subset of the CRS's axes where the direct positions have to be derived algorithmically from some otherwise abstract parameters (hence, the alternative name "transformation model").

Requirement 28: <https://standards.iso211.org/19123/-1/1/req/grid/algorithmic-axis>

In class Grid, an Algorithmic axis set shall be given by a list of $n > 0$ axis identifiers as defined in the grid CRS and a parameter set P . Direct positions shall be given through some algorithm parametrized with P resulting in coordinate values consistent with the definition of the axes involved.

NOTE The structure and meaning of P is not specified further in this document.

EXAMPLE A satellite image timeseries can perhaps define geographic pixel coordinates through ground control points acting as input to the orthorectification pipeline, whereas the time axis can perhaps be regular or irregular.

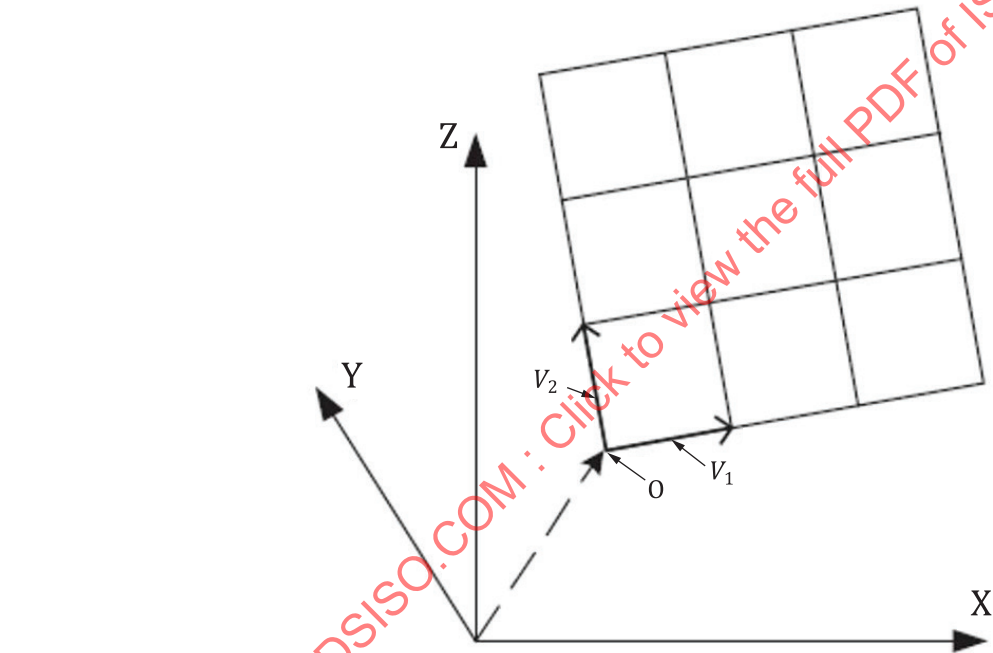
7.2.2.5 Combinations

By combining all the above axis types freely, any type of grid can be modelled. The list of possible axis types is not exhaustive. Some standards or applications can define their own additional axis types. [Figure 11](#) through [Figure 14](#) show some sample grid types combining different axis types defined in this document.

7.3 Rectified and referenceable grid coverages

Superseded ISO 19123:2005 makes use of the terms “rectified grid coverage” and “referenceable grid coverage”. These terms can be defined as special cases of the concepts introduced above.

EXAMPLE [Figure 16](#) shows a 2D grid in the 3-space determined by the axes X, Y, and Z. The grid origin is at “O”. There are two offset vectors labelled V_1 and V_2 which specify the orientation of the grid axes and the spacing between the grid lines. The coordinates of the grid points are of the form: $O + aV_1 + bV_2$.



Key	
X, Y, Z	axes to determine 3-space
V_1, V_2	offset vectors
O	grid origin

Figure 16 — Geometry of a rectified grid

Rectified grids can be mapped to a Cartesian grid through an affine transformation. For referenceable grids, this transformation does not have to be affine. Therefore, the term (geo-)referenceable grid encompasses all (geo-)referenced cases beyond (geo-)rectified grids. In the terminology of this document this means the grid can contain irregular, warped or algorithmic axes.

In summary, therefore, the terms “rectified grid coverage” and “referenceable grid coverage”, which have been used in earlier coverage standard versions but are now obsolete, can be described through the conceptualization used in this document as follows ([Figure 17](#)):

- a rectified grid coverage is a grid coverage where every axis is either an index axis or a regular axis;

- a referenceable grid coverage is a grid coverage where at least one axis is neither an index axis nor a regular axis.

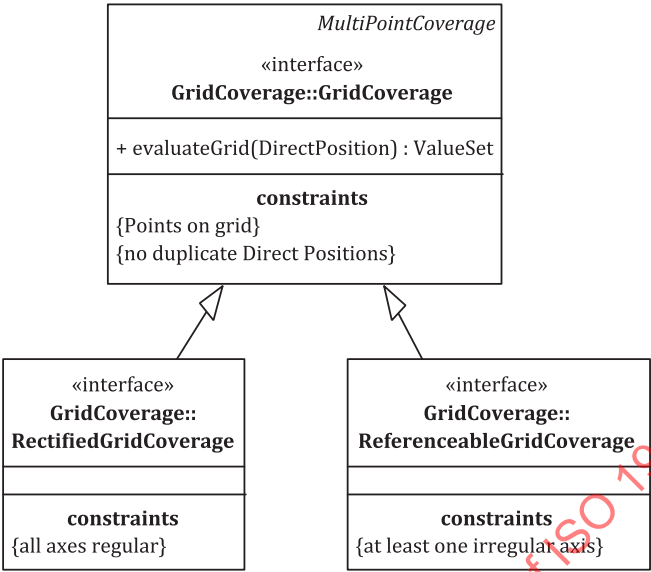


Figure 17 — Rectified and referenceable grid coverages vs. general grid coverage

7.4 Grid cells

7.4.1 Grid cell concept

Common practice is to consider the information stored at a direct position (such as radiance energy) not concentrated in the zero-extent point, but distributed over some area around this direct position.

EXAMPLE In charge-coupled device (CCD) sensor arrays, each individual sensor collects photons on some finite surface, hence the electrical charge delivered is representative not only for the direct position, but it also represents an aggregated value for the whole area seen by the CCD sensor.

This concept is captured by the notion of a “grid cell” which, in its full generality, is given by a neighbourhood around a direct position. Cells are constrained in that they do not overlap, while “empty” space not covered by any cell may exist within a coverage’s domain extent. Cells may have different shapes, including a circular shape around the direct position, a rectangle where the direct position sits in the geometric centre or in some corner of the cell.

One way to define cells is based on the grid shape. Taking the (imaginary) grid lines yields minimal boxes between adjacent direct positions (Figure 18 through Figure 20). In case of regular grids with cell shapes defined as the rectangles between neighbouring direct positions, all of the grid cells share the same shape and size. Otherwise, the grid cells in general are not equal in size and shape.

Optional function `cellShift()` in class `grid` allows specifying a particular position of a cell relative to some input position. If no such function is indicated, then a 0 offset is applied.

Requirement 29: <https://standards.iso/211.org/19123/-1/1/req/grid/cell-shift>

In class `Grid`, function `cellShift()` shall indicate the relative cell offset against the original direct position of the cell, with default being no offset (i.e. a 0 vector).

NOTE 1 This function can be realized in various ways, including a code list for common offsets such as pixel-in-centre in regular grids.

NOTE 2 The concept of grid cells is related to interpolation.

7.4.2 Pixel-in-centre, pixel-in-corner

Intuitively, cells by default are assumed to extend around their direct position as the geometric centre (Figure 18). If cells are positioned in a way that the direct positions corresponding to a cell sit in its centre, then this is commonly referred to as “pixel in centre”.

If cells are positioned such that the corresponding direct positions sit in the “upper-left”, “lower-left”, etc. of the cell’s corners, then this is called “pixel in corner”. Figure 18 through Figure 20 show both situations for a regular 2D grid.

In case of a regular grid, all cell positions can be obtained from a natural “pixel in centre” case by applying a translation by half the grid distance along each axis affected. Technically, this can be expressed as a derived CRS in which the deriving conversion is (in mathematical terms) an affine transformation. The skewed grid retains the datum of its base CRS. Irregular grids require special treatment, up to keeping individual offsets for each direct position.

EXAMPLE In the Atmospheric science community the Arakawa grid system is used for grid offset handling.

NOTE 1 The choice between pixel-in-corner and pixel-in-centre potentially has dramatic effects on the data interpretation. For example, a half-pixel geographic offset in a 1° resolution grid can correspond to a shift in the order of around 100 km. A half-pixel offset on a temporal resolution of one year corresponds to six months, like June versus December.

NOTE 2 In ISO 19111 where the image CRS originally resided before incorporation into this document, class ImageDatum had a codelist attribute PixelInCell with possible values cellCenter and cellCorner. The code cellCorner however was ambiguous because an n -dimensional cell has 2^n corners. In this document the cell shift mechanism is generalized to capture all possible situations, and is also now applicable to all kinds of grids, irrespective of the CRS used. By changing dependency references from ISO 19111:2007 to ISO 19111:2019, the new definitions of ISO 19111 are adopted.

7.5 Grid coverage

A grid coverage is a multi-point coverage with the specific restriction that the direct positions are not arbitrary, but given by some underlying grid as defined in 7.2. This holds recursively also for grid coverages built from partitions.

Requirement 30: <https://standards.iso/211.org/19123/-1/1/req/grid/sub-coverage>

The sub-coverage partitions contained in a given grid coverage shall, in their entirety, satisfy all requirements established in 7.2 and Figure 10.

Requirement 31: <https://standards.iso/211.org/19123/-1/1/req/grid/grid-evaluate>

Function *evaluate()* shall be defined, for some grid coverage c and position p , as

$evaluate(p) = v$ for the corresponding point feature $f \in c$ with $f.contains(p)$, where *contains()* is from ISO 19107.

