

Open Geospatial Consortium

Submission Date: <yyyy-mm-dd>

Approval Date: <yyyy-mm-dd>

Publication Date: <2021-01-15>

External identifier of this OGC® document: <http://www.opengis.net/doc/EG/{standard}/{m.n}>

Internal reference number of this OGC® document: YY-nnnrx

Version: n.n

Category: OGC® Engineering Guidance

Editor: <Name(s) of Editor or Editors>

OGC (add title text)

Copyright notice

Copyright © <year> Open Geospatial Consortium

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>

Warning

This document provides Engineering Guidance on the use of the OGC <name of standard> Standard. This document is a non-normative resource and not an official position of the OGC membership. It is subject to change without notice and may not be referred to as an OGC Standard. Further, Engineering Guidance should not be referenced as required or mandatory technology in procurements.

Document type: OGC®Engineering Guidance

Document subtype:

Document stage: Draft

Document language: English

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications. This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

Table of Contents

1. Introduction	5
2. How To Use This Resource	6
3. Scope	7
4. References	8
5. Terms and Definitions	9
6. Conventions	11
6.1. Identifiers	11
7. GeoPOS Foundations	12
7.1. Concept Name	12
7.2. Concept Name	12
8. GeoPOS Model	13
8.1. Core	13
8.1.1. Key Concepts	13
8.1.2. ISO Dependencies	14
8.1.3. City Models and City Objects	20
8.1.4. Space Concept	21
8.1.5. Geometry and LOD	21
8.1.6. CityGML Core UML Model	21
8.2. Appearance	21
8.2.1. Synopsis	21
8.2.2. Key Concepts	21
8.2.3. Discussion	21
8.2.4. UML Model	21
8.2.5. Examples	21
9. CityGML Data Dictionary	22
Annex A: Revision History	23
Annex B: Glossary	24
B.1. ISO Concepts	25
B.2. Abbreviated Terms	29
Annex C: Bibliography	31

i. Abstract

<Insert Abstract Text here>

ii. Keywords

The following are keywords to be used by search engines and document catalogues.

ogcdoc, OGC document, <tags separated by commas>

iii. Preface

NOTE

Insert Preface Text here. Give OGC specific commentary: describe the technical content, reason for document, history of the document and precursors, and plans for future work. > Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

iv. Submitting organizations

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

Organization name(s)

v. Submitters

All questions regarding this submission should be directed to the editor or the submitters:

Name Affiliation

Chapter 1. Introduction

Tell us about the GeoPOS standard, the Users Guide, and what it is for.

Chapter 2. How To Use This Resource

The Users Guide to the GeoPOS Conceptual Model Standard is not intended to be read from start to finish. Rather, it is a resource structured to provide quick answers to questions which an implementer may have about the GeoPOS Standard.

The GeoPOS Standard includes hyperlinks which can be used to navigate directly to relevant sections of the Users Guide.

NOTE | This is a rough draft. A more refined writeup will be available at a latter date.

Chapter 3. Scope

NOTE

Insert Scope text here. Give the subject of the document and the aspects of that scope covered by the document.

Chapter 4. References

The following documents contain provisions that, through reference in this text, constitute provisions of this Users Guide. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the document referred to applies.

- IETF: RFC 2045 & 2046, Multipurpose Internet Mail Extensions (MIME). (November 1996),
- IETF: RFC 3986, Uniform Resource Identifier (URI): Generic Syntax. (January 2005)
- INSPIRE: D2.8.III.2 Data Specification on Buildings – Technical Guidelines. European Commission Joint Research Centre.
- ISO: ISO 19101-1:2014, Geographic information - Reference model - Part 1: Fundamentals

NOTE

Each reference has an anchor. That allows users to jump to this citation from any hyperlinked reference in the text. The second part of the anchor is the text that will be displayed such as [RFC 2045](#)

Chapter 5. Terms and Definitions

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

2D data

geometry of features is represented in a two-dimensional space

NOTE In other words, the geometry of 2D data is given using (X,Y) coordinates.

[INSPIRE D2.8.III.2, definition 1]

2.5D data

geometry of features is represented in a three-dimensional space with the constraint that, for each (X,Y) position, there is only one Z

[INSPIRE D2.8.III.2, definition 2]

3D data

Geometry of features is represented in a three-dimensional space.

NOTE In other words, the geometry of 2D data is given using (X,Y,Z) coordinates without any constraints.

[INSPIRE D2.8.III.2, definition 3]

application schema

A set of [conceptual schema](#) for data required by one or more applications. An application schema contains selected parts of the base schemas presented in the ORM Information Viewpoint. Designers of application schemas may extend or restrict the types defined in the base schemas to define appropriate types for an application domain. Application schemas are information models for a specific information community.

OGC Definitions Register at <http://www.opengis.net/def/glossary/term/ApplicationSchema>

codelist

A value domain including a code for each permissible value.

conceptual model

model that defines concepts of a universe of discourse

[ISO 19101-1:2014, 4.1.5]

conceptual schema

1. formal description of a [conceptual model](#)
[ISO 19101-1:2014, 4.1.6]
2. base schema. Formal description of the model of any geospatial information. [Application schemas](#) are built from conceptual schemas.
OGC Definitions Register at <http://www.opengis.net/def/glossary/term/ConceptualSchema>

Implementation Specification

Specified on the OGC Document Types Register at <http://www.opengis.net/def/doc-type/is>

NOTE

Notice that each definition has an anchor. Anchor text would also be a good idea which we may include latter. Terms used within a definition should be cross-linked to their definition if it is included in this document (see [Application Schema](#) for an example).

Chapter 6. Conventions

6.1. Identifiers

The normative provisions in this document are denoted by the URI

<http://www.opengis.net/spec/{standard}/{m.n}>

All requirements and conformance tests that appear in this document are denoted by partial URIs which are relative to this base.

