
**Geographic information — Geography
Markup Language (GML) —**

**Part 2:
Extended schemas and encoding rules**

*Information géographique — Langage de balisage en géographie
(GML) —*

Partie 2: Schémas étendus et règles d'encodage



STANDARDSISO.COM : Click to view the full PDF of ISO 19136-2:2015



COPYRIGHT PROTECTED DOCUMENT

© ISO 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, 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
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword	vi
Introduction	vii
1 Scope	1
2 Conformance	1
3 Normative references	2
4 Terms, definitions, symbols and abbreviated terms	2
4.1 General	2
4.2 Terms and definitions	2
4.3 Symbols and abbreviated terms	3
5 Conventions	3
5.1 MIME media types	3
5.2 XML namespaces	3
5.3 Deprecated parts of previous versions of GML	4
6 Additional base types	5
6.1 Target namespace	5
6.2 Localisable strings	5
6.2.1 LanguageStringType	5
6.2.2 Additional types based on LanguageStringType	5
6.3 TimePositionUnion	5
6.4 Requirements class	7
6.5 Conformance	7
7 Compact Encodings of Commonly Used GML Geometries	8
7.1 Target namespace	8
7.2 Introduction	8
7.3 SimplePolygon	8
7.4 SimpleRectangle	9
7.5 SimpleTriangle	10
7.6 SimpleArcString	10
7.7 SimpleArc	11
7.8 SimpleArcByCenterPoint	11
7.9 SimpleArcStringByBulge	12
7.10 SimpleArcByBulge	12
7.11 SimpleCircle	13
7.12 SimpleCircleByCenterPoint	13
7.13 SimpleMultiPoint	14
7.14 MultiPointPropertyType	14
7.15 Requirements class	14
7.16 Conformance	14
8 Triangulated Irregular Networks	15
8.1 Target namespace	15
8.2 Introduction	15
8.3 TriangulatedSurface	15
8.4 SimpleTrianglePatch	15
8.5 TIN	16
8.6 TINElement	16
8.7 TINElementPropertyType	17
8.8 TINElementType	17
8.9 Requirements class	19
8.10 Conformance	20
9 Linear Referencing	20
9.1 Target namespaces	20

9.2	Introduction	20
9.3	Basic Linear Referencing	21
9.3.1	Target namespace	21
9.3.2	Introduction	21
9.3.3	PositionExpression	21
9.3.4	PositionExpressionPropertyType	21
9.3.5	LinearElement	22
9.3.6	LinearElementPropertyType	23
9.3.7	StartValueType	23
9.3.8	LinearReferencingMethod	23
9.3.9	LinearReferencingMethodPropertyType	24
9.3.10	DistanceExpressionType	24
9.3.11	DistanceExpressionPropertyType	25
9.3.12	AlongReferent	25
9.3.13	AlongReferentPropertyType	25
9.3.14	Referent	26
9.3.15	ReferentPropertyType	27
9.3.16	MeasureType	27
9.3.17	LRMNameType	27
9.3.18	LRMTypeType	31
9.3.19	ReferentTypeType	32
9.3.20	LinearSRS	33
9.3.21	LinearSRSPROPERTYType	33
9.4	Linear Referencing Towards Referent	34
9.4.1	Target namespace	34
9.4.2	Introduction	34
9.4.3	DualAlongReferent	34
9.4.4	DualAlongReferentPropertyType	34
9.5	Linear Referencing Offset	35
9.5.1	Target namespace	35
9.5.2	Introduction	35
9.5.3	LRMWithOffset	35
9.5.4	LRMWithOffsetPropertyType	35
9.5.5	LateralOffsetDistanceExpressionType	36
9.5.6	LateralOffsetExpressionType	36
9.5.7	VerticalOffsetExpressionType	37
9.5.8	LateralOffsetDirectionType	38
9.5.9	VerticalOffsetDirectionType	39
9.5.10	LateralOffsetLinearSRS	39
9.5.11	LateralOffsetLinearSRSPROPERTYType	40
9.6	Linear Referencing Offset Vectors	41
9.6.1	Target namespace	41
9.6.2	Introduction	41
9.6.3	VectorOffsetDistanceExpressionType	41
9.6.4	VectorOffsetExpressionType	41
9.6.5	VectorOffsetLinearSRS	42
9.6.6	VectorOffsetLinearSRSPROPERTYType	45
9.7	Requirements classes	45
9.8	Conformance	47
10	ReferenceableGrid	48
10.1	Target namespace	48
10.2	Clarifications to ISO 19123:2005	48
10.3	AbstractReferenceableGrid	49
10.4	ReferenceableGridByArray	50
10.5	ReferenceableGridByVectorsType, ReferenceableGridByVectors	51
10.6	ReferenceableGridByTransformation	57
10.7	gridCRS	58
10.8	Coverages using ReferenceableGrid	58

10.9	Requirements classes.....	58
10.10	Conformance.....	59
11	Code lists, dictionaries and definitions.....	60
11.1	Overview	60
11.2	Requirements class	61
11.3	Conformance.....	62
12	Encoding rule.....	62
12.1	Target namespace.....	62
12.2	Improved conversion rule.....	62
12.2.1	Conversion rule changes.....	62
12.2.2	Requirements class.....	64
12.2.3	Conformance.....	65
12.3	Association class conversion rule.....	67
12.3.1	Overview.....	67
12.3.2	Requirements class.....	68
12.3.3	Conformance.....	69
12.4	Encoding rule extensions.....	70
12.4.1	Overview.....	70
12.4.2	gmlexr:extendedEncodingRule	70
12.4.3	Requirements class.....	70
12.4.4	Conformance class.....	71
Annex A	(informative) Linear referencing method examples.....	73
Bibliography		78

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 documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2. www.iso.org/directives

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received. www.iso.org/patents

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

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: [Foreword - Supplementary information](#).

The Geography Markup Language (GML) was originally developed within the Open Geospatial Consortium (OGC). The Committee responsible for this document is ISO/TC 211, *Geographic information/Geomatics*.

ISO 19136 consists of the following parts, under the general title *Geographic Information — Geography Markup Language*:

— *Part 2: Extended schemas and encoding rules*

A future Part 1 will revise and replace the currently available ISO 19136:2007.

Introduction

Geography Markup Language is an XML grammar written in XML Schema for the description of application schemas as well as the transport and storage of geographic information.

The key concepts used by Geography Markup Language (GML) to model the world are drawn from the ISO 19100- series of International Standards and the OpenGIS Abstract Specification.

A feature is an “abstraction of real world phenomena” (ISO 19101); it is a geographic feature if it is associated with a location relative to the Earth. So a digital representation of the real world may be thought of as a set of features. The state of a feature is defined by a set of properties, where each property may be thought of as a {name, type, value} triple.

The number of properties a feature may have, together with their names and types, is determined by its type definition. Geographic features with geometry are those with properties that may be geometry-valued. A feature collection is a collection of features that may itself be regarded as a feature; as a consequence a feature collection has a feature type and thus may have distinct properties of its own, in addition to the features it contains.

Following ISO 19109, the feature types of an application or application domain is usually captured in an application schema. A GML application schema is specified in XML Schema and can be constructed in two different and alternative ways:

- by adhering to the rules specified in ISO 19109 for application schemas in UML, and conforming to both the constraints on such schemas and the rules for mapping them to GML application schemas specified in this part of ISO 19136;
- by adhering to the rules for GML application schemas specified in this part of ISO 19136 for creating a GML application schema directly in XML Schema.

Both ways are supported by this part of ISO 19136. To ensure proper use of the conceptual modelling framework of the ISO 19100- series of International Standards, all application schemas are expected to be modelled in accordance with the General Feature Model as specified in ISO 19109. Within the ISO 19100- series, UML is the preferred language by which to model conceptual schemas.

GML specifies XML encodings, conformant with ISO 19118, of several of the conceptual classes defined in the ISO 19100- series of International Standards and the OpenGIS Abstract Specification. These conceptual models include those defined in:

- ISO/TS 19103, *Geographic information — Conceptual schema language* (units of measure, basic types);
- ISO 19107, *Geographic information — Spatial schema* (geometry and topology objects);
- ISO 19108, *Geographic information — Temporal schema* (temporal geometry and topology objects, temporal reference systems);
- ISO 19109, *Geographic information — Rules for application schemas* (features);
- ISO 19111, *Geographic information — Spatial referencing by coordinates* (coordinate reference systems);
- ISO 19123, *Geographic information — Schema for coverage geometry and functions*;
- ISO 19148, *Geographic information — Linear referencing*.

The aim is to provide a standardized encoding (i.e. a standardized implementation in XML) of types specified in the conceptual models specified by the International Standards listed above. If every application schema were encoded independently and the encoding process included the types from, for example, ISO 19108, then, without unambiguous and completely fixed encoding rules, the XML encodings would be different. Also, since every implementation platform has specific strengths and weaknesses, it is helpful to standardize XML encodings for core geographic information concepts modelled in the ISO 19100- series of International Standards and commonly used in application schemas.

In many cases, the mapping from the conceptual classes is straightforward, while in some cases the mapping is more complex (a detailed description of the mapping is part of this part of ISO 19136).

In addition, GML provides XML encodings for additional concepts not yet modelled in the ISO 19100-series of International Standards or the OpenGIS Abstract Specification, for example, dynamic features, simple observations or value objects.

Predefined types of geographic feature in GML include coverages and simple observations.

A coverage is a subtype of feature that has a coverage function with a spatiotemporal domain and a value set range of homogeneous 1- to n -dimensional tuples. A coverage may represent one feature or a collection of features “to model and make visible spatial relationships between, and the spatial distribution of, Earth phenomena” (OGC Abstract Specification Topic 6) and a coverage “acts as a function to return values from its range for any direct position within its spatiotemporal domain” (ISO 19123).

An observation models the act of observing, often with a camera or some other procedure, a person or some form of instrument (Merriam-Webster Dictionary: “an act of recognizing and noting a fact or occurrence often involving measurement with instruments”). An observation is considered to be a GML feature with a time at which the observation took place, and with a value for the observation.

A reference system provides a scale of measurement for assigning values to a position, time or other descriptive quantity or quality.

A coordinate reference system consists of a set of coordinate system axes that is related to the Earth through a datum that defines the size and shape of the Earth.

A temporal reference system provides standard units for measuring time and describing temporal length or duration.

A reference system dictionary provides definitions of reference systems used in spatial or temporal geometries.

Spatial geometries are the values of spatial feature properties. They indicate the coordinate reference system in which their measurements have been made. The “parent” geometry element of a geometric complex or geometric aggregate makes this indication for its constituent geometries.

Temporal geometries are the values of temporal feature properties. Like their spatial counterparts, temporal geometries indicate the temporal reference system in which their measurements have been made.

Spatial or temporal topologies are used to express the different topological relationships between features.

A units-of-measure dictionary provides definitions of numerical measures of physical quantities, such as length, temperature and pressure, and of conversions between units.

Geographic information — Geography Markup Language (GML) —

Part 2: Extended schemas and encoding rules

1 Scope

The Geography Markup Language (GML) is an XML encoding in compliance with ISO 19118 for the transport and storage of geographic information modelled in accordance with the conceptual modelling framework used in the ISO 19100- series of International Standards and including both the spatial and non-spatial properties of geographic features.

This part of ISO 19136 defines the XML Schema syntax, mechanisms and conventions that:

- provide an open, vendor-neutral framework for the description of geospatial application schemas for the transport and storage of geographic information in XML;
- allow profiles that support proper subsets of GML framework descriptive capabilities;
- support the description of geospatial application schemas for specialized domains and information communities;
- enable the creation and maintenance of linked geographic application schemas and datasets;
- support the storage and transport of application schemas and datasets;
- increase the ability of organizations to share geographic application schemas and the information they describe.

Implementers may decide to store geographic application schemas and information in GML, or they may decide to convert from some other storage format on demand and use GML only for schema and data transport.

This part of ISO 19136 builds on ISO 19136:2007 (GML 3.2), and extends it with additional schema components and requirements.

NOTE If an ISO 19109 conformant application schema described in UML is used as the basis for the storage and transportation of geographic information, this part of ISO 19136 provides normative rules for the mapping of such an application schema to a GML application schema in XML Schema and, as such, to an XML encoding for data with a logical structure in accordance with the ISO 19109 conformant application schema.

2 Conformance

This part of ISO 19136 defines XML implementations of concepts used in spatiotemporal datasets. It extends the XML implementations specified in ISO 19136:2007 (GML 3.2). Requirements and conformance classes specified in ISO 19136:2007 also apply for this part of ISO 19136.

XML instances that encode geographic information using one or more of the schemas specified in this part of ISO 19136 are the standardization target of the requirements stated in this part of ISO 19136.

The implementation is described using the XML Schema language and Schematron.

Conformance classes are specified in [Clauses 6](#) to [12](#) of this part of ISO 19136.

3 Normative references

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

ISO 19136:2007, *Geographic information — Geography Markup Language (GML)*

ISO 19148:2012, *Geographic information — Linear referencing*

OGC Technical Committee Policies and Procedures: *MIME Media Types for GML*¹⁾

ISO 8601:2004, *Data elements and interchange formats — Information interchange — Representation of dates and times*

ISO/IEC 13249-3:2011, *Information technology — Database languages — SQL multimedia and application packages — Part 3: Spatial*

4 Terms, definitions, symbols and abbreviated terms

4.1 General

For the purposes of this document, the terms, definitions, symbols and abbreviated terms listed in ISO 19136:2007 (GML 3.2), [Clause 4](#), apply.

4.2 Terms and definitions

In addition to the terms listed in ISO 19136:2007 (GML 3.2), the following terms and definitions apply.

4.2.1

grid coordinate reference system

grid CRS

coordinate reference system for the positions in a grid that uses a defined coordinate system congruent with the coordinate system described by the GridEnvelope and axisLabels of gml:GridType

Note 1 to entry: A grid CRS uses a defined coordinate system with the same grid point positions and origin as the GridEnvelope, with the same axisLabels, but need not define any limits on the grid size. This coordinate system is sometimes called the internal grid coordinate system.

4.2.2

referenceable grid

grid associated with a transformation that can be used to convert grid coordinate values to values of coordinates referenced to 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 georeferenceable grid.