NOTE By construction, direct positions in a grid are pairwise disjoint. Hence, there can be only one associated value for any given direct position.

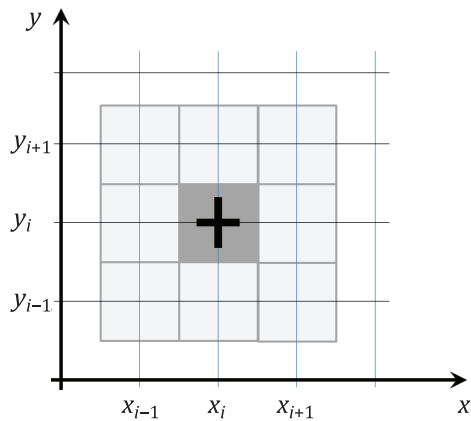


Figure 18 — Grid cell with “pixel-in-centre”

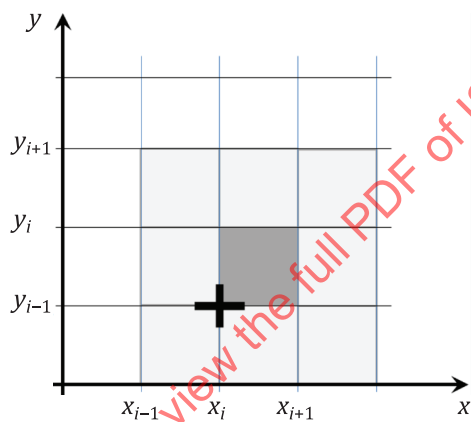


Figure 19 — Grid cell with “pixel-in-corner” sitting “lower-left” with direct position (x_i, y_i) and corresponding grid cell marked with upward y axis

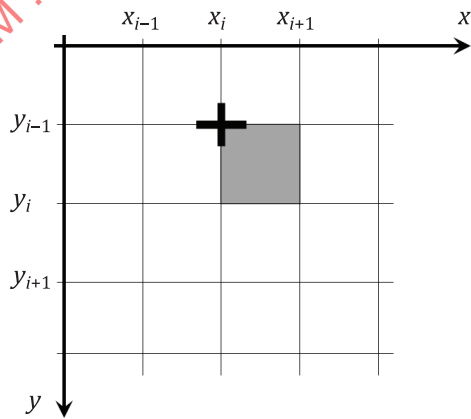


Figure 20 — Grid cell with “pixel-in-corner” sitting “lower-left” with direct position (x_i, y_i) and corresponding grid cell marked with downward y axis

7.6 Further grid coverage types

Aside from the quadrilateral tiling used in the grid coverages discussed above, there are two more regular tilings in the 2D Euclidean plane: triangular and hexagonal. More generally, n -dimensional tiling is addressed by the theory of n -honeycombs.

One example is constituted by hexagonal grid coverages which are given by tessellations composed of regular hexagons. Such tessellations are usually called hexagonal grids. The centres of a set of regular hexagons that form such a tessellation correspond to the grid points of a quadrilateral grid (Figure 21). That grid can be described as a rectified grid in which the two offset vectors are of equal length but differ in direction by 60°. The length of a side of the hexagon is $L = S \tan 30^\circ$, where S is the length of the offset vector. This means that the values in the coverage range can be stored in a computer as a multi-dimensional array. The hexagons are the Thiessen polygons that are generated around the grid points.

NOTE 1 A set of Thiessen polygons generated from the grid points of any 2D rectified grid described by two offset vectors that are equal in length but not orthogonal will be a set of congruent hexagons. The hexagons will be irregular, and, hence, out of scope, unless the offset vectors differ in direction by exactly 60°.



Figure 21 — Sample hexagonal grid

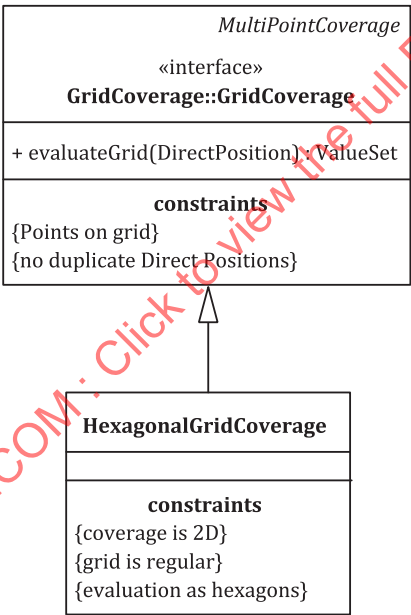


Figure 22 — Hexagonal grid coverage

A hexagonal grid coverage (Figure 22) evaluates a coverage at direct positions within a network of hexagons centred on a set of grid points. Evaluation is based on interpolation between the centres of the value hexagons surrounding the input position.

8 Multi-curve coverages

8.1 Overview

This clause defines the conformance class package MultiCurveCoverage.

8.2 General multi-curve coverages

A multi-curve coverage is a coverage consisting of a collection of curves.

EXAMPLE A coverage that assigns a route number, a name, a pavement width and a pavement material type to each segment of a road network can be represented as a multi-curve coverage.

Requirement 32: <https://standards.isotc211.org/19123/-1/1/req/multi-curve/multi-curve>

A multi-curve coverage shall contain only geometric objects of type ISO 19107 CurveData or a subtype thereof, as described by Figure 23.

Requirement 33: <https://standards.isotc211.org/19123/-1/1/req/multi-curve/multi-curve-evaluate>

Function *evaluate()* shall be defined, for some multi-curve coverage *c* and position *p*, as:

$evaluate(p) = \{ v \mid \exists \text{ curve feature } f \in c : f.contains(p) \}$ where *contains()* is defined in ISO 19107.

NOTE For the avoidance of doubts, a multi-curve coverage can contain additional dimensions (such as time).

8.3 Segmented curve coverages

Segmented curve coverages are used to model phenomena that vary continuously or discontinuously along curves, which may be elements of a network. The domain of a segmented curve coverage is described by a set of curves and includes all the direct positions in all of the curves in the set (Figure 24).

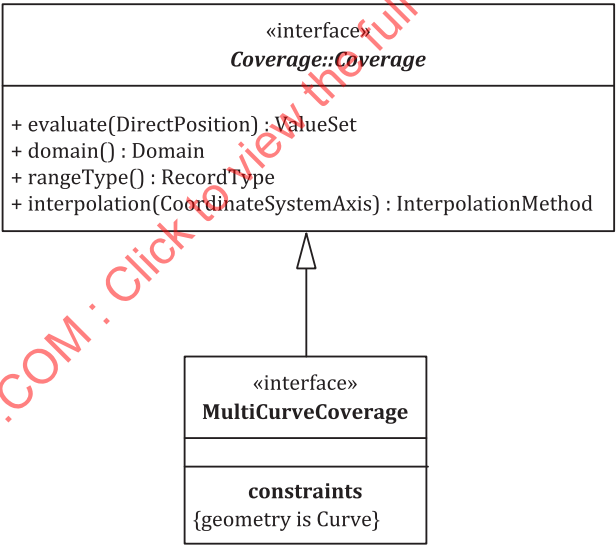


Figure 23 — Class MultiCurveCoverage

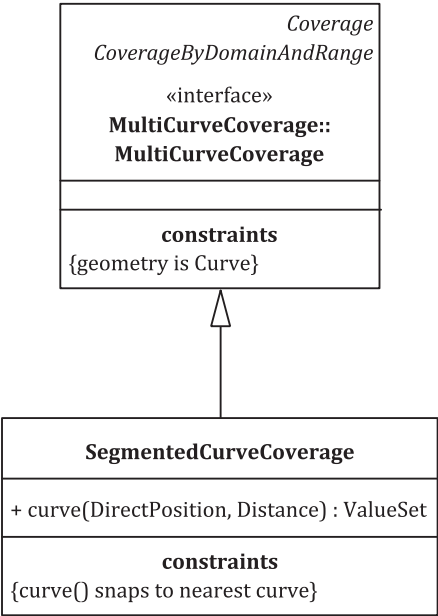


Figure 24 — Class SegmentedCurveCoverage

9 Multi-surface coverages

9.1 Overview

This clause defines the conformance class package MultiSurfaceCoverage.

9.2 General multi-surface coverages

A MultiSurfaceCoverage is a coverage consisting of a collection of surfaces.

EXAMPLE A coverage that represents soil types typically has a spatial domain composed of surfaces with irregular boundaries.

Requirement 34: <https://standards.iso211.org/19123/-1/1/req/multi-surface/multi-surface>

A MultiSurfaceCoverage shall contain only geometric objects of type ISO 19107 SurfaceData or a subtype thereof, as described by Figure 25.

Requirement 35: <https://standards.iso211.org/19123/-1/1/req/multi-surface/multi-surface-evaluate>

Function evaluate() shall be defined, for some multi-surface coverage c and position p, as $evaluate(p) = \{ v \mid \exists \text{ surface feature } f \in c: f.contains(p) \}$ where contains() is defined in ISO 19107.

There are various practically relevant subtypes of multi-surface coverages, including polyhedral surfaces and their special case of Triangulated Irregular Networks (TINs). While in the previous edition of ISO 19123 TIN coverages were modelled separately, they can now be obtained through subtyping of ISO 19107 Surface.

NOTE For the avoidance of doubt, a MultiSurfaceCoverage can contain additional dimensions (such as time).