Chapter 7. GeoPOS Foundations

This section in introductory text for the foundational concepts which will be discussed.

This file may be omitted if you have nothing to say.

7.1. Concept Name

Describe your concept here. Write to the developer, this is not a disertation.

7.2. Concept Name

Describe your concept here.

Don't forget to include anchors for hyperlinks. Allow users to jump in, get what they need, and jump out.

Chapter 8. GeoPOS Model

8.1. Core

Contributors	
C. Heazel - first draft	

NOTE in this example this module is first divided into multiple sections. This is done to make the concept more accessible.

The CityGML Core module defines the basic concepts and components of city models. This rather large body of work is divided into seven sections. These sections build on each other from the fundamental principles specified by the relevant ISO standards up to the full CityGML model. These sections are summarized in [Table 1](#).

Table 1. CityGML Core Sections

Key Concepts	Summarizes the key concepts described in the Core module.
The Use of ISO Standards	Describes the use of the ISO 19100 series of International Standards to provide a foundation to the CityGML model.
City Models and City Objects	Defines the basic building blocks of the CityGML model.
Space Concept	Defines the concepts of space as used in the CityGML model.
Geometry and LOD	Defines the geometry and Levels Of Detail concepts.
CityGML Core Model	Presents the complete Core model.
Types, Enumerations, and Codelist	Defines the little things which make this model work.

8.1.1. Key Concepts

The following is a summary of the key concepts described by the Core Module. This is not an exhaustive listing of all of the Core concepts. Rather, it is an introduction those concepts which are essential for understanding the role of the Core Module in the CityGML Conceptual Model.

CityModel: ([Discussion](#)) The CityModel class is the root class of every CityGML conceptual model. It's primary purpose is to aggregate CityModelMembers.

NOTE Each entry includes a link to the corresponding data dictionary (click on the name) and a link to further discussion of the concept (discussion). Furthermore, each concept has an anchor which makes it a potential target of an internal link.

8.1.2. ISO Dependencies

NOTE these classes are commonly used in OGC standards. I left the content in case you have a use for it.

CityGML builds on the ISO 19100 family of standards. The applicable standards are identified in [Figure 1](#). A [Data dictionary](#) is also included for all of the ISO-defined classes explicitly referenced in the CityGML UML model. These data dictionaries are provided for the convenience of the user. The ISO standards are the normative source.

Use of ISO Standards in CityGML

/// image::.../standard/figures/Core/ISOandOASISstandardsinCityGML.png[align="center"]

Classes

The ISO classes explicitly used in the CityGML UML model are introduced in [Table 2](#). Detailed descriptions are provided in the [Data Dictionary](#).

Table 2. ISO Classes used in CityGML

Class Name	Description
AnyFeature	A generalization of all feature types
CV_DiscreteGridPointCoverage	A coverage that returns the same feature attribute values for every direct position within any object in its domain.
DirectPosition	The coordinates for a position within some coordinate reference system.
GM_Object	The root class of the geometric object taxonomy.
GM_MultiCurve	An aggregate class containing only instances of GM_OrientableCurve.
GM_MultiPoint	An aggregate class containing only points.
GM_MultiSurface	An aggregate class containing only instances of GM_OrientableSurface.
GM_Point	The basic data type for a geometric object consisting of one and only one point.
GM_Solid	The basis for 3-dimensional geometry. The extent of a solid is defined by the boundary surfaces.
GM_Surface	The basis for 2-dimensional geometry.
GM_Tin	A GM_TriangulatedSurface which uses the Delaunay or similar algorithm.
GM_TriangulatedSurface	A GM_PolyhedralSurface that is composed only of triangles
SC_CRS	Coordinate reference system which is usually single but may be compound.
TM_Position	A union class that consists of one of the data types listed as its attributes.

Geometry

The most common geometry concept found in the CityGML 3.0 Standard is the concept of [multi-](#)

[primitives](#). These are homogeneous collections of [GM_Primitives](#) which are aggregated to form a more complex geometry.

[GM_Composites](#) are another form of [GM_Primitive](#) collection. These differ from [GM_MultiPrimitive](#) in that the collection can be heterogeneous. It should be noted that none of the classes in the CityGML 3.0 Standard are descended from [GM_Composites](#). However, the terms "CompositeCurve", "CompositeSurface", and "CompositeSolid" do appear in the text. The [composit](#) concept can also be seen in the association between spaces and surfaces. Therefore, an explanation of [composits](#) has been included for completeness.

GM_Primitive

GM_Primitive is the abstract root class of the geometric primitives. Its main purpose is to define the basic "boundary" operation that ties the primitives in each dimension together. A geometric primitive (GM_Primitive) is a geometric object that is not decomposed further into other primitives in the system. This includes curves and surfaces, even though they are composed of curve segments and surface patches, respectively. This composition is a strong aggregation: curve segments and surface patches cannot exist outside the context of a primitive.

NOTE Most geometric primitives are decomposable infinitely many times. Adding a centre point to a line may split that line into two separate lines. A new curve drawn across a surface may divide that surface into two parts, each of which is a surface. This is the reason that the normal definition of primitive as "non-decomposable" is not plausible in a geometry model - the only non-decomposable object in geometry is a point.

GM_MultiPrimitive

Any geometric object that is used to describe a feature is a collection of [geometric primitives](#). A homogeneous collection of geometric primitives may be a multi-primitive (GM_MultiPrimitive). Geometric complexes have additional properties specific to the type of [geometric primitive](#) they aggregate.