[SOURCE: ISO 19123:2005, 4.1.33]

1) The latest version at the publication of this part of ISO 19136 is OGC document 09-144r1. The MIME type is currently in the registration process at IETF / IANA. The reference has intentionally been undated so that the reference is to the latest version in case changes to the MIME media type specification are required as part of the registration process.

4.3 Symbols and abbreviated terms

In addition to the symbols and abbreviated terms listed in ISO 19136:2007 (GML 3.2), the following are used in this document:

LRS	Linear Referencing System
OWL	Web Ontology Language
OWS	OGC Web Services
SKOS	Simple Knowledge Organization System

5 Conventions

5.1 MIME media types

For exchanging GML instance documents over the internet, the media type “application/gml+xml” is used as specified by the OGC Technical Committee Policies and Procedures: MIME Media Types for GML.

5.2 XML namespaces

The XML namespaces used within this part of ISO 19136 are listed in [Table 1](#). For each namespace, the namespace prefix used within this document and the canonical location of the all-components schema document are provided, too.

Table 1 — XML Namespaces

XML Namespace	Name-space prefix	Canonical location of all-components schema document
http://www.opengis.net/gml/3.2	gml	http://schemas.opengis.net/gml/3.2.1/gml.xsd
http://www.opengis.net/gml/3.3/xbt	gmlxbt	http://schemas.opengis.net/gml/3.3/extdBaseTypes.xsd
http://www.opengis.net/gml/3.3/ce	gmlce	http://schemas.opengis.net/gml/3.3/geometryCompact.xsd
http://www.opengis.net/gml/3.3/tin	gmltin	http://schemas.opengis.net/gml/3.3/tin.xsd
http://www.opengis.net/gml/3.3/lr	gmllr	http://schemas.opengis.net/gml/3.3/linearRef.xsd
http://www.opengis.net/gml/3.3/lrtr	gmlrtr	http://schemas.opengis.net/gml/3.3/linearRefTowardsReferent.xsd
http://www.opengis.net/gml/3.3/lro	gmlro	http://schemas.opengis.net/gml/3.3/linearRefOffset.xsd
http://www.opengis.net/gml/3.3/lrov	gmlrov	http://schemas.opengis.net/gml/3.3/linearRefOffsetVector.xsd
http://www.opengis.net/gml/3.3/rgrid	gmlr-grid	http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd
http://www.opengis.net/gml/3.3/xer	gmlexr	http://schemas.opengis.net/gml/3.3/extdEncRule.xsd
http://www.w3.org/1999/xlink	xlink	http://www.w3.org/1999/xlink.xsd
http://www.w3.org/XML/1998/namespace	xml	http://www.w3.org/2001/xml.xsd
http://www.w3.org/2001/XMLSchema	xs (or default)	n/a

NOTE 1 A GML application schema conforming to this part of ISO 19136 will import the GML 3.2 schema plus zero or more additional GML 3.3 schemas as needed.

EXAMPLE 1 The following GML application schema imports both the GML 3.2 schema and the GML 3.3 compact geometry encoding.

```
<schema
  targetNamespace="http://www.example.org/app"
  xmlns:app="http://www.example.org/app"
  xmlns:gmlce="http://www.opengis.net/gml/3.3/ce"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified"
  version="1.0.0">
  <import namespace="http://www.opengis.net/gml/3.2"
    schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
  <import namespace="http://www.opengis.net/gml/3.3/ce"
    schemaLocation="http://schemas.opengis.net/gml/3.3/geometryCompact.xsd"/>
  <!-- ... -->
  <element name="Parcel" substitutionGroup="gml:AbstractFeature"
    type="app:ParcelType"/>
  <complexType name="ParcelType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="geometry" type="gml:SurfacePropertyType"/>
          <!-- ... -->
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

NOTE 2 A GML instance conforming to this part of ISO 19136 will directly or indirectly reference the GML 3.2 schema and zero or more additional GML 3.3 schemas as needed. A schema location attribute in the root element of the document has to include the schema of the namespace of the root element.

EXAMPLE 2 A sample instance for the GML application schema from example 1 as returned from a Web Feature Service. The GML schemas are imported by the GML application schema that is referenced from the instance document:

```
<wfs:FeatureCollection
  timeStamp="2011-04-03T05:40:00Z"
  numberMatched="12"
  numberReturned="12"
  xmlns:wfs="http://www.opengis.net/wfs/2.0"
  xmlns:app="http://www.example.org/app"
  xmlns:gmlce="http://www.opengis.net/gml/3.3/ce"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.example.org/app http://www.example.org/app.xsd
    http://www.opengis.net/wfs/2.0 http://schemas.opengis.net/wfs/2.0/wfs.xsd">
  <wfs:member>
    <app:Parcel gml:id="ol">
      <app:geometry>
        <gmlce:SimplePolygon gml:id="g1"
          srsName="http://www.opengis.net/def/crs/EPSG/0/4258">
          <gml:posList>50 6 50 7 51 7 51 6</gml:posList>
        </gmlce:SimplePolygon>
      </app:geometry>
      <!-- ... -->
    </app:Parcel>
  </wfs:member>
  <!-- ... -->
</wfs:FeatureCollection>
```

5.3 Deprecated parts of previous versions of GML

The verb **“deprecate”** provides notice that the referenced portion is being retained for backwards compatibility with earlier versions but may be removed from or superseded in this or a future version.

6 Additional base types

6.1 Target namespace

All schema components specified in [Clause 6](#) are in the target namespace:

<http://www.opengis.net/gml/3.3/xbt>

6.2 Localisable strings

6.2.1 LanguageStringType

The type `gmlxbt:LanguageStringType` adds an optional `xml:lang` attribute to `xs:string`. It is provided as a base type for linguistic text for use within the GML schemas and in GML application schemas.

```
<complexType name="LanguageStringType">
  <simpleContent>
    <extension base="xs:string">
      <attribute ref="xml:lang"/>
    </extension>
  </simpleContent>
</complexType>
```

NOTE The name "LanguageStringType" was selected to align the naming with the equivalent type in the OWS Common standard.

6.2.2 Additional types based on LanguageStringType

The following types from GML 3.2 are defined in the <http://www.opengis.net/gml/3.3/xbt> schema with the same content model as in GML 3.2 with the exception that `gmlxbt:LanguageStringType` is used in place of `xs:string`:

- `gml:CodeType`,
- `gml:CodeWithAuthorityType`

In addition, the global property elements `gmlxbt:description` and `gmlxbt:remarks` have an optional `xml:lang` attribute in their content model. These elements are in the substitution group of the GML 3.2 property element with the same local name.

- `gmlxbt:description`
- `gmlxbt:remarks`.

6.3 TimePositionUnion

In GML 3.2 the simple type `gml:TimePositionUnion` is a union of XML Schema simple types which instantiate the subtypes for temporal position described in ISO 19108. These are:

- `gml:CalDate` (union of `xs:date`, `xs:gYearMonth` and `xs:gYear`)
- `xs:time`
- `xs:dateTime`
- `xs:anyURI`
- `xs:decimal`

ISO 8601:2004, 4.1.2 specifies the Calendar Date and its representations with reduced accuracy (i.e., YYYY-MM and YYYY). `gml:CalDate` is designed to support their encoding.

ISO 8601:2004, 4.1.3 specifies the Ordinal Date, which is composed from the calendar year and the calendar day of the year (YYYY-DDD).

ISO 8601:2004, 4.1.4 specifies the Week Date, which is composed from the calendar year, the calendar week and the calendar day of the week (YYYY-Www-D). ISO 8601:2004, 4.1.4.3 specifies a Week Date representation with reduced accuracy that omits the day of the week component (YYYY-Www).

The ISO 8601:2004 Ordinal Date and Week Date with reduced accuracy are commonly used in some communities (for example, aviation) but are not supported by `gml:TimePositionUnion`. `gmlxht:TimePositionUnion` is provided to accommodate representations of these sibling date-representations from ISO 8601:2004 by adding `gmlxht:OrdDate` and `gmlxht:WeekDate` to the union.

```
<simpleType name="TimePositionUnion">
  <union memberTypes="gml:CalDate gmlxht:OrdDate gmlxht:WeekDate time dateTime anyURI
decimal"/>
</simpleType>

<simpleType name="OrdDate">
  <restriction base="string">
    <pattern value="-?[0-9]{4}-[0-9]{3}"/>
  </restriction>
</simpleType>

<simpleType name="WeekDate">
  <restriction base="string">
    <pattern value="-?[0-9]{4}-W(0[1-9]|[1-4][0-9]|5[0-3])(-[1-7])?"/>
  </restriction>
</simpleType>
```

The lexical space of `gmlxht:OrdDate` consists of finite-length sequences of characters of the form

'-?'yyyy '-' ddd

where:

- '-?'yyyy is a four-or-more digit optionally negative-signed numeral that represents the year; if more than four digits, leading zeros are prohibited, and '0000' is prohibited (note that a plus sign is not permitted);
- the '-' is a separator between parts of the date;
- the ddd is a three-digit numeral that represents the calendar day of the year where the first calendar day of any calendar year is represented by '001' and subsequent calendar days are numbered in ascending sequence.

The lexical space of `gmlxht:WeekDate` consists of finite-length sequences of characters of the form

'-?'yyyy '-W' ww ['- d)?

where:

- '-?'yyyy is a four-or-more digit optionally negative-signed numeral that represents the year; if more than four digits, leading zeros are prohibited, and '0000' is prohibited (note that a plus sign is not permitted);
- the '-W' is a separator indicating that week-of-year follows;
- the ww is a two-digit numeral that represents the calendar week of the year where the first calendar week of any calendar year is represented by '01' and subsequent calendar weeks are numbered in ascending sequence;
- the '-' d (if present) is a one-digit numeral that represents the calendar day of the week where Monday shall be identified as calendar day '1' of any calendar week, and subsequent calendar days of the same calendar week shall be numbered in ascending sequence to Sunday (calendar day '7').

6.4 Requirements class

Table 2 specifies the requirement class “Basic types (extensions)”.

Table 2 — Requirements class “Basic types (extensions)”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/xbt	
Target type	Data instance
Name	Basic types (extensions)
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/xbt/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/xbt schema SHALL be well-formed and valid.
Requirement	http://www.opengis.net/spec/GML/3.3/req/xbt/iso8601 Any XML node whose content model is specified using <code>gmlxht:TimePositionUnit</code> , <code>gmlx-bt:OrdDate</code> or <code>gmlxht:WeekDate</code> SHALL conform to ISO 8601:2004.

6.5 Conformance

Table 3 specifies the conformance class “Basic types (extensions)”.

Table 3 — Conformance class “Basic types (extensions)”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/xbt	
Requirements	http://www.opengis.net/spec/GML/3.3/req/xbt
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Test	http://www.opengis.net/spec/GML/3.3/conf/xbt/valid
	Requirement http://www.opengis.net/spec/GML/3.3/req/xbt/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/xbt schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/extdBasicTypes.xsd . Pass if no errors reported. Fail otherwise.
	Test type Basic
Test	http://www.opengis.net/spec/GML/3.3/conf/xbt/iso8601
	Requirement http://www.opengis.net/spec/GML/3.3/req/xbt/iso8601
	Test purpose Verify that any XML node whose content model is specified using <code>gmlxht:TimePositionUnit</code> conforms to ISO 8601:2004.
	Test method Validate the XML document using the Schematron document http://schemas.opengis.net/gml/3.3/iso8601.sch . Replace “{nodesOfTypeTimePositionUnion}” in the file with an Xpath expression to all nodes with a content model of <code>gmlxht:TimePositionUnit</code> . Pass if no errors reported. Fail otherwise.
	Test type Capability

7 Compact Encodings of Commonly Used GML Geometries

7.1 Target namespace

All schema components specified in [Clause 7](#) are in the target namespace:

<http://www.opengis.net/gml/3.3/ce>

7.2 Introduction

The simplified geometry encodings presented in [Clause 7](#) conform to the spatial schema (ISO 19107), but are offered as compact encoding alternatives to some of the existing GML geometries specified in GML 3.2. The compact encodings follow the idea of the compact `gml:LineString` encoding, which represents a curve that is equivalent to the `gml:Curve` encoding restricted to the use of `gml:LineStringSegment` elements as depicted in [Figure 1](#).

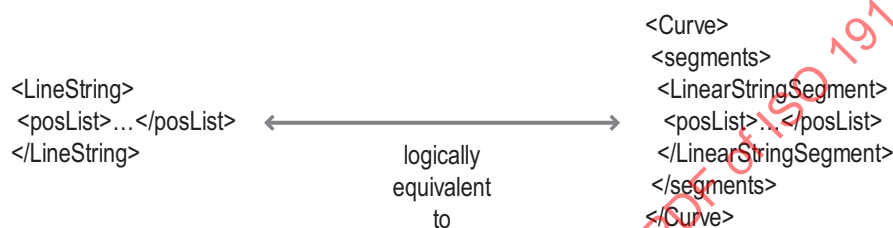


Figure 1 — LineString encoding

7.3 SimplePolygon

A `gmlce:SimplePolygon` is a specialized polygon that has a simplified encoding of the logically equivalent `gml:Surface` with a single `gml:PolygonPatch` as its surface patch consisting of a single `gml:LinearRing` as its exterior boundary and does not have any interior boundary. See [Figure 2](#).

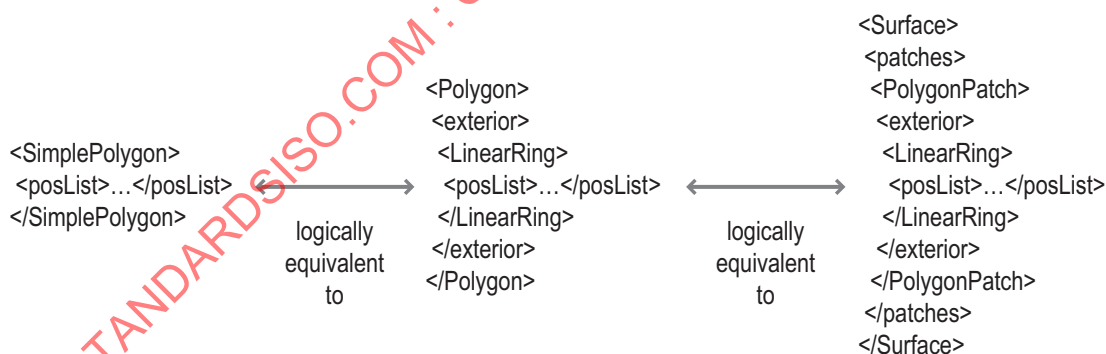


Figure 2 — SimplePolygon encoding

NOTE The boundary of a `gmlce:SimplePolygon` is coplanar and the polygon uses planar interpolation in its interior.

The usage of the term 'simple' here refers to a specialized polygon with a simplified encoding, which is *simply* connected (no interior rings) and uses a simple closed curve (no self-crossings) to represent its single boundary ring. The schema element declaration and corresponding type definition is as follows:

```
<element name="SimplePolygon"
  type="gmlce:SimplePolygonType"
  substitutionGroup="gmlce:AbstractSimplePolygon"/>
```

```

<element name="AbstractSimplePolygon" type="gml:AbstractSurfaceType"
  substitutionGroup="gml:AbstractSurface"/>

<complexType name="SimplePolygonType">
  <complexContent>
    <extension base="gml:AbstractSurfaceType">
      <sequence>
        <choice>
          <choice minOccurs="3" maxOccurs="unbounded">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

The last coordinate does not have to repeat the first coordinate in this simplified encoding, so only three control points are required to specify a simple polygon. For this reason, the inner choice declaration above has the corresponding occurrence constraint `minOccurs="3"`.