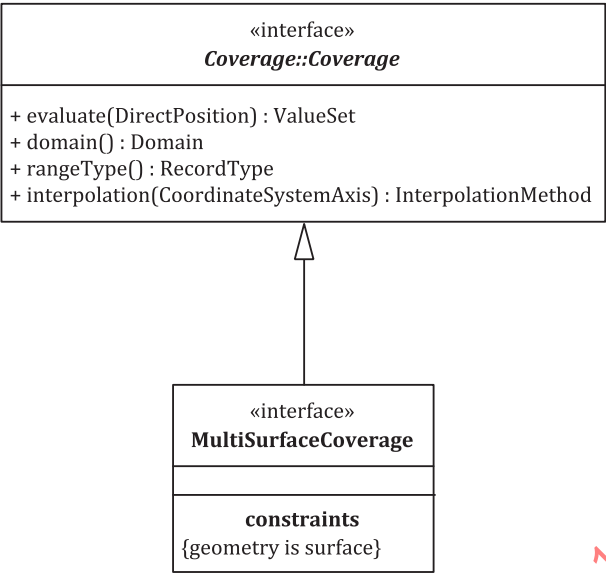


Figure 25 — Class MultiSurfaceCoverage

9.3 Further surface coverages

9.3.1 General

So far surface coverages have been considered which represent a bundle of surfaces not constrained further. Some applications consider surfaces establishing tessellations.

9.3.2 Thiessen polygon coverages

A Thiessen polygon network (see [Figure 26](#) and [Figure 27](#) for example) is a tessellation of a 2D space into surfaces bounded by Thiessen polygons. A finite collection of points on a plane determines a partition of the plane into a collection of polygons equal in number to the collection of points. A Thiessen polygon is generated from one of a defining set of points by forming the set of direct positions that are closer to that point than to any other point in the defining set. The specific point is called the centre of the resulting polygon. The boundaries between neighbouring polygons are the perpendicular bisectors of the lines between their respective centres. Each polygon shares each of its edges with exactly one other polygon. Each polygon contains exactly one point from the defining set. Thiessen polygons are also known as Voronoi Diagrams or Proximal Sets.

EXAMPLE [Figure 27](#) shows a collection of points with their (x, y) coordinates, the perpendicular of the lines that would be drawn between them, and the resultant polygons.

Evaluation of a Thiessen polygon coverage involves two steps. The first is to find the Thiessen polygon that contains the input direct position; the second is to interpolate the feature attribute values at the direct position from the geometry/value pairs at the centres of the surrounding Thiessen polygons.

Technically, a Thiessen polygon coverage is a specialization of MultiSurfaceCoverage as shown in [Figure 25](#).

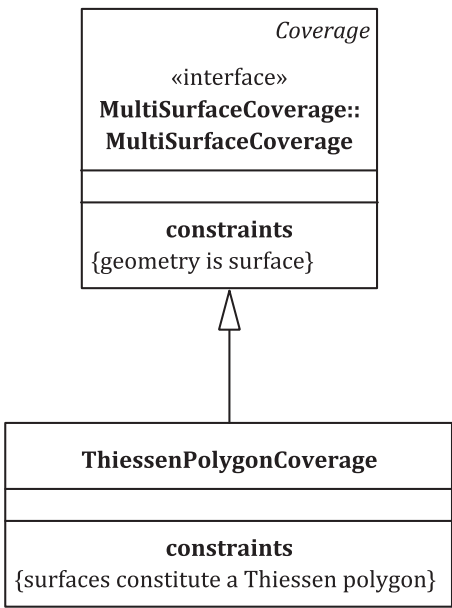


Figure 26 — Class ThiessenPolygonCoverage

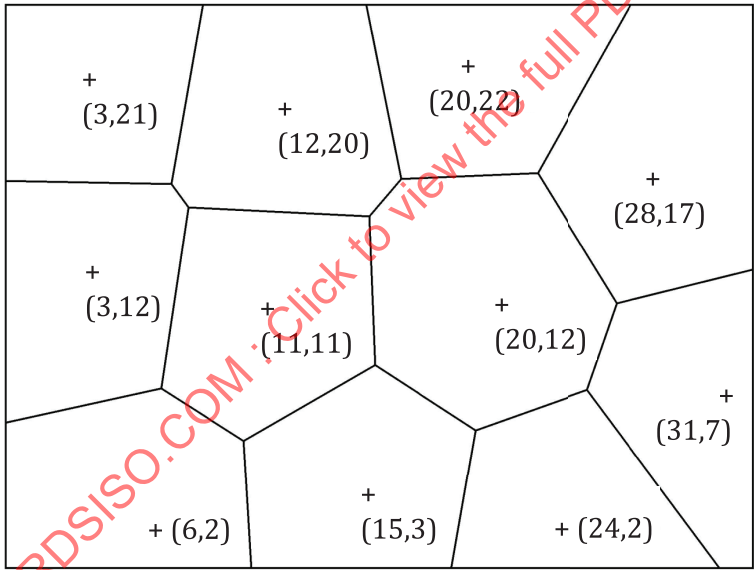


Figure 27 — Sample Thiessen polygon coverage

9.3.3 Triangulated irregular networks (TINs)

Class TinCoverage is shown in [Figure 28](#). The basic idea of a TIN is to partition the points in the spatio-temporal domain of a discrete point coverage into a computationally unique set of non-overlapping triangles. Each triangle is formed by three of the points in the spatio-temporal domain of the discrete point coverage ([Figure 29](#)). The Delaunay triangulation method is commonly used to produce TIN tessellations with triangles that are optimally equiangular in shape, and are generated in such a manner that the circumscribing circle containing each triangle contains no point of the discrete point coverage other than those at the vertices of the triangle.

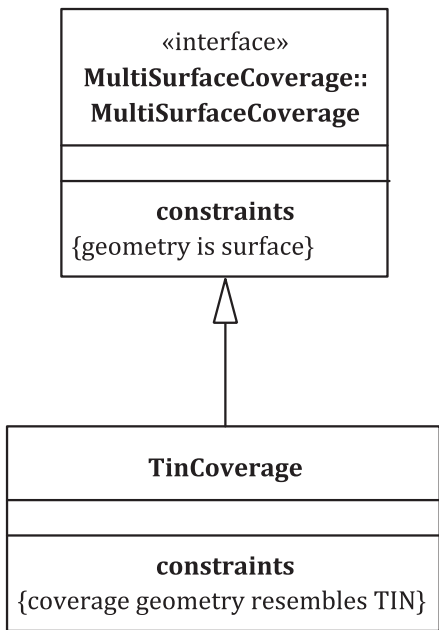


Figure 28 — Class TinCoverage

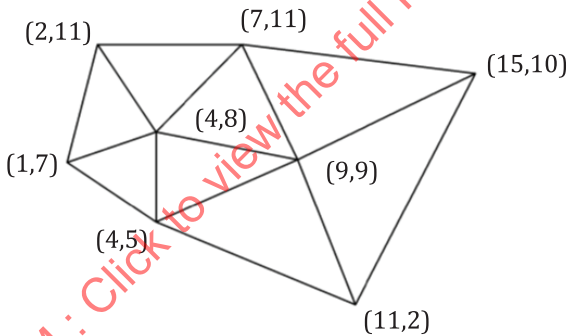


Figure 29 — Sample TIN coverage

10 Multi-solid coverages

This clause defines the conformance class package MultiSolidCoverage.

A multi-solid coverage is a coverage consisting of a collection of solids.

Requirement 36: <https://standards.iso211.org/19123/-1/1/req/multi-solid/multi-solid>

A multi-solid coverage shall contain only geometric objects of type ISO 19107 SolidData or a subtype thereof, as described by [Figure 30](#).

Requirement 37: <https://standards.iso211.org/19123/-1/1/req/multi-solid/multi-solid-evaluate>

Function *evaluate()* shall be defined, for some multi-solid coverage *c* and position *p*, as

$evaluate(p) = \{ v \mid \exists \text{ solid feature } f \in c: f.contains(p) \}$ where *contains()* is defined in ISO 19107.

NOTE For the avoidance of doubt, a multi-solid coverage can contain additional dimensions (such as time).

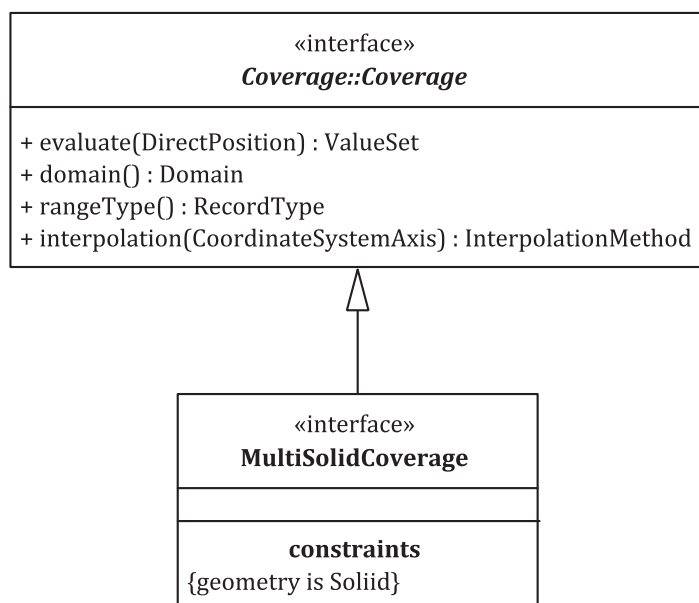


Figure 30 — Class MultiSolidCoverage

Annex A (normative)

Conformance tests

A.1 Conformance classes

This document defines six conformance classes: Coverage Core (specification target: Coverage), multi-point coverage (specification target: MultiPointCoverage), grid coverage (specification target: GridCoverage), multi-curve coverage (specification target: MultiCurveCoverage), multi-surface coverage (specification target: MultiSurfaceCoverage), and multi-solid coverage (specification target: MultiSolidCoverage).

Standardization targets are specifications containing provisions for coverages. A specification claiming conformance to this document shall implement the conformance class relevant to that specification target.

Conformance with this document shall be assessed using all the relevant conformance test cases specified in this annex.

The common base URI for the conformance tests in this document is
<https://standards.iso211.org/19123/-1/1/conf/>.