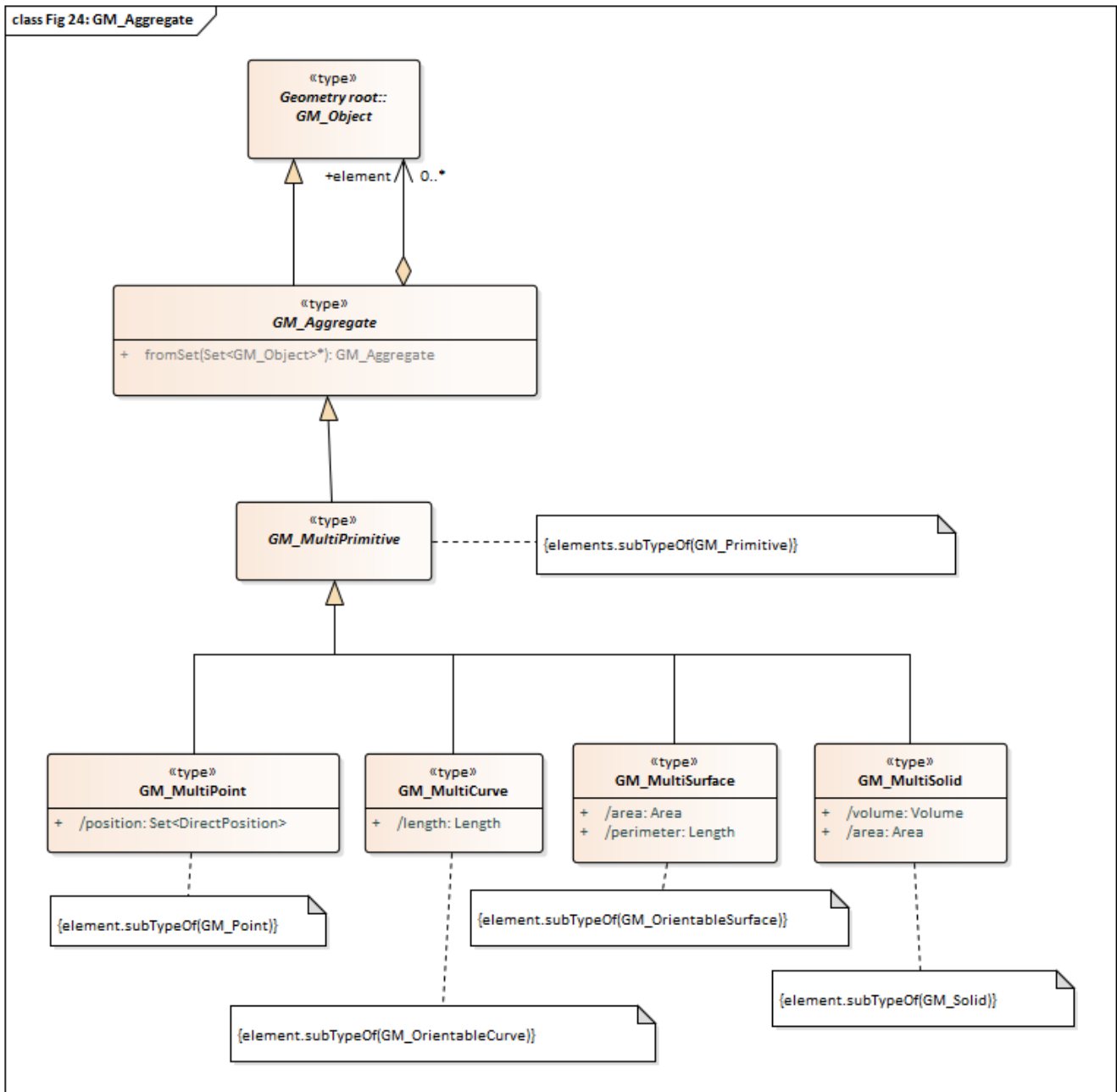


Figure 1. GM_MultiPrimitive Context Diagram

GM_Complex

A GM_Complex is a set of disjoint geometric primitives ([GM_Primitive](#)) such that the boundary of each primitive can be represented as the union of other geometric primitives within the complex.

Any geometric object that is used to describe a feature is a collection of [geometric primitives](#). A collection of geometric primitives may be a geometric complex (GM_Complex). Geometric complexes have additional properties such as closure by boundary operations and mutually exclusive component parts.