7.4 SimpleRectangle

A `gmlce:SimpleRectangle` is the special case of a simple polygon, which has exactly 4 control points in its boundary encoding representing the 4 corners of the rectangle. See [Figure 3](#).

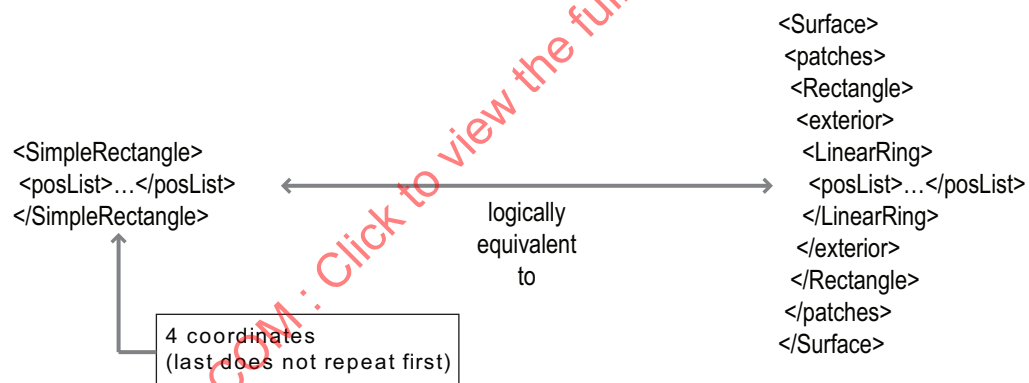


Figure 3 — SimpleRectangle encoding

NOTE The boundary of a `gmlce:SimpleRectangle` is coplanar and the polygon uses planar interpolation in its interior.

The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleRectangle" type="gmlce:SimpleRectangleType"
  substitutionGroup="gmlce:AbstractSimplePolygon"/>

<complexType name="SimpleRectangleType">
  <complexContent>
    <extension base="gml:AbstractSurfaceType">
      <sequence>
        <choice>
          <choice minOccurs="4" maxOccurs="4">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

```

</complexContent>
</complexType>

```

7.5 SimpleTriangle

A `gmlce:SimpleTriangle` is the special case of a simple polygon, which has exactly 3 control points in its boundary encoding representing the 3 corners of the triangle. See [Figure 4](#).

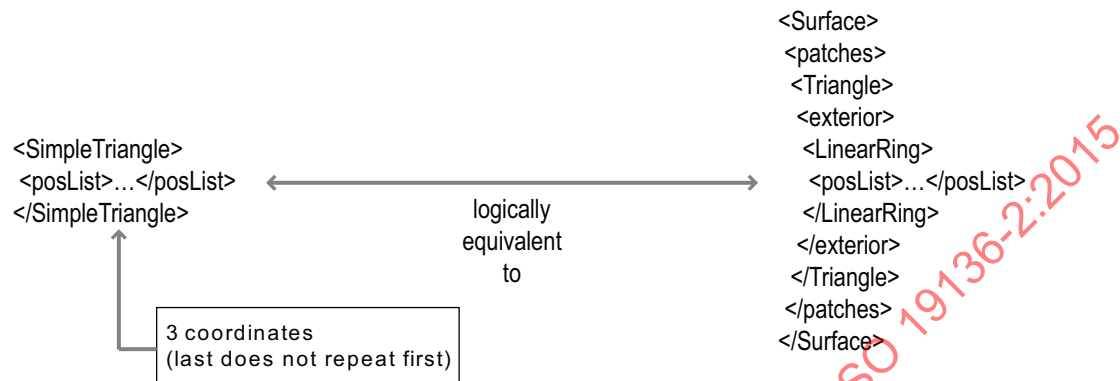


Figure 4 — SimpleTriangle encoding

The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleTriangle" type="gmlce:SimpleTriangleType"
substitutionGroup="gmlce:AbstractSimplePolygon"/>

```

```

<complexType name="SimpleTriangleType">
  <complexContent>
    <extension base="gml:AbstractSurfaceType">
      <sequence>
        <choice>
          <choice minOccurs="3" maxOccurs="3">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

EXAMPLE

```

<gmlce:SimpleTriangle gml:id="ID000"
  srsName="http://www.opengis.net/def/crs/EPSSG/0/4326">
  <gml:posList>50 10 49 10 49.5 11</gml:posList>
</gmlce:SimpleTriangle>

```

7.6 SimpleArcString

A `gmlce:SimpleArcString` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with one or more `gml:Arc` segments. The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleArcString" type="gmlce:SimpleArcStringType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

<element name="AbstractSimpleArcString" type="gml:AbstractCurveType"
  substitutionGroup="gml:AbstractCurve"/>

<complexType name="SimpleArcStringType">
  <complexContent>
    <extension base="gml:AbstractCurveType">

```

```

<sequence>
  <choice>
    <choice minOccurs="3" maxOccurs="unbounded">
      <element ref="gml:pos"/>
      <element ref="gml:pointProperty"/>
    </choice>
    <element ref="gml:posList"/>
  </choice>
</sequence>
<attribute name="interpolation" type="gml:CurveInterpolationType"
fixed="circularArc3Points"/>
<attribute name="numArc" type="integer"/>
</extension>
</complexContent>
</complexType>

```

7.7 SimpleArc

A `gmlce:SimpleArc` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with a single `gml:ArcString` segment. See [Figure 5](#).

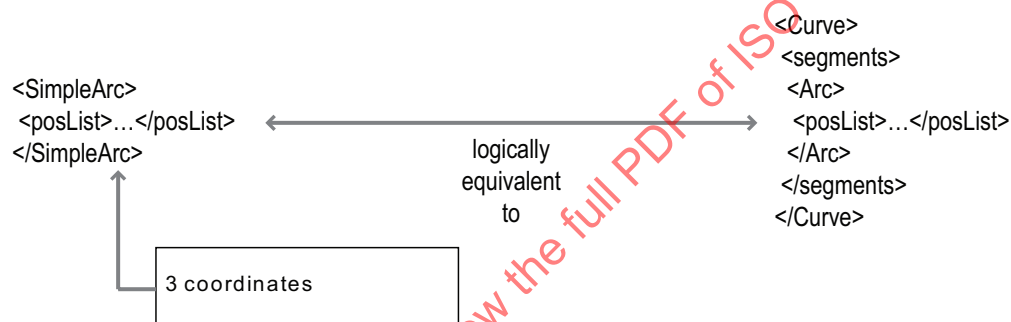


Figure 5 — SimpleArc encoding

The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleArc" type="gmlce:SimpleArcType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

<complexType name="SimpleArcType">
  <complexContent>
    <extension base="gml:AbstractCurveType">
      <sequence>
        <choice>
          <choice minOccurs="3" maxOccurs="3">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
      </sequence>
      <attribute name="interpolation" type="gml:CurveInterpolationType"
fixed="circularArc3Points"/>
      <attribute name="numArc" type="integer" fixed="1"/>
    </extension>
  </complexContent>
</complexType>

```

7.8 SimpleArcByCenterPoint

A `gml:SimpleArcByCenterPoint` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with a single `gml:ArcByCenterPoint` segment. As with the `gml:ArcByCenterPoint`, this representation can be used only in 2D. The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleArcByCenterPoint" type="gmlce:SimpleArcByCenterPointType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

<complexType name="SimpleArcByCenterPointType">
  <complexContent>
    <extension base="gml:AbstractCurveType">
      <sequence>
        <choice>
          <choice>
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
        <element name="radius" type="gml:LengthType"/>
        <element name="startAngle" type="gml:AngleType"/>
        <element name="endAngle" type="gml:AngleType"/>
      </sequence>
      <attribute name="interpolation" type="gml:CurveInterpolationType"
        fixed="circularArcCenterPointWithRadius"/>
      <attribute name="numArc" type="integer" use="required" fixed="1"/>
    </extension>
  </complexContent>
</complexType>

```

NOTE Start and end angle are mandatory in this encoding unlike `gml:ArcByCentrePoint`.

7.9 SimpleArcStringByBulge

A `gmlce:SimpleArcStringByBulge` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with one or more `gml:ArcByBulge` segments. The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleArcStringByBulge" type="gmlce:SimpleArcStringByBulgeType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

<complexType name="SimpleArcStringByBulgeType">
  <complexContent>
    <extension base="gml:AbstractCurveType">
      <sequence>
        <choice>
          <choice minOccurs="2" maxOccurs="unbounded">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
        <element name="bulge" type="double" maxOccurs="unbounded"/>
        <element name="normal" type="gml:VectorType" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="interpolation" type="gml:CurveInterpolationType"
        fixed="circularArc2PointWithBulge"/>
      <attribute name="numArc" type="integer"/>
    </extension>
  </complexContent>
</complexType>

```

7.10 SimpleArcByBulge

A `gml:SimpleArcByBulge` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with one `gml:ArcByBulge` segment. The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleArcByBulge" type="gmlce:SimpleArcByBulgeType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

<complexType name="SimpleArcByBulgeType">
  <complexContent>

```

```

<extension base="gml:AbstractCurveType">
  <sequence>
    <choice>
      <choice minOccurs="2" maxOccurs="2">
        <element ref="gml:pos"/>
        <element ref="gml:pointProperty"/>
      </choice>
      <element ref="gml:posList"/>
    </choice>
    <element name="bulge" type="double" maxOccurs="unbounded"/>
    <element name="normal" type="gml:VectorType" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="interpolation" type="gml:CurveInterpolationType"
fixed="circularArc2PointWithBulge"/>
  <attribute name="numArc" type="integer" fixed="1"/>
</extension>
</complexContent>
</complexType>

```

7.11 SimpleCircle

A `gmlce:SimpleCircle` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with a single `gml:Circle` segment. See [Figure 6](#).

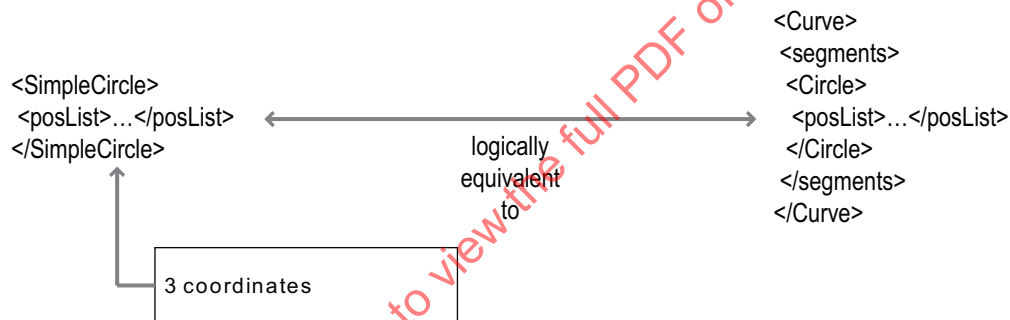


Figure 6 — SimpleCircle encoding

Consistent with `gml:Circle`, a `gmlce:SimpleCircle` is an arc whose ends coincide to form a simple closed loop. The three control points shall be distinct non-co-linear points for the circle to be unambiguously defined. The arc is simply extended past the third control point until the first control point is encountered.

The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleCircle" type="gmlce:SimpleArcType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

```

7.12 SimpleCircleByCenterPoint

A `gmlce:SimpleCircleByCenterPoint` is a specialized curve that has a simplified encoding of the logically equivalent `gml:Curve` with a single `gml:CircleByCenterPoint` segment. As with the `gml:CircleByCenterPoint`, this representation can be used only in 2D. The schema element declaration and corresponding type definition is as follows:

```

<element name="SimpleCircleByCenterPoint"
  type="gmlce:SimpleArcByCenterPointType"
  substitutionGroup="gmlce:AbstractSimpleArcString"/>

```

Start and end angles are mandatory and should not be identical; they should differ by 360 degrees.

7.13 SimpleMultiPoint

`gmlce:SimpleMultiPoint` implements, and provides a simplified encoding for, ISO 19107 `GM_MultiPoint` (see ISO 19107:2003, 6.5.4). A `gmlce:SimpleMultiPoint` consists of a list of `DirectPositions`.

```
<complexType name="SimpleMultiPointType">
  <complexContent>
    <extension base="gml:AbstractGeometricAggregateType">
      <sequence>
        <element ref="gml:posList"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="SimpleMultiPoint" type="gmlce:SimpleMultiPointType"
substitutionGroup="gml:AbstractGeometricAggregate" />
```

7.14 MultiPointPropertyType

A property that has a collection of points as its value domain may either be an appropriate geometry element encapsulated in an element of this type or an XLink reference to a remote geometry element (where remote includes geometry elements located elsewhere in the same document). The rules stated in ISO 19136:2007, 7.2.3.4 apply.

```
<complexType name="MultiPointPropertyType">
  <choice minOccurs="0">
    <element ref="gml:MultiPoint"/>
    <element ref="gmlce:SimpleMultiPoint"/>
  </choice>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
  <attributeGroup ref="gml:OwnershipAttributeGroup"/>
</complexType>
```

7.15 Requirements class

[Table 4](#) specifies the requirements class “Compact Geometry Encoding”.