A.2 Conformance class Coverage Core

Conformance test	https://standards.iso211.org/19123/-1/1/conf/core/allRequirements
Reference	All normative statements in requirements class: <i>Coverage Core</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	<p>Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:</p> <ul style="list-style-type: none"> — Requirement 1: https://standards.iso211.org/19123/-1/1/req/core/packages — Requirement 2: https://standards.iso211.org/19123/-1/1/req/core/model — Requirement 3: https://standards.iso211.org/19123/-1/1/req/core/coordinates — Requirement 4: https://standards.iso211.org/19123/-1/1/req/core/directpositions — Requirement 5: https://standards.iso211.org/19123/-1/1/req/core/crs — Requirement 6: https://standards.iso211.org/19123/-1/1/req/core/record — Requirement 7: https://standards.iso211.org/19123/-1/1/req/core/mininfo — Requirement 8: https://standards.iso211.org/19123/-1/1/req/core/ iso19107-interpolation — Requirement 9: https://standards.iso211.org/19123/-1/1/req/core/gv-pair — Requirement 10: https://standards.iso211.org/19123/-1/1/req/core/domain-range — Requirement 11: https://standards.iso211.org/19123/-1/1/req/core/partitioning — Requirement 12: https://standards.iso211.org/19123/-1/1/req/core/sub-coverages

Conformance test	https://standards.iso211.org/19123/-1/1/conf/core/allRequirements
	<ul style="list-style-type: none"> — Requirement 13: https://standards.iso211.org/19123/-1/1/req/core/cyclefree-hierarchies — Requirement 14: https://standards.iso211.org/19123/-1/1/req/core/by-function — Requirement 15: https://standards.iso211.org/19123/-1/1/req/core/envelope-axes — Requirement 16: https://standards.iso211.org/19123/-1/1/req/core/envelope-tight — Requirement 17: https://standards.iso211.org/19123/-1/1/req/core/envelope-domainset — Requirement 18: https://standards.iso211.org/19123/-1/1/req/core/iso19111-crs
Test type:	Basic

A.3 Conformance class Multi-Point Coverage

Conformance test	https://standards.iso211.org/19123/-1/1/conf/multi-point/allRequirements
Reference	All normative statements in requirements class: <i>Multi-Point Coverage</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	<p>Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:</p> <ul style="list-style-type: none"> — Requirement 19: https://standards.iso211.org/19123/-1/1/req/multi-point/multi-point — Requirement 20: https://standards.iso211.org/19123/-1/1/req/multi-point/multi-point-evaluate
Test type:	Basic

A.4 Conformance class Grid Coverage

Conformance test	https://standards.iso211.org/19123/-1/1/conf/grid/allRequirements
Reference	All normative statements in requirements class: <i>Grid Coverage</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	<p>Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:</p> <ul style="list-style-type: none"> Requirement 21: https://standards.iso211.org/19123/-1/1/req/grid/grid-subtype-of-point — Requirement 22: https://standards.iso211.org/19123/-1/1/req/grid/single-point-value — Requirement 23: https://standards.iso211.org/19123/-1/1/req/grid/grid-structure — Requirement 24: https://standards.iso211.org/19123/-1/1/req/grid/index-axis — Requirement 25: https://standards.iso211.org/19123/-1/1/req/grid/regular-axis

Conformance test	https://standards.iso211.org/19123/-1/1/conf/grid/allRequirements
	<ul style="list-style-type: none"> — Requirement 26: https://standards.iso211.org/19123/-1/1/req/grid/irregular-axis — Requirement 27: https://standards.iso211.org/19123/-1/1/req/grid/displacement-axis-nest — Requirement 28: https://standards.iso211.org/19123/-1/1/req/grid/algorithmic-axis — Requirement 29: https://standards.iso211.org/19123/-1/1/req/grid/cell-shift — Requirement 30: https://standards.iso211.org/19123/-1/1/req/grid/sub-coverage — Requirement 31: https://standards.iso211.org/19123/-1/1/req/grid/grid-evaluate
Test type:	Basic

A.5 Conformance class Multi-Curve Coverage

Conformance test	https://standards.iso211.org/19123/-1/1/conf/multi-curve/allRequirements
Reference	All normative statements in requirements class: <i>Multi-Curve Coverage</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	<p>Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:</p> <ul style="list-style-type: none"> — Requirement 32: https://standards.iso211.org/19123/-1/1/req/multi-curve/multi-curve — Requirement 33: https://standards.iso211.org/19123/-1/1/req/multi-curve/multi-curve-evaluate
Test type:	Basic

A.6 Conformance class Multi-Surface Coverage

Conformance test	https://standards.iso211.org/19123/-1/1/conf/multi-surface/allRequirements
Reference	All normative statements in requirements class: <i>Multi-Surface Coverage</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	<p>Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:</p> <ul style="list-style-type: none"> — Requirement 34: https://standards.iso211.org/19123/-1/1/req/multi-surface/multi-surface — Requirement 35: https://standards.iso211.org/19123/-1/1/req/multi-surface/multi-surface-evaluate
Test type:	Basic

A.7 Conformance class Multi-Solid Coverage

Conformance test	https://standards.iso211.org/19123/-1/1/conf/multi-solid/allRequirements
Reference	All normative statements in requirements class: <i>Multi-Solid Coverage</i>
Test purpose:	Verify that the specification under test conforms to all requirements of this conformance class
Test method:	Evaluate the following tests for every requirement in turn. The overall test passes if every single test passes:

Conformance test	https://standards.iso211.org/19123/-1/1/conf/multi-solid/allRequirements
	<ul style="list-style-type: none"> — Requirement 36: https://standards.iso211.org/19123/-1/1/req/multi-solid/multi-solid — Requirement 37: https://standards.iso211.org/19123/-1/1/req/multi-solid/multi-solid-evaluate
Test type:	Basic

STANDARDSISO.COM : Click to view the full PDF of ISO 19123-1:2023

Annex B
(informative)

Interpolation methods

B.1 General

Evaluation of a continuous coverage involves interpolation between known feature attribute values associated with geometric objects in the spatio-temporal domain of the discrete coverage that is provided as control for the continuous coverage. There are several interpolation methods. Each is used in the context of specified geometric configurations ([Table B.1](#)).

The enumerated data type InterpolationMethod includes the following methods: nearest neighbour, linear, quadratic, cubic, lost area and barycentric. These are described in [B.2](#) through [B.7](#). A specific value, none, is reserved to express that no interpolation is applicable on the coverage holding this interpolation setting.

A set of interpolation methods which also apply to coverages is given in ISO 19107. This document defines further interpolation techniques; some of these are identical to interpolations defined in ISO 19107, but they have been maintained in this document in order to introduce them under the name that has been established by earlier versions of this document. Generally, a given application or standard may define additional ways of interpolation.

Since InterpolationMethod is a CodeList ([Figure B.1](#)) it may be extended in an application schema that specifies additional interpolation methods.

«CodeList» InterpolationMethod
+ nearestNeighbour + linear + quadratic + cubic + barycentric + lostArea

Figure B.1 — Interpolation method codelist

Table B.1 — Interpolation methods

Method	Coverage type	Coverage Dimension	Clause
None	Any	Any	B.1
Nearest neighbour	Any	Any	B.2
Linear	Any	Any	B.3
Quadratic	Any	Any	B.4
Cubic	Any	Any	B.5
Lost area	Thiessen polygon, hexagonal grid	2	B.6
Barycentric	TIN	2	B.7

B.2 Nearest neighbour interpolation

Nearest neighbour interpolation can be applied to any coverage. It generates a feature attribute value at a direct position by assigning it the feature attribute value associated with the nearest direct position in the spatio-temporal domain of the coverage. Nearest neighbour interpolation extends a discrete coverage to a step function defined on the convex hull of the domain objects in the domain of the coverage. Nearest neighbour interpolation is the only interpolation method described in this document that can be used to interpolate nominal or ordinal range values.

B.3 Linear interpolation

Linear interpolation serves to compute values between direct positions in any-dimensional domains through a linear polynomial. The mathematical principle is shown for the 1D and 2D cases.

1D linear interpolation is commonly used to interpolate along curves. It is based on the assumption that feature attribute values at different positions along the curve differ in proportion to the distances between those positions.

Given two point value pairs (p_s, v_s) and (p_t, v_t) , where p_s is the start point and p_t is the end point of a value segment, and v_s and v_t are the range values associated with those points, the range value v associated with the direct position x is shown in [Formula \(B.1\)](#):

$$v(x) = v_s + (v_t - v_s) ((x - p_s)/(p_t - p_s)) \quad (\text{B.1})$$

Bilinear interpolation is used to compute range values at direct positions in two dimensions. Given a direct position p contained in a grid cell whose vertices are V , $V + V_1$, $V + V_2$, and $V + V_1 + V_2$, and with range values at these vertices of w_1 , w_2 , w_3 and w_4 , respectively, there are unique numbers x and y , with $0 < x < 1$, and $0 < y < 1$ such that $p = V + xV_1 + yV_2$. The range value at p is shown in [Formula \(B.2\)](#):

$$w(x,y) = (1-x)(1-y) w_1 + x(1-y) w_2 + y(1-x) w_3 + x y w_4 \quad (\text{B.2})$$

NOTE In a Cartesian grid, V_1 and V_2 are the unit vectors (0,1) and (1,0).

B.4 Quadratic interpolation

Quadratic interpolation serves to compute values between direct positions in any-dimensional domains through a quadratic polynomial. The mathematical principle is shown for the 1D and 2D cases.

1D quadratic interpolation is used to interpolate along curves. It is based on the quadratic polynomial with coefficients a through c as shown in [Formula \(B.3\)](#):

$$v(x) = a + bx + cx^2 \quad (\text{B.3})$$

where

a is the range value at the start of a value segment; and

v is the range value at distance x along the curve from the start.

Three point value pairs are needed to provide control values for calculating the coefficients of the function.

Biquadratic interpolation is used to compute range values at direct positions in two dimensions. It is based on the biquadratic polynomial as shown in [Formula \(B.4\)](#):

$$v = a + bx + cy + dx^2 + exy + fy^2 + gx^2y + hxy^2 + ix^2y^2 \quad (\text{B.4})$$

B.5 Cubic interpolation

1D cubic interpolation serves to compute values between direct positions in any-dimensional domains through a cubic polynomial. The mathematical principle is shown for the 1D and 2D cases.