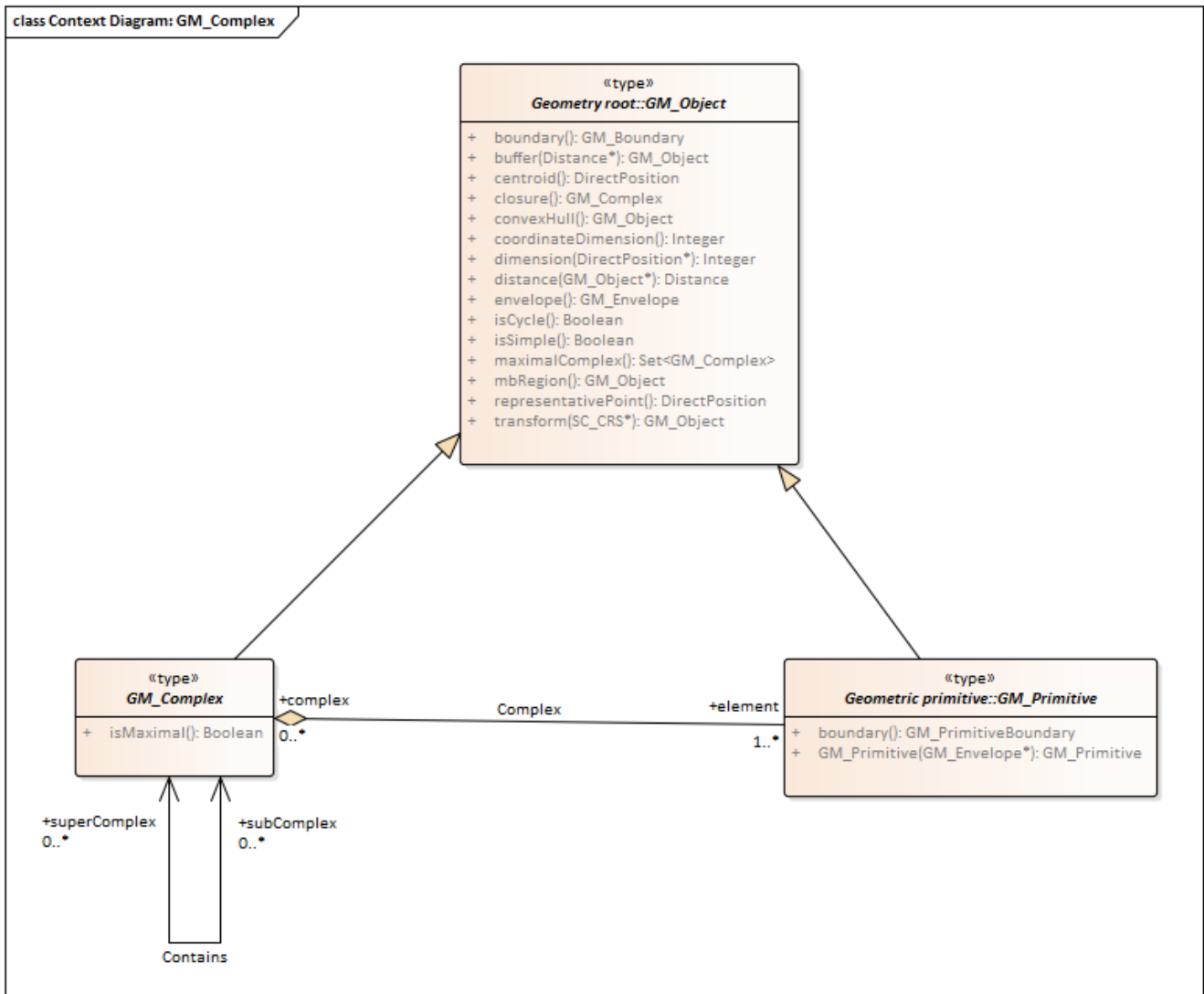


Figure 2. GM_Complex Context Diagram

GM_Primitive and GM_Complex share most semantics, in the meaning of operations, attributes and associations. There is an exception in that a GM_Primitive shall not contain its boundary (except in the trivial case of GM_Point where the boundary is empty), while a GM_Complex shall contain its boundary in all cases. This means that if an instantiated object implements GM_Object operations both as GM_Primitive and as a GM_Complex, the semantics of each set theoretic operation is determined by the its name resolution. Specifically, for a particular object such as GM_CompositeCurve, GM_Primitive::contains (returns FALSE for end points) is different from GM_Complex::contains (returns TRUE for end points). Further, if that object is cast as a GM_Primitive value and as a GM_Complex value, then the two values need not be equal as GM_Objects.

GM_Complex aggregates GM_Primitives through the **element** property. Since this is an aggregation, the target GM_Primitive may be associated with more than one GM_Complex.

A GM_Complex object can also have a whole/part relationship with other GM_Complex objects. The **contains** association is used to associate the **superComplex** instance with the **subComplex** instance.

Note that the geometric primitives in the set are mutually exclusive in the sense that no point is interior to more than one primitive. The set is closed under boundary operations, meaning that for each element in the complex, there is a collection (also a complex) of geometric primitives that

represents the boundary of that element.

GM_Composite

GM_Composite is a subclass of [GM_Complex](#). Like [GM_Complex](#), it has an association with [GM_Primitives](#). In this case this is an **composition** association with a **composite** role (GM_Composite) and a **generator** role (GM_Primitive). As with the GM_Complex, the GM_Primitive may be associated with more than one GM_Composite.

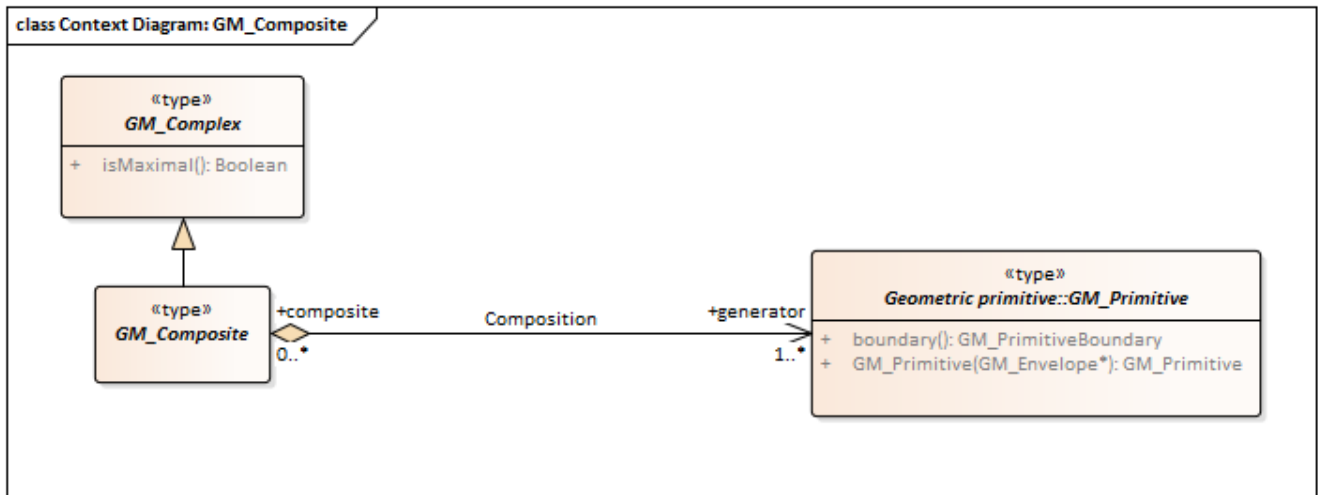


Figure 3. GM_Composite Context Diagram

GM_CompositeSurface

A GM_Composite where the [GM_Primitives](#) is a [GM_OrientatableSurface](#).