Table 4 — Requirements class “Compact Geometry Encoding”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/ce	
Target type	Data instance
Name	Compact Geometry Encoding
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/ce/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/ce schema SHALL be well-formed and valid.

7.16 Conformance

[Table 5](#) specifies the conformance class “Compact Geometry Encoding”.

Table 5 — Conformance class “Compact Geometry Encoding”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/ce	
Requirements	http://www.opengis.net/spec/GML/3.3/req/ce
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Test	http://www.opengis.net/spec/GML/3.3/conf/ce/valid
	Requirement http://www.opengis.net/spec/GML/3.3/req/ce/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/ce schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/geometryCompact.xsd . Pass if no errors reported. Fail otherwise.
	Test type Basic

8 Triangulated Irregular Networks

8.1 Target namespace

All schema components specified in [Clause 8](#) are in the target namespace:

<http://www.opengis.net/gml/3.3/tin>

8.2 Introduction

The support for Triangulated Irregular Networks (TIN) in ISO 19136:2007 (GML 3.2) is consistent with ISO 19107:2003 and ISO 19125-1:2004. However, three areas of concern of the TIN model in these standards are addressed by the new schema components to support typical civil engineering practice:

- Need for more robust set of “TIN elements”
- Need for a simpler triangle encoding
- Need for a by-reference encoding of the triangulated surface

8.3 TriangulatedSurface

`gmltin:TriangulatedSurface` implements ISO 19107 GM_TriangulatedSurface (see ISO 19136:2007, D.2.3.4 and ISO 19107:2003, 6.4.37).

A triangulated surface is a polyhedral surface that is composed only of triangles. There is no restriction on how the triangulation is derived.

`gml:patches` encapsulates the triangles of the triangulated surface. All patches shall be simple triangle patches.

```
<element name="TriangulatedSurface" type="gml:SurfaceType"
substitutionGroup="gml:Surface"/>
```

8.4 SimpleTrianglePatch

`gmltin:SimpleTrianglePatch` is based upon the pattern used in the GML 3.3 SimpleTriangle, rather than the GML 3.2 Triangle.

A `gmltin:SimpleTrianglePatch` is a surface patch that is defined by a set of three points. The points shall be coplanar and the resultant simple triangle uses planar interpolation in its interior. The

boundary of simple triangle patch can be obtained by constructing a `gml:LinearRing`, defined by four coordinate tuples representing the three points such that the first point is repeated as the last point, with linear interpolation between the points.

`gml:interpolation` is fixed to “planar”, i.e. an interpolation shall return points on a single plane. The boundary of the patch shall be contained within that plane.

```
<complexType name="SimpleTrianglePatchType">
  <complexContent>
    <extension base="gml:AbstractSurfacePatchType">
      <sequence>
        <choice>
          <choice minOccurs="3" maxOccurs="3">
            <element ref="gml:pos"/>
            <element ref="gml:pointProperty"/>
          </choice>
          <element ref="gml:posList"/>
        </choice>
      </sequence>
      <attribute name="interpolation"
        type="gml:SurfaceInterpolationType"
        fixed="planar"/>
    </extension>
  </complexContent>
</complexType>

<element name="SimpleTrianglePatch" type="gmltin:SimpleTrianglePatchType"
  substitutionGroup="gml:AbstractSurfacePatch"/>
```

8.5 TIN

`gmltin:TIN` is a triangulated surface that uses the Delaunay algorithm or a similar algorithm complemented with consideration of constraints defined by TIN Elements (`gmltin:tinElements`) and possibly a maximum length of triangle sides (`gmltin:maxLength`).

```
<complexType name="TINType">
  <complexContent>
    <extension base="gml:SurfaceType">
      <sequence>
        <element name="tinElement" type="gmltin:TINElementPropertyType" minOccurs="0"
          maxOccurs="unbounded"/>
        <element name="maxLength" type="gml:LengthType"
          minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="TIN" type="gmltin:TINType"
  substitutionGroup="gmltin:TriangulatedSurface"/>
```

The set of the positions (three or more) used as posts for this TIN (corners of the triangles in the TIN) are specified either in TIN Element control points or group spots.

`gmltin:TIN` implements ISO/IEC 13249-3 ST_TIN (see ISO/IEC 13249-3:2011, 8.6.1 ST_TIN Type) which extends ISO 19107 (see ISO 19136:2007, D.2.3.4 and ISO 19107:2003, 6.4.37).

8.6 TINElement

`gmltin:TINElement` implements ISO/IEC 13249-3 ST_TINElement (see ISO/IEC 13249-3:2011, 15.1.1 ST_TINElement Type). It specifies elements associated with a TIN which can be:

- 1) constraints (boundary, break line, soft break, control contour, break void, drape void, void, hole, stop line) applied to a TIN surface after the initial Delaunay triangulation which modify the surface,

- 2) points (random points, group spots) which are the vertices of simple triangles that define the TIN surface or vertices of constraints, or
- 3) user-defined.

```

<complexType name="TINElementType">
  <complexContent>
    <extension base="gml:AbstractFeatureType">
      <sequence>
        <element name="elementType"
          type="gmltin:TINElementTypeType"/>
        <element name="elementID" type="integer" minOccurs="0"/>
        <element name="elementTag" type="string" minOccurs="0"/>
        <element name="elementGeometry"
          type="gml:GeometryPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="TINElement" type="gmltin:TINElementType"
  substitutionGroup="gml:AbstractObject"/>

```

The `gmltin:elementType` property specifies the type of TIN element. Allowable values are specified by the `gmltin:TINElementTypeType` codelist.

The `gmltin:elementID` property allows for an optional integer value to numerically identify the TIN element.

The `gmltin:elementTag` property allows for an optional string value to alphanumerically identify the TIN element.

The `gmltin:elementGeometry` property specifies the geometry of the TIN element. The allowable geometry type is dependent upon the value of the `gmltin:elementType`:

- random points and group spot TIN elements shall have a 3D `gml:MultiPoint` geometry
- boundary, break void and void TIN elements shall have a 3D `gml:Polygon` geometry
- drape void and hole TIN elements shall have a 2D or 3D `gml:Polygon` geometry
- break line, soft break and control contour TIN elements shall have a 3D `gml:LineString` geometry
- stop line TIN elements shall have a 2D or 3D `gml:LineString` geometry
- for a user-defined TIN element, the choice of `gml:GeometryPropertyType` shall be user-defined.

8.7 TINElementPropertyType

A property that has a TIN element as its value domain may either be an appropriate TIN element encapsulated in an element of this type or an XLink reference to a remote TIN element (where remote includes TIN elements located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```

<complexType name="TINElementPropertyType">
  <sequence minOccurs="0">
    <element ref="gmltin:TINElement"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>

```

8.8 TINElementTypeType

`gmltin:TINElementTypeType` is a codelist which is the union of an enumeration of pre-defined TIN element types and a pattern for specifying a user-defined TIN type.

```
<simpleType name="TINElementType">
  <union>
    <simpleType>
      <restriction base="string">
        <enumeration value="randomPoints"/>
        <enumeration value="groupSpot"/>
        <enumeration value="boundary"/>
        <enumeration value="breakline"/>
        <enumeration value="softBreak"/>
        <enumeration value="controlContour"/>
        <enumeration value="breakVoid"/>
        <enumeration value="drapeVoid"/>
        <enumeration value="void"/>
        <enumeration value="hole"/>
        <enumeration value="stopLine"/>
      </restriction>
    </simpleType>
    <simpleType>
      <restriction base="string">
        <pattern value="other:\w{2,}" />
      </restriction>
    </simpleType>
  </union>
</simpleType>
```

A TIN element of type 'random points' (`gmltin:randomPoints`) represents points on the surface of known elevation from which triangles can be generated or which represent linestring points contained in other TIN element geometries.

A TIN element of type 'group spot' (`gmltin:groupSpot`) represents a collection of related points on the surface of known elevation from which triangles can be generated or which represent linestring points contained in other TIN element geometries.

A TIN element of type 'boundary' (`gmltin:boundary`) is used to specify the boundary of the TIN surface. As a constraint applied to a TIN surface after the initial Delaunay triangulation, it causes the surface to be clipped to the boundary TIN element's `gml:Polygon` geometry value. This may result in the elimination or addition of points contained in the initial random points or group spots TIN elements. It may also result in localized re-triangulation at the boundary. It is implementation-defined whether interior boundaries are supported.

A TIN element of type 'breakline' (`gmltin:breakline`) is used to represent a local ridge or depression in the TIN surface. When a breakline is specified for a TIN surface, simple triangle patches must be adjusted so that no triangle is crossed by the breakline. Part or all of the breakline becomes an edge of two or more triangles. The elevation along the breakline takes precedence over the elevation of the original triangulated surface for the entire length of the breakline.

A TIN element of type 'soft break' (`gmltin:softBreak`) behaves as a 'breakline' (see above) except that contour lines generated for the surface can be smoothed where they cross soft breaks.

A TIN element of type 'control contour' (`gmltin:controlContour`) behaves as a 'breakline' (see above). The *z* coordinate values must be identical for all points in the 'control contour' `gml:LineString` geometry. Triangles within the vicinity of a control contour need to be assessed during re-triangulation to insure that they are not zero slope triangles (all three vertices fall on the same control contour).

Voids enclose a voided area of the TIN surface. They can be represented by either break, drape or (regular) void types of TIN element. Void geometry is of type `gml:Polygon`, which might include interior boundaries. Triangles within a void still exist in the TIN surface but are considered to be void. Triangles inside of the `gml:Polygon` interior boundary are in the exterior of the `gml:Polygon` and therefore not in the void: these triangles are therefore not considered to be void.

For a void TIN element of type 'break void' (`gmltin:breakVoid`) the boundary linestrings (`gml:LineString`) of the 3D `gml:Polygon` behave as breaklines in that triangles in the simple triangle patch collection must be adjusted so that no triangle is crossed by the break void boundary. Part or all of the break void boundary becomes an edge of two or more triangles. The elevation of this

break void boundary takes precedence over the elevation of the original triangulated surface for the entire length of the boundary.

For a void TIN element of type 'drape void' (`gmltin:drapeVoid`) the boundary linestrings (`gml:LineString`) of the 2D or 3D `gml:Polygon` behave as breaklines in that triangles in the simple triangle patch collection must be adjusted so that no triangle is crossed by the drape void boundary. Part or all of the drape void boundary becomes an edge of two or more triangles. However, for drape voids, the elevation of the original triangulated surface takes precedence over the elevation of the drape void boundary.

For a void TIN element of type (regular) 'void' (`gmltin:void`) the boundary linestrings (`gml:LineString`) of the 3D `gml:Polygon` behave as breaklines in that triangles in the simple triangle patch collection must be adjusted so that no triangle is crossed by the void boundary. Part or all of the void boundary becomes an edge of two or more triangles. However, for regular voids, only the elevations of the void boundary vertices take precedence over the elevation of corresponding points on the original surface. That is, these vertices are treated as points for triangulating. The regular void boundaries between these vertices are handled as drape void boundaries – elevations from the original surface take precedence.

A hole is an area of a TIN surface, defined by a 2D or 3D `gml:Polygon`, which is to be treated as a hole in the surface in that the triangles within this area still exist but are considered to be part of the hole. The difference between a void and a hole is realized when two TIN surfaces are merged. When merging TIN surface A with TIN surface B, a void in surface A will take precedence over what is in the same area in B, the result being retention of the voided A triangles. A hole in surface A results in that part of surface B showing through the hole and becoming part of the resultant merged surface. Though GML does not support merging of TINs, it is necessary to persist the distinction between voids and holes to make this information available to applications which do.

Hole boundaries are different than an interior boundary of a surface. The hole is still part of the interior of the surface since the triangles still exist. They are merely considered as being in the hole so applications know how to deal with them in a visibility context. This also allows for "islands" to exist inside the holes (if the hole polygon has interior rings): here the triangles exist and are not hole triangles. Had the triangles in the hole been eliminated, islands could not be supported in a single surface.

A TIN element of type 'hole' (`gmltin:hole`) encloses an area of the TIN surface designated as a hole. When a hole is specified for `gmltin:TINElement`, the boundary linestrings (`gml:LineString`) of the 2D or 3D `gml:Polygon` behave as breaklines in that triangles in the simple triangle patch collection must be adjusted so that no triangle is crossed by the hole boundary. Part or all of the hole boundary becomes an edge of two or more triangles. Hole boundaries are treated like drape void boundaries in that the elevation of the original triangulated surface takes precedence over the elevation of the hole boundary, if present.

A TIN element of type 'stopLine' (`gmltin:stopLine`) is used to specify areas where the local continuity or regularity of the TIN surface is questionable. It is implementation-defined whether triangles in the simple triangle patch collection whose boundaries are (2D) crossed by a stop line are removed from the collection of simple triangle patches or are retained but enclosed within a 'drape void' (`gmltin:drapeVoid`) type of `gmltin:TINElement`.

8.9 Requirements class

[Table 6](#) specifies the requirements class "Triangulated Irregular Networks".

Table 6 — Requirements class “Triangulated Irregular Networks”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/tin	
Target type	Data instance
Name	Triangulated Irregular Networks
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/tin/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/tin schema SHALL be well-formed and valid.

8.10 Conformance

[Table 7](#) specifies the conformance class “Triangulated Irregular Networks”.

Table 7 — Conformance class “Triangulated Irregular Networks”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/tin	
Requirements	http://www.opengis.net/spec/GML/3.3/req/tin
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Test	http://www.opengis.net/spec/GML/3.3/conf/tin/valid
	Requirement http://www.opengis.net/spec/GML/3.3/req/tin/valid Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/tin schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/tin.xsd . Pass if no errors reported. Fail otherwise.
	Test type Basic

9 Linear Referencing

9.1 Target namespaces

Schema components specified in [Clause 9](#) are in the following target namespaces, based on conformance class:

<http://www.opengis.net/gml/3.3/lr> - Basic linear referencing functionality

<http://www.opengis.net/gml/3.3/lrtr> - Towards referent extension; dependent upon gmlr

<http://www.opengis.net/gml/3.3/lro> - Offset extension; dependent upon gmlr

<http://www.opengis.net/gml/3.3/lrov> - Offset vector extension; dependent upon gmlr and gmlro

9.2 Introduction

ISO 19136:2007 so far mainly supports 2-D and 3-D spatial locations. An alternative location scheme is to use linearly referenced locations, where a location is specified as being at a certain distance along (and perhaps offset from) a linear element. The linear element might be a feature which exhibits linear behaviour (i.e., it is linearly measurable), a curve type of geometry, or a directed edge type of topology.

A significant amount of historical data is currently linearly referenced, especially in the transportation, piping, petroleum and mineral exploration industries. There are advantages of using this scheme over (or in addition to) spatial referencing.

Just as spatial data may utilize a variety of spatial (coordinate) reference systems, linearly referenced locations use a wide variety of Linear Referencing Methods (LRM). A well documented theoretical basis has been developed to support many of the more common LRMs in practice internationally today (ISO 19148).

9.3 Basic Linear Referencing

9.3.1 Target namespace

All schema components specified in 9.3 are in the target namespace:

<http://www.opengis.net/gml/3.3/lr>

9.3.2 Introduction

Basic Linear Referencing includes the specification of linearly referenced locations. These are described by a position expression which consists of the linear element being measured, the method of measurement and a measure value specified with a distance expression.

9.3.3 PositionExpression

`gml:PositionExpression` implements ISO 19148 LR PositionExpression (see ISO 19148:2012, 6.2.2).

A `gml:PositionExpression` specifies a position as a linearly referenced location given by the linear element being measured, the method of measurement and a measure value specified by a distance expression.