Cubic interpolation is used to interpolate along curves. It is based on the cubic polynomial, based on coefficients a through d as shown in [Formula \(B.5\)](#):

$$v(x) = a + bx + cx^2 + dx^3 \quad (\text{B.5})$$

where

a is the range value at the start of a value segment; and

v is the range value at distance x along the curve from the start.

Bicubic interpolation is used to compute range values at direct positions in two dimensions. Bicubic interpolation uses the following formula, based on coefficients a_0 through a_{15} , as shown in [Formula \(B.6\)](#):

$$v(x) = a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 + a_6x^2y + a_7xy^2 + a_8x^2y^2 + a_9x^3 + a_{10}y^3 + a_{11}x^3y + a_{12}xy^3 + a_{13}x^3y^2 + a_{14}x^2y^3 + a_{15}x^3y^3 \quad (\text{B.6})$$

B.6 Lost area interpolation

Lost area interpolation extends a multi-point coverage to a continuous function, f , defined on the convex hull of the domain of the point coverage.

For the following example, the following values apply:

- $D = \{x_1, x_2, \dots, x_n\}$ = the domain of the point coverage, and
- $\{V_1, V_2, \dots, V_n\}$ = the Thiessen polygons generated by the set D .

Suppose it is desired to calculate $f(q)$, where q is a direct position in the convex hull of D . First, the Thiessen polygons generated by D are formed; then p is added to the D and the Thiessen polygons for the set of $n+1$ points, $\{x_1, x_2, \dots, x_n, p\}$, are formed. The two sets of polygons are identical, except that each of the polygons coterminous with the polygon containing q “loses area” to the new polygon containing p .

The interpolation forms the weighted average such that each feature attribute value contributes to the feature attribute value at p according to the amount of area its polygon lost to the polygon at p . More formally:

- a) The discrete point coverage is characterized by the point value pairs: $\{(x_1, v_1), (x_2, v_2), \dots, (x_n, v_n)\}$.
- b) Among the Thiessen polygon set formed by $\{x_1, x_2, \dots, x_n, p\}$, those coterminous with the polygon containing p are $\{V_1, V_2, \dots, V_k\}$.
- c) The corresponding Thiessen polygons from the set generated by $\{x_1, x_2, \dots, x_n\}$ are $\{V'_1, V'_2, \dots, V'_k\}$.
- d) The area lost by the i^{th} polygon is $V'_i - V_i$.
- e) The total area lost is $\Sigma (V'_i - V_i)$ where the sum is over i from 1 to k (that is, the sum is over all polygons that lost area to the polygon containing q). Note that this sum is the same as the area of the Thiessen polygon containing p .
- f) Then the interpolated feature attribute value at p is:

$$f(p) = (\Sigma v_i * (V'_i - V_i)) / \Sigma (V'_i - V_i) \text{ where the summations are over the same range: } i = 1, \dots, k.$$

B.7 Barycentric interpolation

Let P , Q and R denote the vertices of a triangle. For any direct position, S , in the triangle, there is a unique triple of numbers, i, j and k , with $0 \leq i \leq 1$, $0 \leq j \leq 1$, and $0 \leq k \leq 1$, and with $i + j + k = 1$, such that as in [Formula \(B.7\)](#):

$$S = iP + jQ + kR \quad (\text{B.7})$$

The numbers (i, j, k) are the barycentric coordinates of S .

The name “barycentric” comes from the fact that using the equation above, S is the centre of mass of a triangle with point masses of size i, j and k at the corners P, Q and R respectively. As one allocates mass to the three corners, the centre of mass can occupy any direct position in the triangle.

Given a value triangle composed of the PointValuePairs (p_1, v_1) , (p_2, v_2) , and (p_3, v_3) , and a direct position, S , inside it, the barycentric coordinates of S are (i, j, k) , where $S = ip_1 + jp_2 + kp_3$ and the feature attribute value at S is $v = iv_1 + jv_2 + kv_3$.

B.8 Other

Other interpolation methods may be defined in an Application Schema that makes use of this document.

Annex C
(informative)

Sequential enumeration

C.1 General

Linearization of coverage range values (and their pertaining direct positions), such as for storage, is a choice of implementation (and storage format) and as such out of scope of this document. However, for reasons of backward compatibility with ISO 19123:2005 this annex discusses this concept, based on the original classes of ISO 19123:2005. Such a linearization will follow a “sequence rule” that assigns order to space (Figure C.1). The rule can be as simple as Row then Column, or it can be more complex. Complex rules allow for Quadtrees, or more general structures in Riemann hyperspace, Hilbert space and other patterns. An example is a spiral search pattern that may be used in Search and Rescue. SequenceRule is a data type that describes the mapping of grid coordinates to a position to attribute values along axes. SequenceType is a codelist that identifies methods for sequential enumeration of the grid points. Sample methods for sequential enumeration are described in this Annex C.

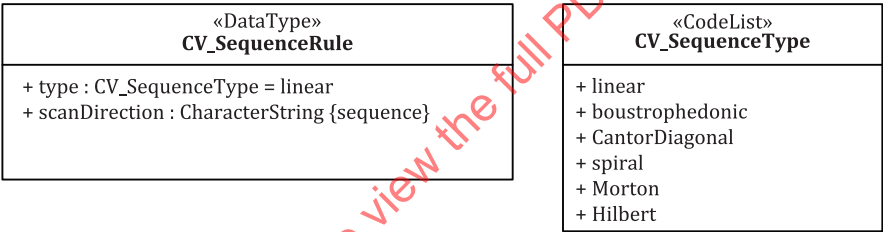


Figure C.1 — Sequence Rule structure and codelist

A sequential enumeration specifies a traversal order of a coverage domain. Two or more axis may be coded together to provide one linear order on a space. Such ordering is sometimes of value because it may provide properties to the number space that use or are in some operations on the data. For example, Riemann hyperspace is traversed by Morton order. This allows for variable size cells (such as in a quadtree in 2 dimensions) and obeys the Riemann criteria where values that are close together in the Morton order are also close together in physical space. Technically this is an extension of the Pythagorean theorem into hyperspace. SequenceType provides a list of codes for identifying types of sequencing methods. This annex explains those types in greater detail.

There are several sequencing rules based on incrementing or decrementing grid coordinate values in a simple fashion. More complex space filling curves can also be used. Space filling curves are generated by progressively subdividing a space in a regular way and connecting the elements resulting from each subdivision according to some rule. They can be used to generate a grid, but they can also be used to assign an ordering to the grid points or grid cells in a separately defined grid. They lend themselves more readily than simple incrementing methods to sequencing in grids that have irregular shapes or cells of variable size.

In every case, ordering of the grid cells starts by incrementing coordinates along one grid axis. At some point in the process, it begins to increment coordinates along a second grid axis, then a third, and so on until it has progressed in the direction of each of the grid axes. The figures in this annex provide examples. The attribute SequenceRule.scanDirection provides a list of signed axis names that identifies the order in which scanning takes place. The list may include an additional element to support interleaving of feature attribute values (see C.8 for a more detailed discussion of interleaving).

Ordering is continuous if consecutive pairs of grid cells in the sequence are maximally connected. It is semi-continuous if consecutive pairs of grid cells are connected, but less than maximally connected, and discontinuous if consecutive pairs of cells are not connected.

EXAMPLE In the 2D case, a quadrilateral grid cell is connected to the eight cells with which it shares at least one corner. It is maximally connected to the four cells with which it shares an edge and two corners. In the 3D case, a cell is maximally connected to those cells with which it shares a face.

NOTE In the example diagrams of this annex, continuous segments of scan lines are shown as solid lines, and discontinuous segments are shown as dashed lines.

C.2 Linear scanning

In linear scanning (Figure C.2 through C.5), feature attribute value records are assigned to consecutive grid points along a single grid line parallel to the first grid axis listed in scanDirection. Once scanning of that row is complete, assignment of feature attribute value records steps to another grid line parallel to the first and continues to step from grid line to grid line in a direction parallel to the second axis. If the grid is 3D, the sequencing process completes the assignment of feature attribute value records to all grid points in one plane, then steps to another plane, then continues stepping from plane to plane in a direction parallel to the third axis of the grid. The process can be extended to any number of axes. Linear scanning is continuous only along a single grid line.

NOTE The axes of 2D grids are often called “row” (horizontal) and “column” (vertical). In this case, scanning in (x,y) order is sometimes called row or row-major scanning.

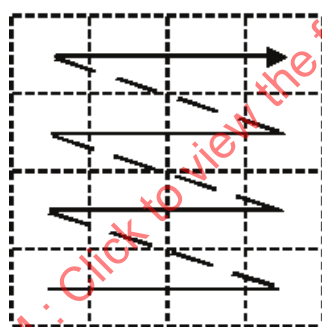


Figure C.2 — Examples of linear scanning in a 2D grid in (x,y) order

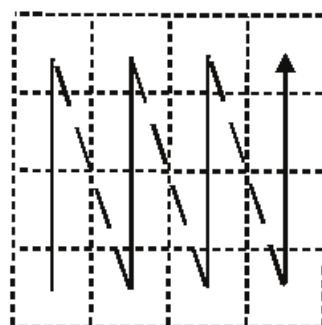


Figure C.3 — Examples of linear scanning in a 2D grid in (y,x) order

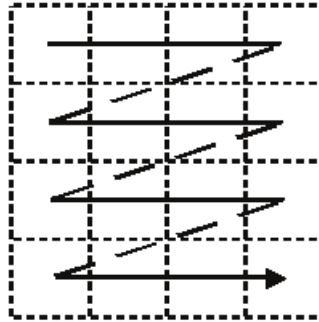


Figure C.4 — Examples of linear scanning in a 2D grid in (x,y) order

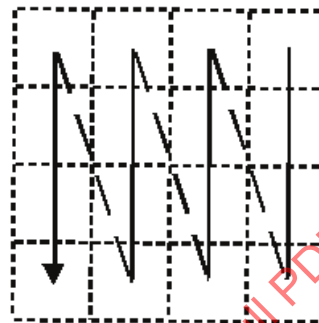


Figure C.5 — Examples of linear scanning in a 2D grid in (y,x) order

C.3 Boustrophedonic scanning

In a variant of linear scanning, known as boustrophedonic or byte-offset scanning, the direction of the scan is reversed on alternate grid lines (Figure C.6 through C.9). In the case of a 3D grid, it will also be reversed in alternate planes. Boustrophedonic scanning is continuous.