A GM_CompositeSurface is also a subclass of [GM_Primitives](#) is a [GM_OrientatableSurface](#). One of the few examples of multiple inheritance.

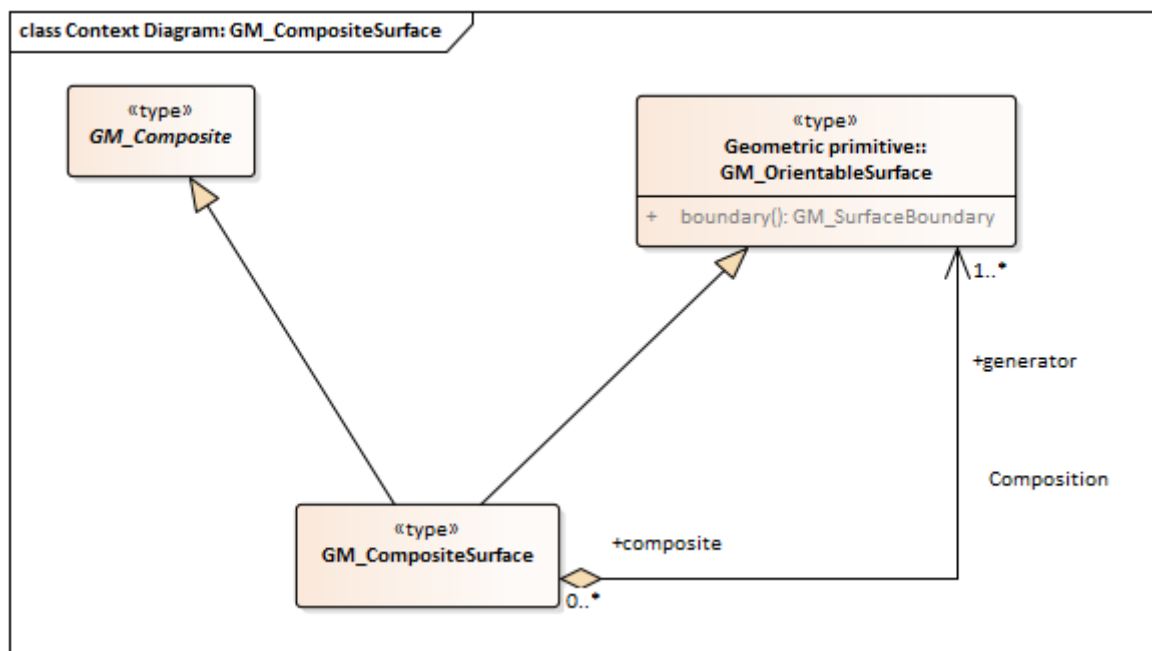


Figure 4. GM_CompositeSurface Context Diagram

GM_OrientableSurface

GM_OrientableSurface consists of a surface and an orientation inherited from GM_OrientablePrimitive. If the orientation is "+", then the GM_OrientableSurface is a GM_Surface. If the orientation is "-", then the GM_OrientableSurface is a reference to a GM_Surface with an upNormal that reverses the direction for this GM_OrientableSurface, the sense of "the top of the surface" (see 6.4.33.2).

```
GM_OrientableSurface:
{Orientation = "+" implies primitive = self};
{(Orientation = "-" and TransfiniteSet::contains(p : DirectPosition)) implies
(primitive.upNormal(p) = - self.upNormal(p))};
```

GM_CompositeCurve

A GM_CompositeCurve is a list of geometric curves such that the each geometric curve in the set terminates at the start point of the subsequent curve in the list

The **generator** is a GM_OrientableCurve.

A GM_CompositCurve is also a subclass of GM_OrientatableCurve. One of the few examples of multiple inheritance.