```
<complexType name="PositionExpressionType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <element name="linearElement"
          type="gml:LinearElementPropertyType"/>
        <element name="lrm"
          type="gml:LinearReferencingMethodPropertyType"/>
        <element name="distanceExpression"
          type="gml:DistanceExpressionPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="PositionExpression" type="gml:PositionExpressionType"
  substitutionGroup="gml:AbstractGML"/>
```

The element `gml:linearElement` specifies the linear element being measured.

The element `gml:lrm` specifies the linear referencing method of measurement.

The element `gml:distanceExpression` specifies the measured value.

9.3.4 PositionExpressionPropertyType

A property that has a position expression as its value domain may either be an appropriate position expression encapsulated in an element of this type or an XLink reference to a remote position expression (where remote includes position expressions located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="PositionExpressionPropertyType">
  <sequence minOccurs="0">
    <element ref="gml:PositionExpression"/>
  </sequence>
</complexType>
```

```

    </sequence>
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
  </complexType>

```

9.3.5 LinearElement

`gmlr:LinearElement` implements ISO 19148 LR_LinearElement (see ISO 19148:2012, 6.2.3).

A `gmlr:LinearElement` specifies the underlying linear element upon which the measures in the Linear Referencing System are made. The linear element can be either a feature, a curve geometry or a topological edge.

```

<complexType name="LinearElementType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <choice>
          <element name="feature"
            type="gml:FeaturePropertyType"/>
          <element name="curve" type="gml:CurvePropertyType"/>
          <element name="edge"
            type="gml:DirectedEdgePropertyType"/>
        </choice>
        <element name="defaultLRM"
          type="gmlr:LinearReferencingMethodPropertyType"/>
        <element name="measure" type="gml:MeasureType"/>
        <element name="startValue" type="gmlr:StartValueType"
          minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="LinearElement" type="gmlr:LinearElementType"
  substitutionGroup="gml:AbstractGML"/>

```

The element `gmlr:feature` specifies that the linear element is a `gml:AbstractFeature`. `gmlr:feature` includes any feature as defined in ISO 19109:2005, 7.3 which can be linearly measured.

The element `gmlr:curve` specifies that the linear element is a `gml:AbstractCurve`. `gmlr:curve` includes any one-dimensional geometry of type `GM_Curve` from ISO 19107:2003, 6.3.16 which can be linearly measured. Spatial positions can be projected onto the `gml:AbstractCurve` to determine a corresponding linearly referenced location along the curve and vice versa.

The element `gmlr:edge` specifies that the linear element is a `gml:DirectedEdge`. `gmlr:edge` includes any one-dimensional topology of type `TP_DirectedEdge` from ISO 19107:2003, 7.3.15 which can be linearly measured. Directed edges typically have one or more weights associated with them instead of having a length. Measuring along a directed edge therefore entails prorating a weight value. Consequently, the Linear Referencing Method of choice is typically of the interpolative type and unlikely to be of the relative type.

`gmlr:defaultLRM` implements the ISO 19148 LR_ILinearElement::defaultLRM() operation (see ISO 19148:2012, 6.2.8.2).

The element `gmlr:defaultLRM` specifies the default Linear Referencing Method of the `gmlr:LinearElement`. This is used for all measurements made along the `gmlr:LinearElement` unless specified otherwise in a `gmlr:PositionExpression` or otherwise explicitly overridden.

`gmlr:measure` implements the ISO 19148 LR_ILinearElement::measure(measureAttribute) operation (see ISO 19148:2012, 6.2.8.3).

The element `gmlr:measure` specifies the value of one of the measure attributes of the `gmlr:LinearElement`. This is usually one of its length attributes or, in the case of `gmlr:edge`, one of its weight attributes. Representing the overall length (or weight) of the `gmlr:LinearElement`, the `gmlr:measure` value is used for all calculations requiring the total linear element length,

unless explicitly overridden, for example, when converting from an absolute to an interpolative Linear Referencing Method.

`gml:LinearElement::startValue` implements the ISO 19148 `LR_ILinearElement::startValue(LRM)` operation (see ISO 19148:2012, 6.2.8.6).

The element `gml:LinearElement::startValue` specifies the measure value at the start of the `gml:LinearElement` for each appropriate Linear Referencing Method. If not provided, a default value of 0 (zero) is assumed for all Linear Referencing Methods.

9.3.6 LinearElementPropertyType

A property that has a linear element as its value domain may either be an appropriate linear element encapsulated in an element of this type or an XLink reference to a remote linear element (where remote includes linear elements located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="LinearElementPropertyType">
  <sequence minOccurs="0">
    <element ref="gml:LinearElement"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.3.7 StartValueType

Because the start value is a function of the Linear Referencing Method, the `gml:StartValueType` extends a start value of type `double` with two attributes.

```
<complexType name="StartValueType">
  <simpleContent>
    <extension base="double">
      <attribute name="uom" type="gml:UomIdentifier"/>
      <attribute name="lrm" type="anyURI" use="required">
        <annotation>
          <appinfo>
            <gml:targetElement>
              gml:LinearReferencingMethod
            </gml:targetElement>
          </appinfo>
        </annotation>
      </attribute>
    </extension>
  </simpleContent>
</complexType>
```

The optional attribute `gml:uom` provides the units of measure for the start value. If none is provided, the `uom` value defaults to the units value of the Linear Referencing Method.

The mandatory attribute `gml:lrm` specifies the Linear Referencing Method for which this start value applies. Though the type is `anyURI`, it is expected that the value will be a reference to an already defined `gml:LinearReferencingMethod` object.

9.3.8 LinearReferencingMethod

`gml:LinearReferencingMethod` implements ISO 19148 `LR_LinearReferencingMethod` (see ISO 19148:2012, 6.2.10).

A `gml:LinearReferencingMethod` specifies the manner in which measurements are made along a linear element.

```
<complexType name="LinearReferencingMethodType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
```

```

<sequence>
  <element name="name" type="gmlr:LRMNameType"/>
  <element name="type" type="gmlr:LRMTypeType"/>
  <element name="units" type="gml:UomIdentifier"/>
  <element name="constraint" type="string" minOccurs="0"/>
</sequence>
</extension>
</complexContent>
</complexType>

<element name="LinearReferencingMethod"
  type="gmlr:LinearReferencingMethodType"
  substitutionGroup="gml:AbstractGML"/>

```

The element `gmlr:name` specifies the name of the Linear Referencing Method, such as "kilometre-point". Allowable values are specified by the `gmlr:LRMNameType` codelist.

The element `gmlr:type` specifies the type of Linear Referencing Method. Allowable values are specified by the `gmlr:LRMTypeType` codelist.

The element `gmlr:units` specifies the units of measure used by the Linear Referencing Method as a `gml:UomIdentifier`. This applies to measurements made along the linear element.

The optional element `gmlr:constraint` allows for optional string values which specify any constraints imposed by the Linear Referencing Method.

9.3.9 LinearReferencingMethodPropertyType

A property that has a Linear Referencing Method as its value domain may either be an appropriate Linear Referencing Method encapsulated in an element of this type or an XLink reference to a remote Linear Referencing Method (where remote includes Linear Referencing Methods located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```

<complexType name="LinearReferencingMethodPropertyType">
  <sequence minOccurs="0">
    <element ref="gmlr:LinearReferencingMethod"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>

```

9.3.10 DistanceExpressionType

`gmlr:DistanceExpression` implements ISO 19148 LR_DistanceExpression (see ISO 19148:2012, 6.2.12).

A `gmlr:DistanceExpression` specifies the linear referenced measure value. This includes the distance measured along the linear element. If the Linear Referencing Method `gmlr:LRMType` is "relative", the distance expression also includes an along referent to specify where the measuring begins. Otherwise, measuring begins at the start of the linear element. Measuring is in the direction of the linear element, unless a towards referent is provided.

```

<complexType name="DistanceExpressionType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <element name="distanceAlong" type="gmlr:MeasureType"/>
        <element name="referent" type="gmlr:AlongReferentPropertyType"
          minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="DistanceExpression" type="gmlr:DistanceExpressionType"
  substitutionGroup="gml:AbstractGML"/>

```

The element `gmlr:distanceAlong` specifies the measure value (usually a distance) of the distance expression. It is of type `gmlr:MeasureType`. `gmlr:MeasureType` is the same as `gml:MeasureType` except that the `uom` value is optional. If none is provided, the `uom` value defaults to the units value of the Linear Referencing Method.

If the Linear Referencing Method `gmlr:type` is 'absolute', the distance along is measured from the start of the linear element. If 'relative', it is measured from the along referent's from referent. If 'interpolative', it is based upon the default length or weight of the linear element (`gmlr:measure`).

The optional element `gmlr:referent` specifies the referent associated with the distance expression `gmlr:distanceAlong` value. A referent is only appropriate if the Linear Referencing Method `gmlr:type` is "relative". If the referent is absent, then the position is measured from the start of the linear element.

9.3.11 DistanceExpressionPropertyType

A property that has a Distance Expression as its value domain may either be an appropriate Distance Expression encapsulated in an element of this type or an XLink reference to a remote Distance Expression (where remote includes Distance Expression located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="DistanceExpressionPropertyType">
  <sequence minOccurs="0">
    <element ref="gmlr:DistanceExpression"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.3.12 AlongReferent

`gmlr:AlongReferent` implements ISO 19148 LR_AloneReferent (see ISO 19148:2012, 6.2.13).

```
<complexType name="AlongReferentType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <element name="fromReferent"
          type="gmlr:ReferentPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="AlongReferent" type="gmlr:AlongReferentType"
  substitutionGroup="gml:AbstractGML"/>
```

For Linear Referencing Methods having a `gmlr:type` of "relative", the `gmlr:AlongReferent` specifies a from referent.

The element `gmlr:fromReferent` specifies where along the `gmlr:LinearElement` measuring begins for the distance expression when the Linear Referencing Method is of type 'relative'.

9.3.13 AlongReferentPropertyType

A property that has an along referent as its value domain may either be an appropriate along referent encapsulated in an element of this type or an XLink reference to a remote along referent (where remote includes along referents located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="AlongReferentPropertyType">
  <sequence minOccurs="0">
    <element ref="gmlr:AlongReferent"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

```
</complexType>
```

9.3.14 Referent

`gmlr:Referent` implements ISO 19148 LR_Referent (see ISO 19148:2012, 6.2.14).

For from and towards referents, `gmlr:Referent` is used to specify a known location along the `gmlr:LinearElement`. This can be a reference marker, an intersection, a jurisdictional boundary, or a landmark.

```
<complexType name="ReferentType">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <element name="name" type="gml:CodeType"/>
        <element name="type" type="gmlr:ReferentTypeType"/>
        <element name="position" type="gml:PointPropertyType"
          minOccurs="0"/>
        <element name="location"
          type="gmlr:PositionExpressionPropertyType" minOccurs="0"/>
        <element name="ownedBy" type="gml:FeaturePropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="Referent" type="gmlr:ReferentType"
  substitutionGroup="gml:AbstractGML"/>
```

The element `gmlr:name` specifies the name of the referent, which is of type `gml:CodeType` to allow specification of the authority which created the name.

The element `gmlr:type` specifies the type of referent. Allowable values are specified by the `gmlr:ReferentTypeType` codelist.

The optional element `gmlr:position` specifies the spatial position of the referent, given in some coordinate reference system.

The optional element `gmlr:location` specifies the location of the referent given as a linearly referenced location (`gmlr:PositionExpression`) along the feature which owns the referent. The Linear Referencing Method contained within the position expression specifies how this measurement is made: it could be absolute from the start of the element, relative from the previous (or other) referent, or interpolative. The Linear Referencing Method of the referent location `gmlr:PositionExpression` does not have to be the same as the Linear Referencing Method of the position expression containing the distance expression which uses this referent. There shall be no offset expression in the referent location `gmlr:PositionExpression`; all referents shall lie on the linear feature.

At least one of the attributes `gmlr:position` or `gmlr:location` is usually specified, unless it can be implied. If both are given they shall refer to the same physical location.

The element `gmlr:ownedBy` specifies the feature which owns the referent. Referents are owned by a single feature. For example, the reference markers along Interstate 95 are owned by the feature which represents Interstate 95. The referent representing the intersection with First Avenue along Washington Street is owned by Washington Street if it is used to specify relative linearly referenced locations along Washington Street. The location of this referent will most likely be specified with a position expression along Washington Street. A different referent can represent the intersection with Washington Street along First Avenue. This referent is owned by First Avenue and is used to specify relative linearly referenced locations along First Avenue.

The `gml:AbstractFeature` which owns a `gmlr:Referent` does not have to be the same `gml:AbstractFeature` which is the `gml:LinearElement` for the `gml:PositionExpression` containing that `gmlr:Referent`. For example, if US Interstate Highway 95 is coincident with US Federal Route 1, it is possible to specify locations along Route 1 (the linear element in the position expression) using referents owned by Interstate 95 (reference post 18). In this case, the location of the

referent itself most likely would have been made along Interstate 95 as the linear element since the referent is owned by Interstate 95.

9.3.15 ReferentPropertyType

A property that has a referent as its value domain may either be an appropriate referent encapsulated in an element of this type or an XLink reference to a remote referent (where remote includes referents located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="ReferentPropertyType">
  <sequence minOccurs="0">
    <element ref="gmlr:Referent"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.3.16 MeasureType

`gmlr:MeasureType` is the same as `gml:MeasureType` except that the uom value is optional.