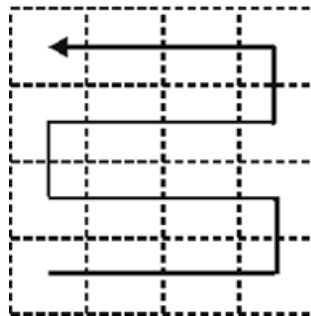


Figure C.6 — Examples of boustrophedonic scanning in a 2D grid in (x,y) order

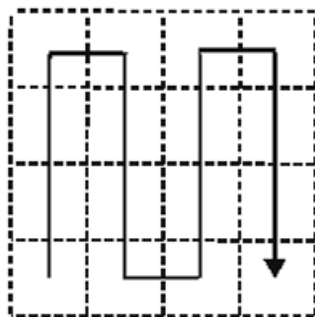


Figure C.7 — Examples of boustrophedonic scanning in a 2D grid in (y,x) order

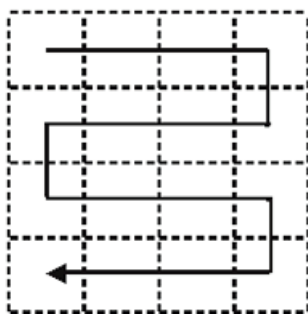


Figure C.8 — Examples of boustrophedonic scanning in a 2D grid in (x,y) order

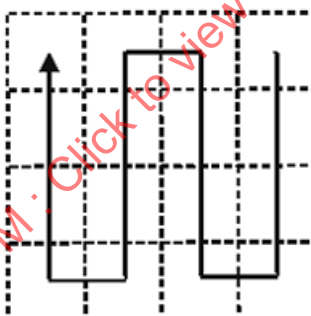


Figure C.9 — Examples of boustrophedonic scanning in a 2D grid in $(-y,-x)$ order

C.4 Cantor-diagonal scanning

Cantor-diagonal scanning, also called zigzag scanning, orders the grid points in alternating directions along parallel diagonals of the grid (Figure C.10 through C.13). The scan pattern is affected by the direction of the first step. Like linear scanning, Cantor-diagonal scanning can be extended to grids of three or more dimensions by repeating the scan pattern in consecutive planes. Cantor-diagonal scanning is semi-continuous within a single plane.

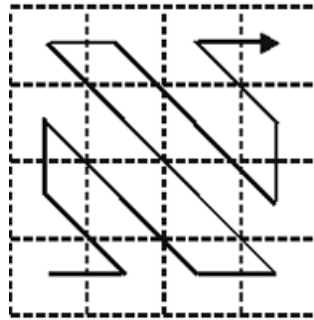


Figure C.10 — Examples of Cantor-diagonal scanning in a 2D grid in (x,y) order

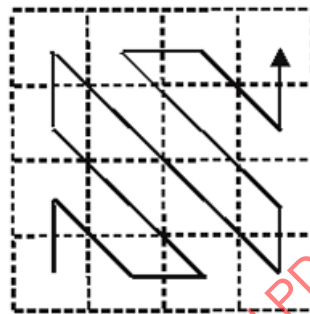


Figure C.11 — Examples of Cantor-diagonal scanning in a 2D grid in (y,x) order

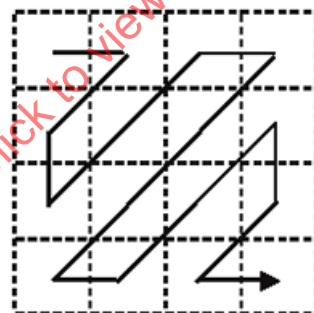


Figure C.12 — Examples of Cantor-diagonal scanning in a 2D grid in (x,y) order

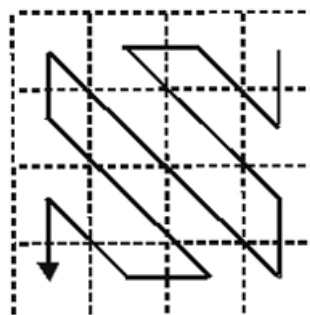


Figure C.13 — Examples of Cantor-diagonal scanning in a 2D grid in $(-y,-x)$ order

C.5 Spiral scanning

Spiral scanning (Figure C.14 through C.17) can begin either at the centre of the grid (outward spiral), or at a corner (inward spiral). Like linear or Cantor-diagonal scanning, spiral scanning can be extended to grids of three or more dimensions by repeating the scan pattern in consecutive planes. Spiral scanning is continuous in any one plane, but continuity in grids of more than two dimensions can only be maintained by reversing the inward/outward direction of the scan in alternate planes.

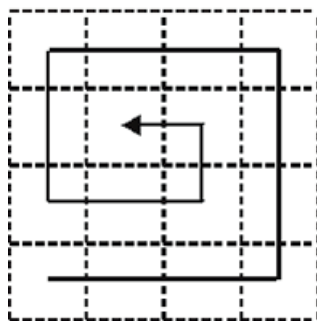


Figure C.14 — Examples of spiral scanning in a 2D grid in (x,y) order

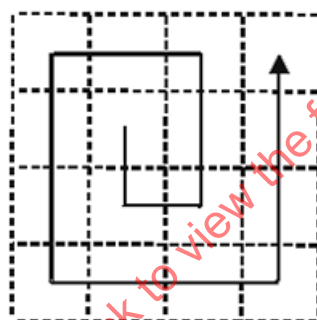


Figure C.15 — Examples of spiral scanning in a 2D grid in (y,x) order

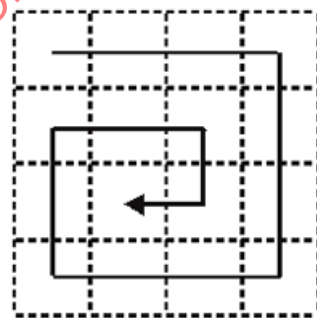


Figure C.16 — Examples of spiral scanning in a 2D grid in $(x,-y)$ order

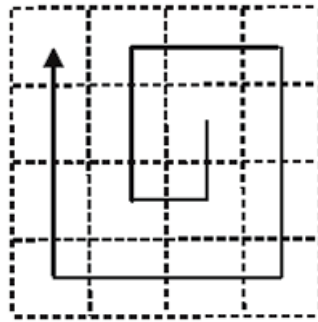


Figure C.17 — Examples of spiral scanning in a 2D grid in $(-y, -x)$ order

C.6 Morton order

Morton ordering is typically based on a space-filling curve generated by progressively subdividing a space into quadrants and ordering the quadrants in a Z pattern as shown in [Figure C.18](#) through [C.21](#). The ordering index for each grid point is computed by converting the grid coordinates to binary numbers and interleaving the bits of the resulting values. Given the list of the grid axes specified by *SequenceRule.scanDirection*, the bits of the coordinate corresponding to an axis are less significant than those of the coordinate corresponding to the next axis in the list. Morton ordering can be extended to any number of dimensions. Morton ordering is discontinuous.

NOTE Because of the shape of the curve formed by the initial ordering of quadrants, Morton ordering is also known as Z ordering.

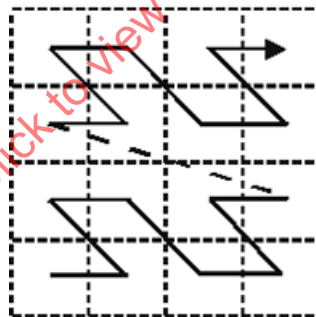


Figure C.18 — Examples of Morton ordering in a 2D grid in (x, y) order

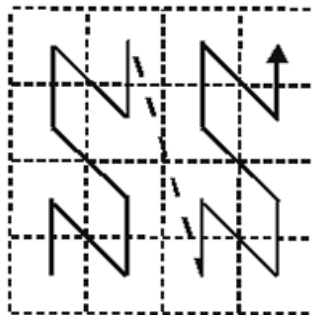


Figure C.19 — Examples of Morton ordering in a 2D grid in (y, x) order

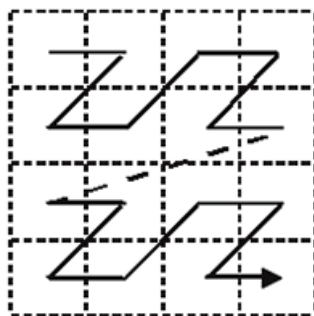


Figure C.20 — Examples of Morton ordering in a 2D grid in (x,y) order

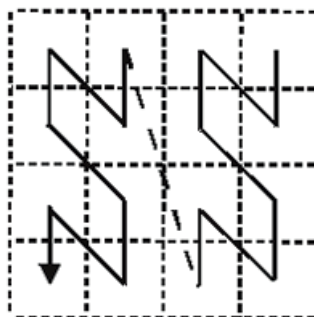


Figure C.21 — Examples of Morton ordering in a 2D grid in $(-y,-x)$ order

A grid generated with the Morton ordering technique described will be square and its size in each direction will be a multiple of a power of two. However, the bit interleaving technique for generating an index can be used to order the grid points in any grid, including grids that are irregular in shape or have grid cells of different sizes ([Figure C.22](#) to [Figure C.25](#)).

Morton ordering can also be used with subdivisions of higher order than 2×2 . A 3×3 subdivision for example, or any other odd number, preserves the location of the central cell in systems with hierarchical subdivisions ([Figure C.22](#) to [Figure C.25](#)). For 3×3 subdivision the ordering index for each grid point is computed by converting the grid coordinates to base3 digits and interleaving the base3 digits of the resulting values. In the 2D case pairs of base3 digits can be combined to form a base 9 digit, in 3- or 4-dimensions groups of 3- or 4- base3 digits can be combined to form base 27 or base 81 digits, any of which can be coded as a single ASCII digit.