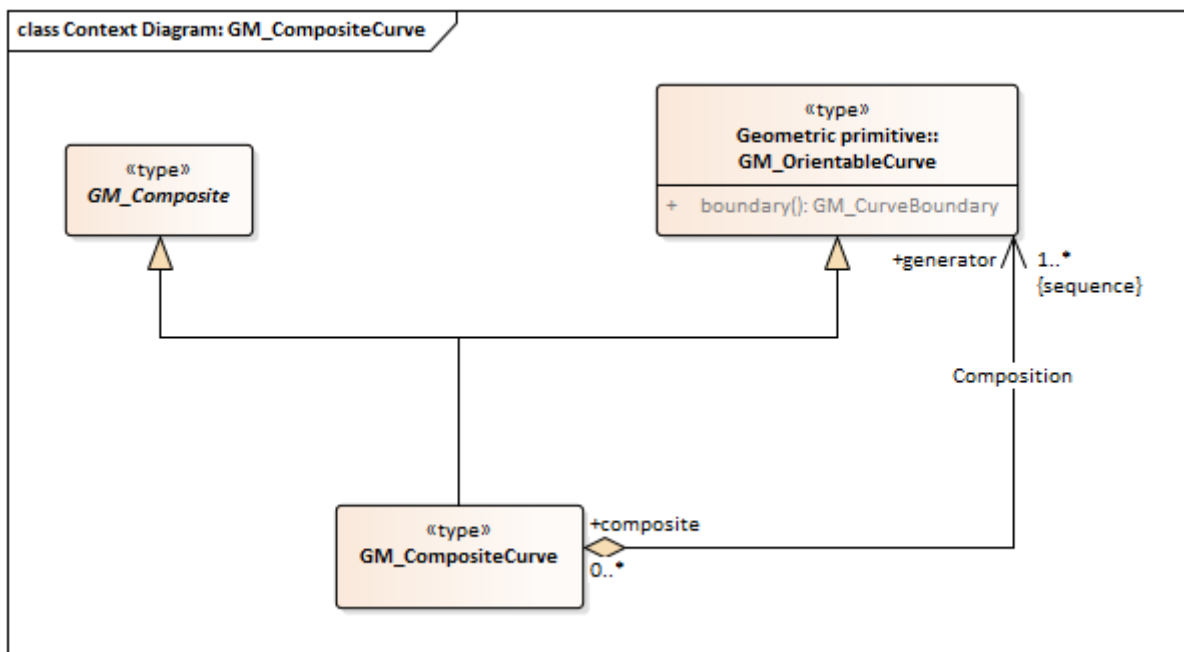


Figure 5. GM_CompositeCurve Context Diagram

GM_OrientableCurve

GM_OrientableCurve consists of a curve and an orientation inherited from GM_OrientablePrimitive. If the orientation is "+", then the GM_OrientableCurve is a GM_Curve. If the orientation is "-", then the GM_OrientableCurve is related to another GM_Curve with a parameterization that reverses the sense of the curve traversal.

```
GM_OrientableCurve:
{Orientation = "+" implies primitive = self}; +
{Orientation = "-" implies primitive.parameterization(length()-s) =
parameterization(s)};
```

GM_CompositeSolid

A GM_CompositeSolid is a set of geometric solids adjoining one another along common boundary geometric surfaces.

The **generator** is a [GM_Solid](#).

A GM_CompositSolid is also a subclass of [GM_Solid](#). One of the few examples of multiple inheritance.

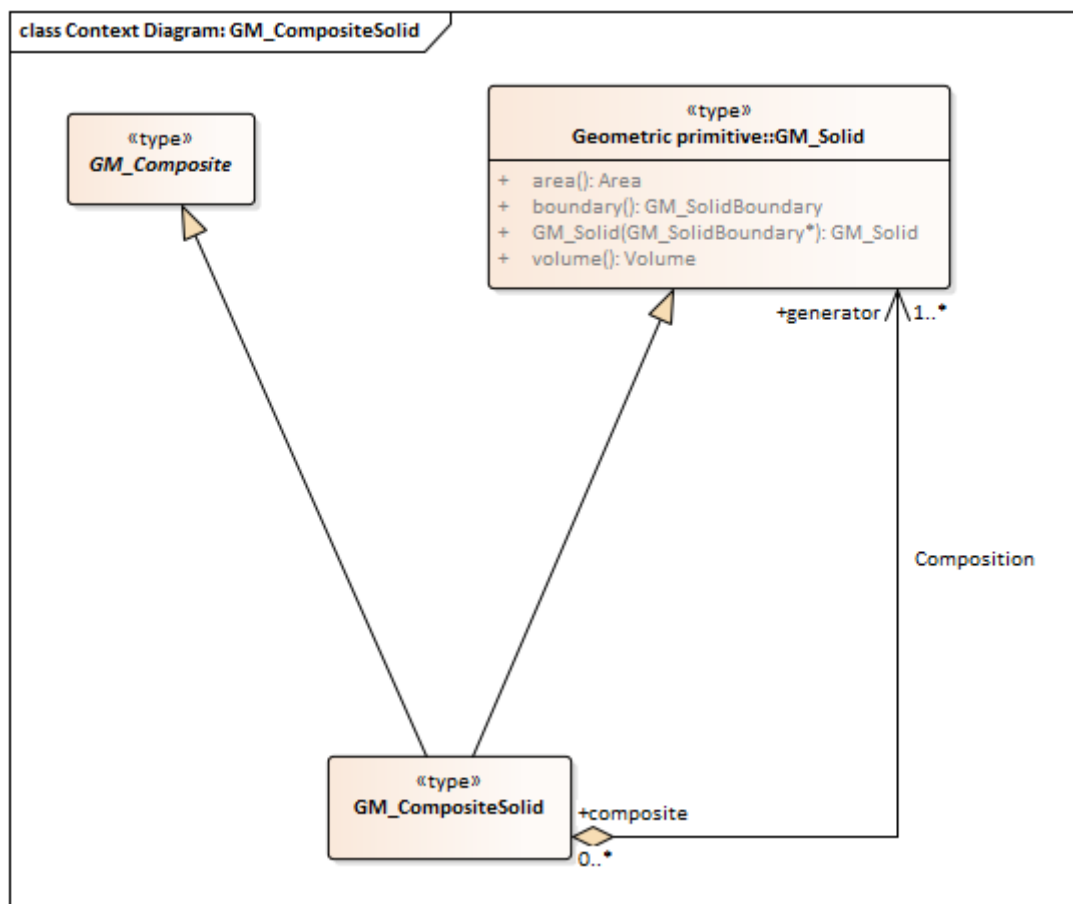


Figure 6. GM_CompositeSolid Context Diagram

8.1.3. City Models and City Objects

/// This section provides informative text in support of your standard. Most standards have a "core" module which defines the basic functionality which all implementations must support. Informative text about that "core" modeule goes here. ///

8.1.4. Space Concept

More core concepts

8.1.5. Geometry and LOD

And more concepts

8.1.6. CityGML Core UML Model

And finally the UML model.

8.2. Appearance

8.2.1. Synopsis

This is a short write-up on what this package is about.

8.2.2. Key Concepts

Appearance: An Appearance is a collection of surface data, i.e. observable properties for surface geometry objects in the form of textures and material.

A type of [AbstractAppearance](#).