```
<complexType name="MeasureType">
  <simpleContent>
    <extension base="double">
      <attribute name="uom" type="gml:UomIdentifier"/>
    </extension>
  </simpleContent>
</complexType>
```

9.3.17 LRMNameType

`gmlr:LRMNameType` is a codelist which includes the informative values of Linear Referencing Method names from ISO 19148:2012, Annex C and also allows for extensions of user-defined Linear Referencing Method names. Because there is such a diverse range of Linear Referencing Methods and names, this is offered as informative only. The enumerated names should only be used if their meaning matches that of the so-named Linear Referencing Method in ISO 19148:2012, Annex C.

In accordance with Linear Referencing Methods having a `gmlr:type` of 'absolute', measurements are made along the linear element from its start, in the direction of the linear element. Commonly used absolute Linear Referencing Methods include milepoint (also known as true mileage), its metric analogue kilometre-point (also known as kilopoint), chainage, hectometer-point, reverse milepoint and its metric equivalent reverse kilometer-point and milepoint with lateral offsets in feet.

```
<complexType name="LRMNameType">
  <annotation>
    <documentation>
      Informative values from ISO 19148 Annex C
    </documentation>
  <appinfo>
    <restriction base="string">
      <enumeration value="milepoint"/>
      <enumeration value="trueMileage"/>
      <enumeration value="kilometre-point"/>
      <enumeration value="kilopoint"/>
      <enumeration value="chainage"/>
      <enumeration value="hectometre-point"/>
      <enumeration value="reverseMilepoint"/>
      <enumeration value="reverseKilometre-point"/>
      <enumeration value="milepointWithLateralOffsetsInFeet"/>
      <enumeration value="milepost"/>
      <enumeration value="kilopost"/>
      <enumeration value="kilometre-post"/>
      <enumeration value="referencePost"/>
      <enumeration value="countyMilepoint"/>
      <enumeration value="crossStreet"/>
    </restriction>
  </appinfo>
</complexType>
```

```

        <enumeration value="controlSection"/>
        <enumeration value="percentage"/>
        <enumeration value="normalized"/>
        <enumeration value="stationing"/>
        <enumeration value="address"/>
        <enumeration value="mileMeasure"/>
        <enumeration value="kilometreMeasure"/>
    </restriction>
</appinfo>
</annotation>
<simpleContent>
    <extension base="gml:CodeType"/>
</simpleContent>
</complexType>

```

An LRM name of ‘milepoint’ designates a Linear Referencing Method where measurements are made in decimal miles along the linear element from its start, in the direction of the linear element (see ISO 19148:2012, C.2.2).

An LRM name of ‘trueMileage’ designates an alternative name for a ‘milepoint’ Linear Referencing Method (see ISO 19148:2012, C.2.3).

An LRM name of ‘kilometre-point’ designates a Linear Referencing Method which is identical to the ‘milepoint’ Linear Referencing Method except that the `gmlr:units` is kilometre instead of mile and the `gmlr:distanceAlong` is measured in kilometres (see ISO 19148:2012, C.2.4).

An LRM name of ‘kilopoint’ designates an alternative name for a ‘kilometre-point’ Linear Referencing Method (see ISO 19148:2012, C.2.4).

An LRM name of ‘chainage’ designates a Linear Referencing Method which is identical to the ‘milepoint’ Linear Referencing Method except that the `gmlr:units` is metre instead of mile. Though the original British definition of “chainage” might imply measuring in chains, “chainage” was adopted as a consensus term for this Linear Referencing Method by a team of European Road Authority representatives (including British) for ISO 14825 Geographic Data Files (GDF) (see ISO 19148:2012, C.2.5).

An LRM name of ‘hectometre-point’ designates a Linear Referencing Method which is identical to the milepoint LRM except that the `gmlr:units` is hectometre (100 metres) instead of mile and the `gmlr:distanceAlong` is measured in hectometres (see ISO 19148:2012, C.2.6).

An LRM name of ‘reverseMilepoint’ designates a Linear Referencing Method where measurements are made in decimal miles along the linear element from its end, in the direction opposite to the direction of the linear element (see ISO 19148:2012, C.2.7).

An LRM name of ‘reverseKilometre-point’ designates a Linear Referencing Method which is identical to the ‘reverseMilepoint’ Linear Referencing Method except that the `gmlr:units` is kilometre instead of mile and the `gmlr:distanceAlong` is measured in kilometres (see ISO 19148:2012, C.2.7).

Many organizations establish a topological network of links and nodes and then specify linear locations along uniquely identified links as an absolute distance from the start of the link (the begin node location). This International Standard is very specific about separating the linear element (that which is being measured) from the Linear Referencing Method (how it is measured). Therefore, what is typically referred to as “link offset” is actually the application of an absolute Linear Referencing Method (milepoint, kilometre-point) to link linear elements.

In a topological network, links do not have length and cannot be measured. In order to use a milepoint or kilometre-point Linear Referencing Method then, a `gmlr:Curve` geometry is typically associated with each link. Though measurement is actually done along the curve, it is portrayed as being along the associated link. This International Standard also supports the notion of interpolative measurement along a `gmlr:DirectedEdge`, so that a percentage or normalized Linear Referencing Method can be used to specify linear locations along a link (see ISO 19148:2012, C.2.8).

Therefore, an LRM name of ‘linkOffset’ designating a Linear Referencing Method with a fixed linear element type of link is therefore not included in the enumeration.

An LRM name of 'milepointWithLateralOffsetsInFeet' designates a Linear Referencing Method which is an extension of the 'milepoint' Linear Referencing Method. A location is first determined by measuring along the linear element a distance in decimal miles specified by the `gmllr:distanceAlong` value. Then, if a `gmllr:LateralOffsetExpression` is specified, this location is adjusted in accordance with the `gmllr:offsetLateralDistance` given in feet. The metric analogue would be kilometre-point or hectometre-point with offsets in metres (see ISO 19148:2012, C.2.9). Support for these Linear Referencing Methods requires a Linear Referencing Method With Offsets (see 9.5).

In accordance with Linear Referencing Methods having a `gmllr:type` of 'relative', measurements are made along the linear element from a specified from referent location, usually in the direction of the linear element unless overridden by a towards referent, in which case the direction from the from referent towards the towards referent. Commonly used relative Linear Referencing Methods include milepost, its metric equivalent kilopost (also known as kilometre-post), reference post, County milepoint, cross street, and Control Section.

An LRM name of 'milepost' designates a Linear Referencing Method where measurements are made in decimal miles along the linear element from the closest, preceding milepost. Measuring is done in the direction of increasing mileposts which is usually the direction of the linear element.

In some States in the US, a single set of mileposts are used for both carriageways of a divided highway. Mileposts are usually numbered in the north or east predominant direction of the highway. Even if separate linear elements are defined for each carriageway, the along measuring is always done in the direction of increasing mileposts, even though this is opposite the direction of the westbound and southbound carriageway linear elements.

Mileposts are one mile apart. Sometimes reconstruction of a part of a roadway may change its overall length. To maintain the milepost spacing of exactly one mile, all posts within as well as those beyond the area of reconstruction must be relocated. This is not always feasible or desirable, especially if the road is hundreds of miles in length and the reconstruction occurs near the beginning of the road. Many times the State will only relocate mileposts within the reconstructed area, prorating the reconstructed distance between the number of mileposts affected. The consequence is that they are no longer one mile apart. This means that the Linear Referencing Method henceforth becomes a reference post Linear Referencing Method, requiring knowledge of the inter-post spacings.

Mileposts are owned by a `gmllr:Feature` linear element. The milepost location, if specified, shall be specified using a `gmllr:PositionExpression`. The linear element of this `gmllr:PositionExpression` will be the owning linear element. The Linear Referencing Method of this `gmllr:PositionExpression` does not have to be milepost. A milepoint Linear Referencing Method would allow locating the milepost a certain number of miles from the start of the linear element (for example, milepost 3 would have a distance along equal to 3.0). Alternatively, a milepost Linear Referencing Method could be used to specify the location of the milepost relative to the previous one (here milepost 3 would have a from referent of milepost 2 and a distance along of 1.0). Since all mileposts would have a distance along of 1.0 from the previous milepost and since they are named by their mileage from the start of the linear element, the location does not have to be persisted; it can be implied (milepost 3 is 3.0 miles from the start of the linear element that owns it) (see ISO 19148:2012, C.3.2).

An LRM name of 'kilopost' designates a Linear Referencing Method which is identical to the 'milepost' Linear Referencing Method except that the `gmllr:units` is kilometre instead of mile and the `gmllr:distanceAlong` is measured in kilometres. Kiloposts are exactly one kilometre apart. If not, it is actually a kilometre reference post Linear Referencing Method. Kilopost locations would be specified by using a metric Linear Referencing Method such as kilopoint or kilopost, or would be implicitly inferred by the kilopost name (see ISO 19148:2012, C.3.3).

An LRM name of 'kilometre-post' designates an alternative name for a 'kilopost' Linear Referencing Method (see ISO 19148:2012, C.3.4).

An LRM name of 'referencePost' designates a Linear Referencing Method which is similar to the milepost Linear Referencing Method except that the reference posts are not necessarily exactly one mile apart. It is therefore necessary to explicitly specify the location (or possibly position) of each reference post. This can be done with a milepoint Linear Referencing Method, specifying their absolute distance along

from the beginning of the owning `gml:Feature`. Alternatively, they can be located using a relative reference post Linear Referencing Method, locating each reference post a distance along from the previous reference post. Then, if reconstruction occurs, only those reference posts in the reconstruction area which are moved would have to have their distance along updated (see ISO 19148:2012, C.3.5).

An LRM name of 'countyMilepoint' designates a Linear Referencing Method which allows the milepoint value to be reset to zero when a highway enters a new County. If the linear element along which locations are to be linearly referenced is a single highway which traverses through many Counties, then this Linear Referencing Method is used. Linearly referenced locations are specified as a distance in miles from where the highway enters the County (the from referent).

If however, the highway changes its identity when it enters the new County (for example, it becomes County Route 42) and locations are to be linearly referenced along Route 42, then Route 42 is the linear element and a simple milepoint (absolute) Linear Referencing Method is appropriate, since it already starts at zero at the County line (see ISO 19148:2012, C.3.6).

An LRM name of 'crossStreet' designates a Linear Referencing Method which allows a location to be specified at a distance along a linear element, measured in miles from the intersection of one street with the linear element in a direction towards the intersection of another street with the linear element (see ISO 19148:2012, C.3.7). Support for this Linear Referencing Method requires support for Towards Referents (see 9.4).

An LRM name of 'controlSection' designates a Linear Referencing Method which allows the measure value to be reset to zero when a highway enters a new Control Section. If the linear element along which locations are to be linearly referenced is a single highway which traverses through multiple Control Sections that are defined according to agency specific business rules, then this Linear Referencing Method is used. Linearly referenced locations are specified as a distance in the specified measurement unit from where the highway enters the Control Section (the from referent or starting Control Section Anchor Point).

The philosophy behind the establishment of Control Sections is to distinguish between significant differences in road physical characteristics (for example, change from an undivided highway section to a divided highway section or vice versa) and/or to limit the length of a measured section of road in order to minimize accumulative measure value errors. Control Section begin/end points (Control Section Anchor Points) may be established either at topological nodes or at points where the linear element intersects a stable administrative boundary (for example, a County boundary).

If each Control Section is a separate linear element, then any of the absolute Linear Referencing Methods (milepoint, kilometre-point) should be used instead (see ISO 19148:2012, C.3.8).

In accordance with Linear Referencing Methods having a `gml:type` of 'interpolative', measurements are interpolated in accordance with the default length of the linear element (`gml:measure`). Commonly used interpolative Linear Referencing Methods include percentage and normalized. Interpolative Linear Referencing Methods are the only ones which are likely to be used for linear elements of type `gml:DirectedEdge`.

An LRM name of 'percentage' designates a Linear Referencing Method where measurements are made along the linear element from its start and are expressed as the percentage that this is of the total length (or weight) of the linear element (see ISO 19148:2012, C.4.2).

An LRM name of 'normalized' designates a Linear Referencing Method which is identical to the percentage Linear Referencing Method except that the measured values range from 0 (zero) to 1 (one) instead of 0 (zero) to 100. A measured value of 0 (zero) represents a location at the start of the linear element and 1 (one) represents a location at the end of the linear element.

Two other Linear Referencing Methods commonly used are stationing and address, though neither is addressed in ISO 19148.

An LRM name of 'stationing' designates the simplest stationing Linear Referencing Method. It has a `gml:type` of absolute and `gml:units` of feet. A plus sign, "+", is often interjected between the hundreds place and the tens place (1+00 is 100 feet). If station equations are supported, the Linear

Referencing Method `gmllr:name` should instead be “stationingWithStationEquations”, with a `gmllr:type` of `relative`; the location of each station equation should be a `gmllr:Referent`. There would also be metric equivalents, such as “metricStationing” with, for example, 1+000 representing 1 kilometre.

An LRM name of ‘address’ designates the simplest addressing Linear Referencing Method. It has a `gmllr:type` of `interpolative`. Here the `gmllr:distanceAlong` would be an address number. This would be interpolated against the minimum and maximum address numbers held by the linear element (typically a block). So, if the block goes from address number 101 through 199, a `gmllr:distanceAlong` address of 150 would be half way down the block ($(150-101)/(199-101) = 0.5$). More sophisticated address methods might accommodate independent address numbering on each side of the street but might require splitting the street into two linear elements.

Linear referencing by address is very approximate and only works for simple addressing schemes. Complications such as address numbers containing alpha suffixes, addressing around cul-de-sacs, apartment complex addressing, addresses which differ by building floor and addresses assigned in date order preclude the use of an address Linear Referencing Method.

In accordance with Linear Referencing Methods having a `gmllr:type` of ‘localInterpolative’, measurements are interpolated locally along a segment of a curve type of linear element bounded by two control points having *m* coordinate values bracketing the distance along measure value. Commonly used local interpolative Linear Referencing Methods include mile measure and kilometre measure. Support for these Linear Referencing Methods requires a linear element of type ‘curve’ (see 9.3.5).

NOTE Measured (“M”) coordinates are described in ISO 19133:2005, 6.5.

An LRM name of ‘mileMeasure’ designates a Linear Referencing Method where the `gmllr:distanceAlong` measurement is expressed as miles along the linear element. The exact location is determined by finding the two control points along the linear element which have *M* coordinate values which bracket the `gmllr:distanceAlong` value and then interpolating between these two points based on the length of the curve segment between them.

An LRM name of ‘kilometreMeasure’ designates a Linear Referencing Method where the `gmllr:distanceAlong` measurement is expressed as kilometres along the linear element. The exact location is determined by finding the two control points along the linear element which have *M* coordinate values which bracket the `gmllr:distanceAlong` value and then interpolating between these two points based on the length of the curve segment between them.

Specification of the above defined Linear Referencing Methods is provided in informative [Annex A](#).

9.3.18 LRMTTypeType

`gmllr:LRMTTypeType` implements ISO 19148 LR_LRMTType (see ISO 19148:2012, 6.2.11).

`gmllr:LRMTTypeType` is a codelist which includes the normative values of Linear Referencing Method types from ISO 19148 and also allows for extensions of user-defined Linear Referencing Method type. Pre-defined Linear Referencing Methods types include ‘absolute’ (see ISO 19148:2012, C.2), ‘relative’ (see ISO 19148:2012, C.3), ‘interpolative’ (see ISO 19148:2012, C.4) or ‘localInterpolative’ which is common in Geographic Information Systems supporting *m* coordinate values.