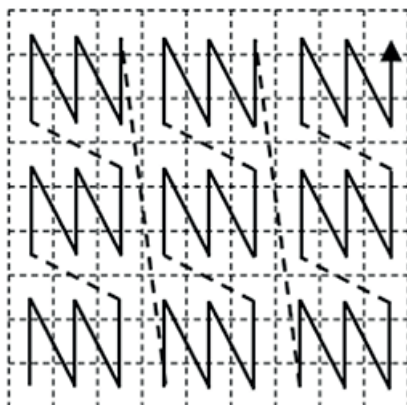


Figure C.22 — Examples of Morton ordering in irregular grids: 3×3 grid

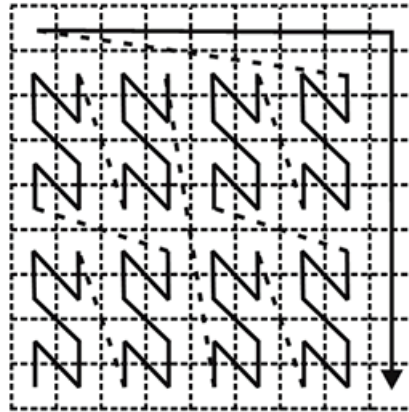


Figure C.23 — Examples of Morton ordering in irregular grids: 9x9 grid

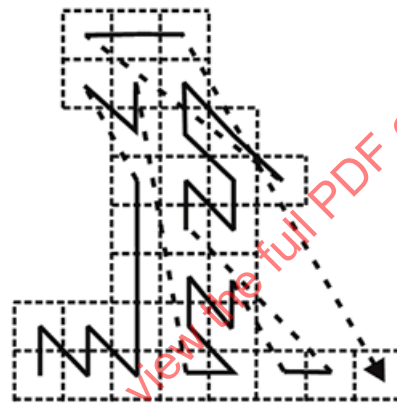


Figure C.24 — Examples of Morton ordering in irregular grids: irregular shape

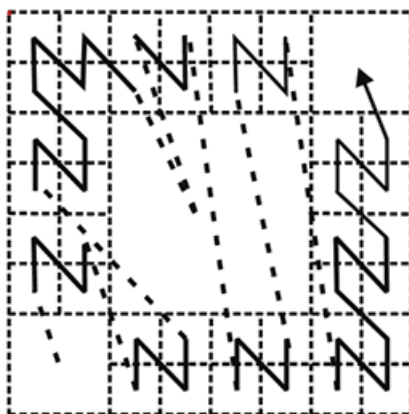


Figure C.25 — Examples of Morton ordering in irregular grids: variable size cells

C.7 Hilbert order

Like Morton ordering, Hilbert ordering is based on a space-filling curve generated by progressively subdividing a space into quadrants, but the initial pattern of subdivision is different for Hilbert curves. Further subdivision involves replacement of parts of the curve by different patterns ([Figure C.26](#)), unlike the simple replication of a single pattern as in Morton ordering. There are two sets of patterns. The left-hand column of the figure includes those for which the sense of the scan directions is the same

– both are positive or both negative. The right-hand column of the figure includes those for which the sense of the scan directions is opposite – one is positive and one is negative. A Hilbert curve can only be constructed with patterns from the same set; it uses all the patterns in that set.

NOTE Because of the shape of the curve formed by the initial ordering of quadrants, Hilbert ordering is also known as Pi ordering.

Computation of the ordering index is more complicated for Hilbert ordering than for Morton ordering. Algorithms for the 2D case (Figure C.27) are described in Reference [26] and Reference [19]. 3D Hilbert curves are discussed in Reference [18]. Hilbert ordering is continuous.

C.8 Interleaving of feature attribute values

When the range of a grid coverage includes more than one feature attribute, the feature attribute values may be interleaved in various ways within a list. Such interleaving can be described by including an element for the range in the list of axes provided by the *scanDirection* attribute of *SequenceRule*. The index for the record of attributes is then incremented in the same way as the coordinates.

EXAMPLE Consider the 2×2 grid in Figure C.28. It has a range (r) of two attributes, A and B. Assuming a linear scan positive first in the x and then in the y direction, the scan order can be selected to access the feature attribute values in the different ways shown in Table C.1.

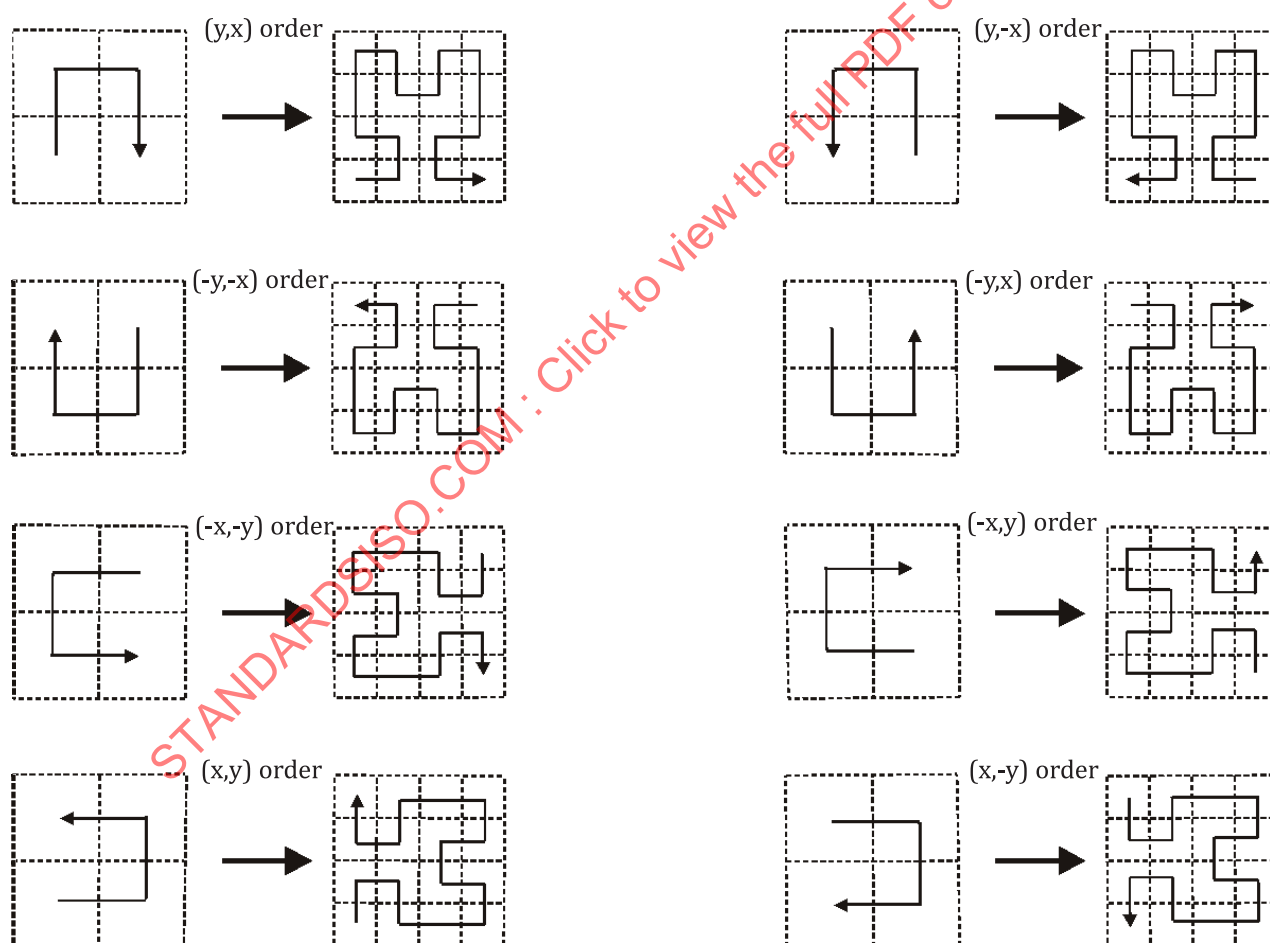


Figure C.26 — Replacement patterns for generating a Hilbert curve

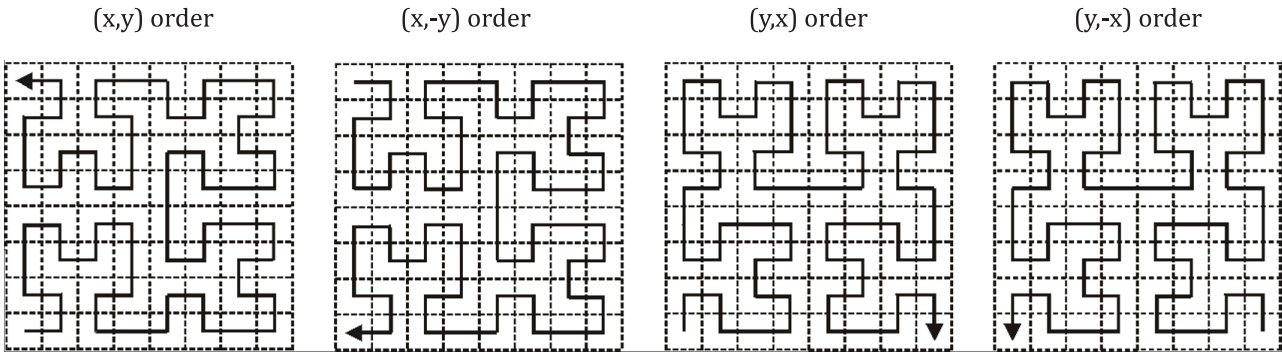


Figure C.27 — Examples of Hilbert ordering in a 2D grid

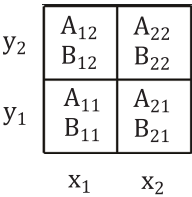


Figure C.28 — Example of a 2D grid with a range having two attributes

Table C.1 — Examples of interleaving

Order	Scan direction		
	r, x, y	x, y, r	x, r, y
1	A_{11}	A_{11}	A_{11}
2	B_{11}	A_{21}	A_{21}
3	A_{21}	A_{12}	B_{11}
4	B_{21}	A_{22}	B_{21}
5	A_{12}	B_{11}	A_{12}
6	B_{12}	B_{21}	A_{22}
7	A_{22}	B_{12}	B_{12}
8	B_{22}	B_{22}	B_{22}

Annex D (normative)

Legacy data-centric coverage specification

D.1 Overview

This annex summarizes the original coverage specification of ISO 19123:2005 which has been reshaped in this document according to ISO's current modelling approach. This document (ISO 19123-1) establishes an abstract conceptual coverage model. [Annex D](#) is intended to instantiate the abstract concepts described in [Clauses 5](#) to [10](#) in accordance with the legacy specifications contained in ISO 19123:2005.