NOTE	This is a smaller package so a separate "discussion" link is not necessary.
-------------	---

8.2.3. Discussion

This is the informative text that most users will be looking for. Remember that hyperlinks are important. For example, each time a class name is used you can include a link back to the definition of that class (ex. [City Object](#))

The beauty of hypermedia is that users can quickly get the additional data they may need to understand your writing.

8.2.4. UML Model

The UML model goes here.

8.2.5. Examples

Chapter 9. CityGML Data Dictionary

The CityGML UML model is the normative definition of the CityGML Conceptual Model. The Data Dictionary tables in this section were software generated from the UML model. As such, this section provides a normative representation of the CityGML Conceptual Model.

CityGML UML Packages

```
/// image::../standard/figures/CityGMLPackageDiagram.png[align="center"]
```

```
/// include::../standard/data-dictionaries/ISO-Classes.adoc[]
```

```
/// include::../standard/data-dictionaries/Core.adoc[]
```

NOTE

CityGML exported a data dictionary from its' UML model as a normative representation of the standard. It's included in the Users Guide as a quick reference.

Annex A: Revision History

Date	Release	Editor	Primary clauses modified	Description
2016-04-28	0.1	G. Editor	all	initial version

Annex B: Glossary

conformance test class

set of conformance test modules that must be applied to receive a single certificate of conformance
[OGC 08-131r3, definition 4.4]

feature

abstraction of real world phenomena
[ISO 19101-1:2014, definition 4.1.11]

feature attribute

characteristic of a feature
[ISO 19101-1:2014, definition 4.1.12]

feature type

class of features having common characteristics
[ISO 19156:2011, definition 4.7]

measurement

set of operations having the object of determining the value of a quantity
[ISO 19101-2:2018, definition 3.21] / [VIM:1993, 2.1]

model

abstraction of some aspects of reality
[ISO 19109:2015, definition 4.15]

observation

act of measuring or otherwise determining the value of a property
[ISO 19156:2011, definition 4.11]

observation procedure

method, algorithm or instrument, or system of these, which may be used in making an observation
[ISO 19156:2011, 4.12]

observation result

estimate of the value of a property determined through a known observation procedure
[ISO 19156:2011, 4.14]

property

facet or attribute of an object referenced by a name.
[ISO 19143:2010, definition 4.21]

requirements class

aggregate of all requirement modules that must all be satisfied to satisfy a conformance test class
[OGC 08-131r3, definition 4.19]

schema

formal description of a model
[ISO 19101-1:2014, definition 4.1.34]

sensor

type of observation procedure that provides the estimated value of an observed property at its output

[OGC 08-094r1, definition 4.5]

Standardization Target

TBD

timeseries

sequence of data values which are ordered in time

[OGC 15-043r3]

universe of discourse

view of the real or hypothetical world that includes everything of interest

[ISO 19101-1:2014, definition 4.1.38]

version

Particular variation of a spatial object

[INSPIRE Glossary]

B.1. ISO Concepts

The following concepts from the ISO TC211 Harmonized UML model are referenced by the CityGML Conceptual UML model but do not play a major role in its' definition. They are provided here to support a more complete understanding of the model.

Area

The measure of the physical extent of any topologically 2-D geometric object. Usually measured in "square" units of length.

[[[iso19103](#)]]

Boolean

boolean is the mathematical datatype associated with two-valued logic

[[[iso19103](#)]]

CC_CoordinateOperation

mathematical operation on coordinates that transforms or converts coordinates to another coordinate reference system.

[[[iso19111](#)]]

Character

symbol from a standard character-set.

[[[iso19103](#)]]

CharacterString

Characterstring is a family of datatypes which represent strings of symbols from standard character-sets.

[[[iso19103](#)]]

CRS

Coordinate reference system which is usually single but may be compound.

[\[\[iso19111\]\]](#)

CV_DiscreteCoverage

A subclass of CV_Coverage that returns a single record of values for any direct position within a single geometric object in its spatiotemporal domain.

[\[\[iso19123\]\]](#)

CV_DomainObject

[\[\[iso19123\]\]](#)

CV_GridPointValuePair

[\[\[iso19123\]\]](#)

CV_GridValuesMatrix

The geometry represented by the various offset vectors is in the image plane of the grid.

[\[\[iso19123\]\]](#)

CV_ReferenceableGrid

[\[\[iso19123\]\]](#)

Date

Date gives values for year, month and day. Representation of Date is specified in ISO 8601. Principles for date and time are further discussed in ISO 19108.

[\[\[iso19103\]\]](#)

DateTime

A DateTime is a combination of a date and a time types. Representation of DateTime is specified in ISO 8601. Principles for date and time are further discussed in ISO 19108.

[\[\[iso19103\]\]](#)

Distance

Used as a type for returning distances and possibly lengths.

[\[\[iso19103\]\]](#)

Engineering CRS

A contextually local coordinate reference system which can be divided into two broad categories:

1. earth-fixed systems applied to engineering activities on or near the surface of the earth;
2. CRSs on moving platforms such as road vehicles, vessels, aircraft or spacecraft.

[\[\[iso19111\]\]](#)

Generic Name

Generic Name is the abstract class for all names in a Namespace. Each instance of a GenericName is either a LocalName or a ScopedName.

[\[\[iso19103\]\]](#)

Geometry

[\[\[iso19107\]\]](#)

GM_CompositePoint

[\[\[iso19107\]\]](#)

GM_CompositeSolid

set of geometric solids adjoining one another along common boundary geometric surfaces

[\[\[iso19107\]\]](#)

GM_GenericSurface

GM_Surface and GM_SurfacePatch both represent sections of surface geometry, and therefore share a number of operation signatures. These are defined in the interface class GM_GenericSurface.