```
<complexType name="LRMTTypeType">
  <annotation>
    <documentation>
      Normative values from ISO 19148
    </documentation>
    <appinfo>
      <restriction base="string">
        <enumeration value="absolute"/>
        <enumeration value="relative"/>
        <enumeration value="interpolative"/>
        <enumeration value="localInterpolative"/>
      </restriction>
    </appinfo>
  </annotation>
```

```
<simpleContent>
  <extension base="gml:CodeType"/>
</simpleContent>
</complexType>
```

An LRM type of ‘absolute’ designates a Linear Referencing Method where measurements are made from the start of the linear element being measured, in the direction of the linear element.

An LRM type of ‘relative’ designates a Linear Referencing Method where measurements are made from the specified from referent location, in the direction of the linear element unless overridden by a towards referent in which case the direction is from the from referent towards the towards referent.

An LRM type of ‘interpolative’ designates a Linear Referencing Method where measurements are interpolated in accordance with the default length or weight (`gml:measure`) of the linear element.

An LRM type of ‘localInterpolative’ designates a Linear Referencing Method where measurements are interpolated locally along a segment of a curve type linear element bounded by two control points having *m* coordinate values bracketing the distance along measure value.

9.3.19 ReferentTypeType

`gmlr:ReferentTypeType` implements ISO 19148 LR_ReferentType (see ISO 19148:2012, 6.2.15).

`gmlr:ReferentTypeType` is a codelist which is the union of an enumeration of pre-defined referent types and a pattern for specifying a user-defined referent type. Allowable types are typically dependent upon the Linear Referencing Method.

```
<simpleType name="ReferentTypeType">
  <restriction base="string">
    <enumeration value="referenceMarker"/>
    <enumeration value="intersection"/>
    <enumeration value="boundary"/>
    <enumeration value="landmark"/>
  </restriction>
</simpleType>
```

A referent type of ‘referenceMarker’ indicates that the referent is a reference marker typically physically located in the right of way of the road, rail or other transportation system. Usually reference markers are initially spaced at a uniform distance along the linear element being measured, though subsequent re-alignments may result in uneven spacing between the markers. Specifying their location with a relative Linear Referencing Method a distance from the preceding marker minimizes the impact of such changes.

A referent type of ‘intersection’ indicates that the referent is the location of an intersection specified by the referent name. The intersection location is typically taken as the location of the intersection of the reference lines of the streets comprising the intersection and is therefore not necessarily precise or deterministic. Physical markers may be installed to remedy this. The Linear Referencing System should include specific rules about how intersection locations are determined if this type of referent is to be permitted.

A referent type of ‘boundary’ indicates that the referent represents where an administrative or maintenance boundary crosses the linear element being measured. This is typically the first time the boundary crosses the linear element. If the boundary runs along the linear element, it would be the point at which they first become collinear. The Linear Referencing System should include specific rules about how boundaries are handled if this type of referent is to be permitted. If the linear element changes at the boundary as for a County route beginning at the County boundary, then the Linear Referencing method is more correctly categorized as absolute.

A referent type of ‘landmark’ indicates that the referent is the location of a physical landmark visible in the field.

9.3.20 LinearSRS

`gml:LinearSRS` specifies a Linear Spatial Reference System as a combination of a linear element and a Linear Referencing Method, the first two components of a `gml:PositionExpression`.

```
<complexType name="LinearSRSType">
  <complexContent>
    <extension base="gml:IdentifiedObjectType">
      <sequence>
        <element name="linearElement"
          type="gmlr:LinearElementPropertyType"/>
        <element name="lrm"
          type="gmlr:LinearReferencingMethodPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="LinearSRS" type="gmlr:LinearSRSType"
  substitutionGroup="gml:Definition"/>
```

The element `gmlr:linearElement` specifies the linear element being measured.

The element `gmlr:lrm` specifies the linear referencing method of measurement.

This combination of a linear element and a Linear Referencing Method, identifiable by a `gml:id`, can be used as an SRS. For example:

```
<gml:Point gml:id="p1" srsName="#LSRS123">
  <gml:pos>15.5</gml:pos>
</gml:Point>
```

defines a Point geometry as a distance along (15.5) the linear element encapsulated in LSRS123, measured in accordance with the Linear Referencing Method which is also encapsulated in LSRS123:

```
<gmlr:LinearSRS gml:id="LSRS123">
  <gmlr:linearElement>
    <gml:LineString srsName="LS" srsDimension="3" gml:id="LS_BH18">
      <gml:posList>407829 268621 23.93 407415 268600 8.43</gml:posList>
    </gml:LineString>
  </gmlr:linearElement>
  <gmlr:lrm>
    <gmlr:LinearReferencingMethod gml:id="LRM001">
      <gmlr:name>chainage</gmlr:name>
      <!--chainage = measurement in metres -->
      <gmlr:type>absolute</gmlr:type>
      <!--absolute = measure from start of linear element -->
      <gmlr:units uom="m"/>
    </gmlr:LinearReferencingMethod>
  </gmlr:lrm>
</gmlr:LinearSRS>
```

Because `gml:pos` only allows values of type double, relative Linear Referencing Methods would be precluded from this short-hand encoding since the distance along would require inclusion of a `gmlr:referent`.

9.3.21 LinearSRSPROPERTYType

```
<complexType name="LinearSRSPROPERTYType">
  <sequence minOccurs="0">
    <element ref="gmlr:LinearSRS"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.4 Linear Referencing Towards Referent

9.4.1 Target namespace

All schema components specified in 9.4 are in the target namespace:

<http://www.opengis.net/gml/3.3/lrtr>

9.4.2 Introduction

Linear Referencing Towards Referent includes the specification of linearly referenced locations which can have a towards referent in their distance expressions.

9.4.3 DualAlongReferent

`gml:lrtr: DualAlongReferent` implements ISO 19148 LRTR_DualAlongReferent (see ISO 19148:2012, 6.3.3).

```
<complexType name="DualAlongReferentType">
  <complexContent>
    <extension base="gml:AlongReferentType">
      <sequence>
        <element name="towardsReferent"
          type="gml:ReferentPropertyType" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="DualAlongReferent" type="gml:lrtr: DualAlongReferentType"
  substitutionGroup="gml:AlongReferent"/>
```

For Linear Referencing Methods having a `gml:lrtr:type` of "relative", the `gml:lrtr: DualAlongReferent` specifies an optional towards referent.

The optional element `gml:lrtr:towardsReferent` specifies a second location along the `gml:LinearElement` to help disambiguate the `gml:fromReferent` and the direction in which the measurement is made. The `gml:lrtr:towardsReferent` overrides the directional sense of the `gml:LinearElement`, and may influence the offset direction if an offset is included. When a `gml:lrtr:towardsReferent` is part of a distance expression, the `gml:distanceAlong` value is measured along the `gml:LinearElement`, beginning at the location specified by the `gml:fromReferent` in the direction towards the `gml:lrtr:towardsReferent`. A `gml:lrtr:towardsReferent` is only appropriate for Linear Referencing Method with towards referent, for example, the "cross-street" Linear Referencing Method.

9.4.4 DualAlongReferentPropertyType

A property that has a dual along referent as its value domain may either be an appropriate dual along referent encapsulated in an element of this type or an XLink reference to a remote dual along referent (where remote includes dual along referents located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```
<complexType name="DualAlongReferentPropertyType">
  <sequence minOccurs="0">
    <element ref="gml:lrtr: DualAlongReferent"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.5 Linear Referencing Offset

9.5.1 Target namespace

All schema components specified in 9.5 are in the target namespace:

<http://www.opengis.net/gml/3.3/lro>

9.5.2 Introduction

Linear Referencing Offset includes the specification of linearly referenced locations which can have lateral and vertical offsets in their distance expressions.

9.5.3 LRMWithOffset

`gml:LRMWithOffset` implements ISO 19148 LRO_LRMWithOffset (see ISO 19148:2012, 6.4.2).

A `gml:LRMWithOffset` specifies the manner in which measurements are made along (and optionally laterally offset from) a linear element.

```
<complexType name="LRMWithOffsetType">
  <complexContent>
    <extension base="gmlr:LinearReferencingMethodType">
      <sequence>
        <element name="offsetUnits" type="gml:UomIdentifier"/>
        <element name="positiveLateralOffsetDirection"
          type="gmlro:LateralOffsetDirectionType" default="right"
          minOccurs="0"/>
        <element name="positiveVerticalOffsetDirection"
          type="gmlro:VerticalOffsetDirectionType" default="up"
          minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="LRMWithOffset" type="gmlro:LRMWithOffsetType"
  substitutionGroup="gmlr:LinearReferencingMethod"/>
```

The element `gmlro:offsetUnits` specifies the offset units of measure used by the Linear Referencing Method as a `gml:UomIdentifier`. It is mandatory for Linear Referencing Methods that allow offsets. This applies to measurements made laterally, vertically or along a vector from the linear element to an offset location.

The optional element `gmlro:positiveLateralOffsetDirection` gives the `gmlro:LateralOffsetDirectionType` value used as the positive direction for the Linear Referencing Method for lateral measures perpendicular to the linear element. It is only applicable to Linear Referencing Methods that allow offsets. Allowable values are specified by the `gmlro:LateralOffsetDirectionType` codelist. The default value is "right".

The optional element `gmlro:positiveVerticalOffsetDirection` gives the `gmlro:VerticalOffsetDirectionType` value used as the positive direction for the Linear Referencing Method for vertical measures perpendicular to the linear element. It is only applicable to Linear Referencing Methods that allow offsets. Allowable values are specified by the `gmlro:VerticalOffsetDirectionType` codelist. The default value is "up".

9.5.4 LRMWithOffsetPropertyType

A property that has a LRM With Offset as its value domain may either be an appropriate LRM With Offset encapsulated in an element of this type or an XLink reference to a remote LRM With Offset (where remote includes LRMs With Offset located elsewhere in the same document). Either the reference or the contained element shall be given, but neither both nor none.

```

<complexType name="LRMWithOffsetPropertyType">
  <sequence minOccurs="0">
    <element ref="gml:LRMWithOffset"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>

```

9.5.5 LateralOffsetDistanceExpressionType

`gmlro:LateralOffsetDistanceExpressionType` implements ISO 19148 LRO_LateralOffsetDistanceExpression (see ISO 19148:2012, 6.4.5).

A `gmlro:LateralOffsetDistanceExpressionType` specifies a linear referenced measure value which may contain a lateral offset value. It extends the `gmlr:DistanceExpressionType` with lateral offsets.

```

<complexType name="LateralOffsetDistanceExpressionType">
  <complexContent>
    <extension base="gmlr:DistanceExpressionType">
      <sequence>
        <element name="lateralOffsetExpression"
          type="gmlro:LateralOffsetExpressionType" minOccurs="0"/>
        <element name="verticalOffsetExpression"
          type="gmlro:VerticalOffsetExpressionType" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="LateralOffsetDistanceExpression"
  type="gmlro:LateralOffsetDistanceExpressionType"
  substitutionGroup="gmlr:DistanceExpression"/>

```

Lateral offsets extend the definition of a position expression to accommodate locations which do not lie directly on the linear element, but instead are located some distance horizontally or vertically (or both) from the linear element. The horizontal and vertical distances are measured along a perpendicular from the linear element or from an offset referent.

The optional element `gmlro:lateralOffsetExpression` specifies the laterally offset distance of the distance expression. If absent, the position is not displaced laterally left or right of the `gmlr:LinearElement`. A lateral offset expression is only appropriate if the Linear Referencing Method allows lateral offsets.

The optional element `gmlro:verticalOffsetExpression` specifies the vertically offset distance of the distance expression. If absent, the position is not displaced vertically above or below the `gmlr:LinearElement`. A vertical offset expression is only appropriate if the Linear Referencing Method allows vertical offsets.

9.5.6 LateralOffsetExpressionType

`gmlr:LateralOffsetExpressionType` implements ISO 19148 LRO_LateralOffsetExpression (see ISO 19148:2012, 6.4.6) and LRO_LateralOffsetReferent (see ISO 19148:2012, 6.4.7).