Modelling approaches in ISO have evolved over time. In the past, data structures were described. More recently, ISO has moved to specifying interfaces instead because this better hides implementation details. Such a step of abstraction is particularly important in the context of separating the original, single International Standard, ISO 19123:2005, into two parts: an abstract specification in ISO 19123-1 (this document) and implementation guidance in ISO 19123-2.

The interface-centric approach to coverage modelling is adopted in [Clauses 5](#) through [10](#). At the same time, this document normatively retains the data-centric coverage description from ISO 19123 as [Annex D](#). As such, the coverage data structures of ISO 19123:2005 represent one valid generic data structure out of many possible realizations satisfying the interface of ISO 19123-1. Specifically, the ISO 19123:2005 coverage structure specification is retained for compatibility with realizations that have realized these generic classes, albeit in their individual ways. This annex is not a retention of the old elements from ISO 19123:2005, but rather a definition of equivalent classes in terms of the new ISO 19123-1 structure to satisfy the backward compatibility requirement.

ISO 19123:2005 defined generic data classes which could be specialized by the developers of more detailed product specifications which could then be implemented. External organizations reference these generic data classes. These classes are realized in these external standards. Since references to these classes exist in external standards, it is important that equivalent generic data classes be maintained in this edition of the document.

This edition of ISO 19123-1 takes a different approach. It defines an interface which provides a standardized window through which a variety of different data structures could produce the same result. In that way it is more flexible than the previous approach. But it is not in conflict with the previous approach. The previous generic data classes are still valid, and the external standards or product specifications that referenced them also remain valid. The new edition of ISO 19123-1 is simply more flexible, allowing many different implementation structures, not just the defined generic data classes presented in this annex.

This annex specifies the equivalent abstract data classes from ISO 19123:2005 that are developed in this version to retain compatibility. These form a compliant data structure that can be interpreted through the interface structure defined in ISO 19123-1. The classes and attributes are mapped from the older structure to the new interface structure.

No XML Schema is provided for the instantiated structures defined in [Annex D](#).

D.2 Generic data structure coverage schema

D.2.1 Generic data structure package

This annex is described in a separate package from the rest of the model in this document in order to ensure that the namespace is unique. That is, the class name *CV_Coverage* <<interface>> is used to describe the root class of the interface structure, and the class name *CV_Coverage* <<featureType>> is used in this annex to describe the root class of the generic data structure. *CV_Coverage* <<featureType>> realizes *CV_Coverage* <<interface>>. The names are used in this way to ensure backward compatibility. A mapping is shown from the generic data structure classes to the new interface classes *CV_Coverage* Feature Type Generic Data Class.

D.2.2 Class *CV_Coverage* as a data structure template

D.2.2.1 Class description

The class *CV_Coverage* <<featureType>> is an instance of the <<metaclass>> *GF_FeatureType* (ISO 19109), which therefore represents a feature type. *CV_Coverage* <<featureType>> supports five attributes, one operation and three associations. This is illustrated in [Figure D.1](#).

D.2.2.2 Attributes

D.2.2.2.1 *domainExtent*

The attribute *domainExtent*: *EX_Extent*[1..*] contains the extent of the domain of the coverage. The data type *EX_Extent* is defined in ISO 19115-1. Extents may be specified in space, time or space-time. The attribute *domainExtent* has been generalized in this document. This older narrower extent has been retained for backward compatibility.

D.2.2.2.2 *rangeType*

The attribute *rangeType*: *RecordType* describes the range of the coverage. The data type *RecordType* is defined in ISO 19103. It consists of a list of attribute name/data type pairs. A simple list is the most common form of *rangeType*, but *RecordType* can be used recursively to describe more complex structures. The *rangeType* for a specific coverage shall be specified in an application schema.

D.2.2.2.3 *commonPointRule*

The optional attribute *commonPointRule*: *CV_CommonPointRule* identifies the procedure to be used for evaluating the *CV_Coverage* at a position that falls either on a boundary between geometric objects or within the boundaries of two or more overlapping geometric objects, where the geometric objects are either *CV_DomainObjects* or *CV_ValueObjects*. This attribute is optional and takes the default value of the attribute as “average”. The attribute is not required in the case when there is only one extent per coverage; that is, when geometric objects do not overlap.

D.2.2.2.4 *interpolationType*

The optional attribute *interpolationType*: *CV_InterpolationMethod* is a code that identifies the interpolation method that shall be used to derive a feature attribute value at any direct position within the *CV_ValueObject*. The attribute is optional. The default value is “nearestNeighbor”. No value is needed for an analytical coverage (one that maps direct position to attribute value by using a mathematical function rather than by interpolation). Interpolation methods are described in [5.6](#) and in [Annex B](#). The codelist identifying the interpolation method is shown in [Figure D.3](#).

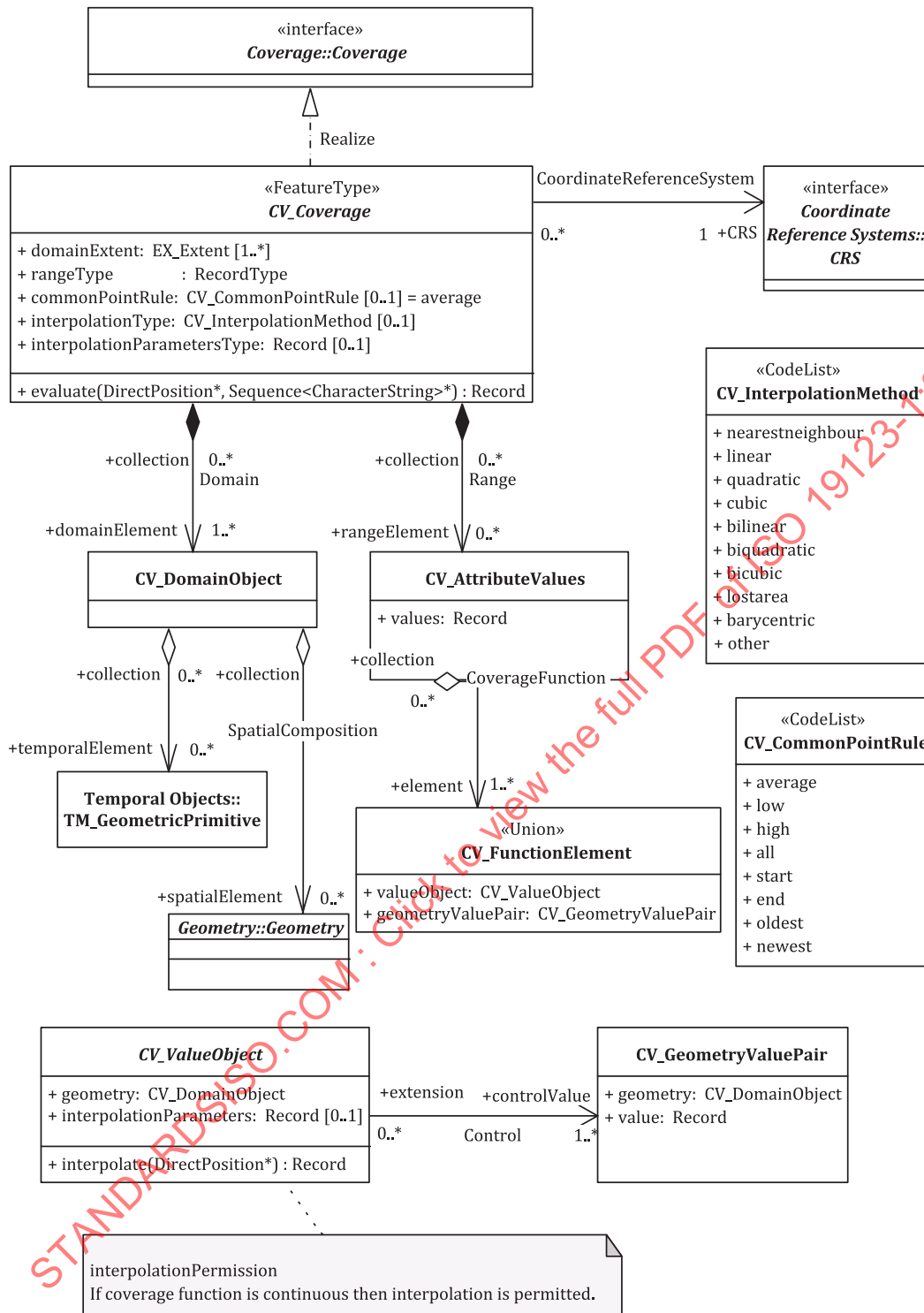


Figure D.1 — CV_Coverage

D.2.2.2.5 interpolationParameterTypes

Although many interpolation methods use only the values in the coverage range as input to the interpolation function, there are some methods that require additional parameters. The optional attribute *interpolationParameterTypes* specifies the types of parameters that are needed to support the interpolation method identified by *interpolationType*. The data type RecordType is specified in ISO 19103. It is a dictionary of names and data types.