[\[\[iso19107\]\]](#)

GM_LineString

consists of sequence of line segments, each having a parameterization like the one for GM_LineSegment

[\[\[iso19107\]\]](#)

GM_MultiPrimitive

[\[\[iso19107\]\]](#)

GM_OrientableSurface

a surface and an orientation inherited from GM_OrientablePrimitive. If the orientation is "+", then the GM_OrientableSurface is a GM_Surface. If the orientation is "-", then the GM_OrientableSurface is a reference to a GM_Surface with an upNormal that reverses the direction for this GM_OrientableSurface, the sense of "the top of the surface".

[\[\[iso19107\]\]](#)

GM_PolyhedralSurface

a GM_Surface composed of polygon surfaces (GM_Polygon) connected along their common boundary curves.

[\[\[iso19107\]\]](#)

GM_Position

a union type consisting of either a DirectPosition or of a reference to a GM_Point from which a DirectPosition shall be obtained.

[\[\[iso19107\]\]](#)

GM_Primitive

The abstract root class of the geometric primitives. Its main purpose is to define the basic "boundary" operation that ties the primitives in each dimension together.

[\[\[iso19107\]\]](#)

Integer

An exact integer value, with no fractional part.

[\[\[iso19103\]\]](#)

Internet of Things

The network of physical objects--"things"--that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the Internet.

IO_IdentifiedObjectBase

[\[\[iso19103\]\]](#)

Length

The measure of distance as an integral, i.e. the limit of an infinite sum of distances between points on a curve.

[\[\[iso19103\]\]](#)

Measure

The result from performing the act or process of ascertaining the extent, dimensions, or quantity of some entity.

[\[\[iso19103\]\]](#)

Number

The base type for all number data, giving the basic algebraic operations.

[\[\[iso19103\]\]](#)

Point

GM_Point is the basic data type for a geometric object consisting of one and only one point.

[\[\[iso19107\]\]](#)

Real

The common binary Real finite implementation using base 2.

[\[\[iso19103\]\]](#)

RS_ReferenceSystem

Description of a spatial and temporal reference system used by a dataset.

[\[\[iso19111\]\]](#)

Scoped Name

ScopedName is a composite of a LocalName for locating another NameSpace and a GenericName valid in that NameSpace. ScopedName contains a LocalName as head and a GenericName, which might be a LocalName or a ScopedName, as tail.

[\[\[iso19103\]\]](#)

Solid

GM_Solid, a subclass of GM_Primitive, is the basis for 3-dimensional geometry. The extent of a solid is defined by the boundary surfaces.

[\[\[iso19107\]\]](#)

Time

Time is the designation of an instant on a selected time scale, astronomical or atomic. It is used in the sense of time of day.

[\[\[iso19103\]\]](#)

TM_Duration

[\[\[iso19108\]\]](#)

TM_TemporalPosition

The position of a TM_Instant relative to a TM_ReferenceSystem.

[\[\[iso19108\]\]](#)

Unit of Measure

Any of the systems devised to measure some physical quantity such distance or area or a system devised to measure such things as the passage of time.

[\[\[iso19103\]\]](#)

URI

Uniform Resource Identifier (URI), is a compact string of characters used to identify or name a resource

[\[\[iso19103\]\]](#)

Volume

Volume is the measure of the physical space of any 3-D geometric object.

[\[\[iso19103\]\]](#)

B.2. Abbreviated Terms

- 2D Two Dimensional
- 3D Three Dimensional
- AEC Architecture, Engineering, Construction
- ALKIS German National Standard for Cadastral Information
- ATKIS German National Standard for Topographic and Cartographic Information
- BIM Building Information Modeling
- B-Rep Boundary Representation
- bSI buildingSMART International
- CAD Computer Aided Design
- COLLADA Collaborative Design Activity
- CSG Constructive Solid Geometry
- DTM Digital Terrain Model
- DXF Drawing Exchange Format
- EuroSDR European Spatial Data Research Organisation
- ESRI Environmental Systems Research Institute
- FM Facility Management
- GDF Geographic Data Files
- GDI-DE Spatial Data Infrastructure Germany (Geodateninfrastruktur Deutschland)
- GDI NRW Geodata Infrastructure North-Rhine Westphalia
- GML Geography Markup Language

- IAI International Alliance for Interoperability (now buildingSMART International (bSI))
- IETF Internet Engineering Task Force
- IFC Industry Foundation Classes
- IoT Internet of Things
- ISO International Organization for Standardisation
- ISO/TC211 ISO Technical Committee 211
- LOD Levels of Detail
- MQTT
- NBIMS National Building Information Model Standard
- OASIS Organisation for the Advancement of Structured Information Standards
- OGC Open Geospatial Consortium
- OSCRE Open Standards Consortium for Real Estate
- SIG 3D Special Interest Group 3D of the GDI-DE
- TIC Terrain Intersection Curve
- TIN Triangulated Irregular Network
- UML Unified Modeling Language
- URI Uniform Resource Identifier
- VRML Virtual Reality Modeling Language
- W3C World Wide Web Consortium
- W3DS OGC Web 3D Service
- WFS OGC Web Feature Service
- X3D Open Standards XML-enabled 3D file format of the Web 3D Consortium
- XML Extensible Markup Language
- xAL OASIS extensible Address Language

Annex C: Bibliography

Example Bibliography (Delete this note).

The TC has approved Springer LNCS as the official document citation type.

Springer LNCS is widely used in technical and computer science journals and other publications

NOTE

- For citations in the text please use square brackets and consecutive numbers:
[1], [2], [3]

– Actual References:

[n] Journal: Author Surname, A.: Title. Publication Title. Volume number, Issue number, Pages Used (Year Published)

[n] Web: Author Surname, A.: Title, <http://Website-Url>

[1] OGC: OGC Testbed 12 Annex B: Architecture. (2015).