`gmlr:LateralOffsetExpressionType` is used to describe the lateral offset for a linearly referenced location.

```

<complexType name="LateralOffsetExpressionType">
  <simpleContent>
    <extension base="double">
      <attribute name="uom" type="gml:UomIdentifier"/>
      <attribute name="lateralOffsetReferent" type="string"/>
      <attribute name="featureGeometry" type="anyURI"/>
    </extension>
  </simpleContent>
  <documentation>
    The featureGeometry attribute value should

```

```

        reference a Geometry object
      </documentation>
      <appinfo>
        <gml:targetElement>
          gmlr:AbstractGeometry
        </gml:targetElement>
      </appinfo>
    </annotation>
  </attribute>
</extension>
</simpleContent>
</complexType>

```

The lateral offset distance is the measure of the lateral offset of the distance expression. This is the distance left or right of the lateral offset referent (or left or right of the linear element being measured if no lateral offset referent is specified) to the position being specified. A positive (+) value is measured in the direction specified by the positive lateral offset direction of the Linear Referencing Method. A value of 0 (zero) is to be interpreted as not having a lateral displacement from the lateral offset referent (or the linear element being measured if no lateral offset referent is specified). The lateral offset distance is a double value extended with three attributes.

The optional attribute `gmlr:uom` specifies the units of measure of the double value. It is of type `gml:UomIdentifier`. If none is provided, the uom value defaults to the `offsetUnits` value of the Linear Referencing Method.

The optional attribute `gmlr:lateralOffsetReferent` indicates the base line for the lateral offset measure.

`gmlr:lateralOffsetReferent` specifies the lateral offset referent as a character string. Having only a value of "back of curb" for example, allows a location to be specified as 5 feet behind the back of the curb. This aides in locating the position in the field without having to measure from the linear element. The value can be a conventional value, such as -L1 or RE from XSP (see ISO 19148:2012, C.5.2).

If the curb is represented as a Feature with a spatial representation, the distance that the curb is from the linear element being measured is determinate and therefore the exact spatial position of the position expression can be calculated. Because the Feature can have multiple spatial representations, the `gml:AbstractGeometry`, rather than the Feature, is specified.

The optional attribute `gmlr:featureGeometry` is the geometry of a Feature. Though the type is anyURI, it is expected that the value will be a reference to an already defined `gml:AbstractGeometry` object.

EXAMPLE The following Linear Referencing Method is Milepoint with lateral offsets measured in feet ("milepointWithLateralOffsetInFeet" from [Annex A](#)). The distance expression:

```

<gmlr:distanceExpression>
  <gmlr:lateralOffsetDistanceExpression gml:id="DE002">
    <gmlr:distanceAlong>25</gmlr:distanceAlong>
    <gmlr:lateralOffsetExpression lateralOffsetReferent="back of curb">
      5
    </gmlr:lateralOffsetExpression>
  </gmlr:lateralOffsetDistanceExpression>
</gmlr:distanceExpression>

```

would specify a location that is 25 miles along the linear element then offset 5 feet behind the back of curb.

9.5.7 VerticalOffsetExpressionType

`gmlr:VerticalOffsetExpressionType` implements ISO 19148 LRO_VericalOffsetExpression (see ISO 19148:2012, 6.4.8) and LRO_VericalOffsetReferent (see ISO 19148:2012, 6.4.9).

`gmlr:VerticalOffsetExpressionType` is used to describe the vertical offset for a linearly referenced location.

```

<complexType name="VerticalOffsetExpressionType">
  <simpleContent>

```

```

<extension base="double">
  <attribute name="uom" type="gml:UomIdentifier"/>
  <attribute name="verticalOffsetReferent" type="string"/>
  <attribute name="featureGeometry" type="anyURI">
    <annotation>
      <appinfo>
        <gml:targetElement>
          gmlr:AbstractGeometry
        </gml:targetElement>
      </appinfo>
    </annotation>
  </attribute>
</extension>
</simpleContent>
</complexType>

```

The vertical offset distance is the measure of the vertical offset of the distance expression. This is the distance above or below the vertical offset referent (or above or below the linear element being measured if no vertical offset referent is specified) to the position being specified. A positive (+) value is measured in the direction specified by the positive vertical offset direction of the Linear Referencing Method. A value of 0 (zero) is to be interpreted as not having a vertical displacement from the vertical offset referent (or the linear element being measured if no vertical offset referent is specified). The vertical offset distance is a double value extended with three attributes.

The optional attribute `gmlr:uom` specifies the units of measure of the double value. It is of type `gml:UomIdentifier`. If none is provided, the `uom` value defaults to the `offsetUnits` value of the Linear Referencing Method.

The optional attribute `gmlr:verticalOffsetReferent` indicates the base line for the vertical offset measure.

`gmlr:verticalOffsetReferent` specifies the vertical offset referent as a character string. Having only a value of "existing ground at lateral offset" for example, allows a location to be specified as 5 feet above the existing ground at the lateral offset (for example, back of the curb). This aides in locating the position in the field without having to measure from the linear element.

If the curb is represented as a Feature with a spatial representation, the height of the curb is determinate and therefore the exact spatial position of the position expression can be calculated. Because the Feature can have multiple spatial representations, the `gml:Geometry`, rather than the Feature, is specified.

The optional attribute `gmlr:featureGeometry` is the geometry of a Feature. Though the type is `anyURI`, it is expected that the value will be a reference to an already defined `gml:AbstractGeometry` object.

9.5.8 LateralOffsetDirectionType

`gmlr:LateralOffsetDirectionType` implements ISO 19148 LRO_LateralOffsetDirection (see ISO 19148:2012, 6.4.3).

`gmlr:LateralOffsetDirectionType` is an enumeration of pre-defined lateral offset directions.

```

<simpleType name="LateralOffsetDirectionType">
  <restriction base="string">
    <enumeration value="left"/>
    <enumeration value="right"/>
  </restriction>
</simpleType>

```

The lateral offset direction is as viewed from above the linear element facing in the direction of increasing measure. If a from referent and a towards referents have both been specified, then the offset direction is as viewed from above the from referent facing in the direction of the towards referent.

9.5.9 VerticalOffsetDirectionType

`gml:VerticalOffsetDirectionType` implements ISO 19148 LRO_VericalOffsetDirection (see ISO 19148:2012, 6.4.4).

`gml:VerticalOffsetDirectionType` is an enumeration of pre-defined vertical offset directions.

```
<simpleType name="VerticalOffsetDirectionType">
  <restriction base="string">
    <enumeration value="up"/>
    <enumeration value="down"/>
  </restriction>
</simpleType>
```

The vertical offset direction is as viewed from above the linear element facing in the direction of increasing measure. If a from referent and a towards referents have both been specified, then the offset direction is as viewed from above the from referent facing in the direction of the towards referent.

9.5.10 LateralOffsetLinearSRS

`gmlro:LateralOffsetLinearSRS` specifies a Linear Spatial Reference System as a combination of a linear element and a Linear Referencing Method With Offset, the first two components of a `gml:PositionExpression`.

```
<complexType name="LateralOffsetLinearSRSType">
  <complexContent>
    <extension base="gml:IdentifiedObjectType">
      <sequence>
        <element name="linearElement"
          type="gml:LinearElementPropertyType"/>
        <element name="lrm"
          type="gmlro:LRMWithOffsetPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="LateralOffsetLinearSRS"
  type="gmlro:LateralOffsetLinearSRSType"
  substitutionGroup="gml:Definition"/>
```

The element `gmlro:linearElement` specifies the linear element being measured.

The element `gmlro:lrm` specifies the linear referencing method of measurement.

This combination of a linear element and a Linear Referencing Method, identifiable by a `gml:id`, can be used as an SRS. Lateral and vertical offsets can be accommodated with the following assumptions:

- 1) the Linear Referencing Method type is either "absolute" or "interpolative"
- 2) the Linear Referencing Method supports offsets
- 3) the first double value in `gml:pos` is the distance along
- 4) the second double value in `gml:pos`, if present, is the offset lateral distance
- 5) if an offset vertical distance is present, it is the third double value in `gml:pos`, thus requiring the presence of a (possibly zero valued) offset lateral distance
- 6) if multiple positions are required (e.g., for defining a `gml:LineString`), multiple `gml:pos` are used instead of a `gml:posList`.

EXAMPLE 1

```
<gml:Point gml:id="p2" srsName="#LSRS234">
  <gml:pos>25 30 5</gml:pos>
```

```
</gml:Point>
```

defines a Point geometry as a distance along (25), an offset lateral distance (30) and an offset vertical distance (5) from the linear element encapsulated in LSRS234, measured in accordance with the Linear Referencing Method which is also encapsulated in LSRS234:

```
<gml:ro:LateralOffsetLinearSRS gml:id="LSRS234">
  <gml:ro:linearElement>
    <gml:ro:LinearElement gml:id="LE005">
      <gml:ro:curve>
        <gml:ro:LineString gml:id="LS001">
          <gml:ro:posList>407829 268621 23.93
            407415 268600 8.43</gml:ro:posList>
        </gml:ro:LineString>
      </gml:ro:curve>
      <gml:ro:defaultLRM>
        <gml:ro:LinearReferencingMethod gml:id="LM001">
          <gml:ro:name>milepoint</gml:ro:name>
          <gml:ro:type>absolute</gml:ro:type>
          <gml:ro:units>mile</gml:ro:units>
        </gml:ro:LinearReferencingMethod>
      </gml:ro:defaultLRM>
      <gml:ro:measure uom="mile">100</gml:ro:measure>
      <gml:ro:startValue lrm="LRM001">0</gml:ro:startValue>
    </gml:ro:LinearElement>
  </gml:ro:linearElement>
  <gml:ro:lrms>
    <gml:ro:LRMWithOffset gml:id="LRM002">
      <gml:ro:name>chainage</gml:ro:name>
      <!--chainage = measurement in metres -->
      <gml:ro:type>absolute</gml:ro:type>
      <!--absolute = measure from start of linear element -->
      <gml:ro:units>metre</gml:ro:units>
      <gml:ro:offsetUnits>metre</gml:ro:offsetUnits>
      <!--positiveLateralOffsetDirection defaults to
        "right" -->
      <!--positiveVerticalOffsetDirection defaults to "up" -->
    </gml:ro:LRMWithOffset>
  </gml:ro:lrms>
</gml:ro:LateralOffsetLinearSRS>
```

EXAMPLE 2

```
<gml:LineString gml:id="fence101" srsName="#LSRS345">
  <gml:pos>15.5 35</gml:pos>
  <gml:pos>16.2 37</gml:pos>
</gml:LineString>
```

defines a line string geometry for a fence relative to a road centerline. The road is the locating feature linear element encapsulated in LSRS345. The absolute Linear Referencing Method With Offset called "milepoint with offsets in feet", also encapsulated in LSRS345, specifies units for distance along in miles, units for offsets in feet, and positive offset measures to the right. The fence is therefore 35 feet right of the road centerline at a location 15.5 miles along the road measured from the start of the road. The fence continues to a point which is 37 feet right of the road centerline at a location 16.2 miles along the road.

9.5.11 LateralOffsetLinearSRSPROPERTYTYPE

```
<complexType name="LateralOffsetLinearSRSPROPERTYTYPE">
  <sequence minOccurs="0">
    <element ref="gml:ro:LateralOffsetLinearSRS"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>
```

9.6 Linear Referencing Offset Vectors

9.6.1 Target namespace

All schema components specified in 9.6 are in the target namespace:

<http://www.opengis.net/gml/3.3/lrov>

9.6.2 Introduction

Linear Referencing Offset Vector includes the specification of linearly referenced locations which can have a vector offset in their distance expression.

9.6.3 VectorOffsetDistanceExpressionType

`gmlrov:VectorOffsetDistanceExpressionType` implements ISO 19148 LROV_VectorOffsetDistanceExpression (see ISO 19148:2012, 6.5.2).

A `gmlrov:VectorOffsetDistanceExpressionType` specifies a linear referenced measure value which may contain a vector offset value. It extends the `gmlr:DistanceExpressionType` with vector offsets.

```
<complexType name="VectorOffsetDistanceExpressionType">
  <complexContent>
    <extension base="gmlr:DistanceExpressionType">
      <sequence>
        <element name="vectorOffsetExpression"
          type="gmlrov:VectorOffsetExpressionType" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="VectorOffsetDistanceExpression"
  type="gmlrov:VectorOffsetDistanceExpressionType"
  substitutionGroup="gmlr:DistanceExpression"/>
```

As an alternative to lateral offsets, vector offsets extend the definition of a position expression to accommodate locations which do not lie directly on the linear element, but instead are located some distance from the linear element specified by a vector.

The optional element `gmlrov:VectorOffsetExpression` specifies the vector offset expression of the distance expression. If absent, the position is not displaced from the `gmlr:LinearElement`. A vector offset expression is only appropriate for Linear Referencing Method With Offset. If a distance expression has either a lateral or vertical offset expression, it cannot also have a vector offset expression.

9.6.4 VectorOffsetExpressionType

`gmlrov:VectorOffsetExpressionType` implements ISO 19148 LROV_VectorOffsetExpression (see ISO 19148:2012, 6.5.3) with a minor extension - the cardinality of the `offsetVector` attribute has been increased from [1] to [1..3] to enable the offset direction to be defined in terms of up to three component base offset vectors.

`gmlrov:VectorOffsetExpressionType` is used to describe the vector offset for a linearly referenced location.

```
<complexType name="VectorOffsetExpressionType">
  <sequence>
    <element name="offsetVector" type="gml:VectorType"/>
  </sequence>
</complexType>
```

The element `gml:rov:offsetVector` is the offset vector of the distance expression. This specifies the distance and direction of the offset from the linear element being measured to the position being specified. A 0 (zero) length vector is to be interpreted as not having a vector displacement from the linear element being measured.

The attribute `srsName`, inherited from `gml:VectorType` specifies the offset vector Coordinate Reference System.

9.6.5 VectorOffsetLinearSRS

`gml:rov:VectorOffsetLinearSRS` specifies a Linear Spatial Reference System as a combination of a linear element and a Linear Referencing Method With Offset, the first two components of a `gml:PositionExpression`. It also contains one or more offset vectors to specify the offset reference frame (examples are provided below).

```
<complexType name="VectorOffsetLinearSRSType">
  <complexContent>
    <extension base="gml:LinearSRSType">
      <sequence>
        <element name="linearElement"
          type="gml:LinearElementPropertyType"/>
        <element name="lrm" type="gml:LRMWithOffsetPropertyType"/>
        <element name="offsetVector" type="gml:rov:VectorType"
          maxOccurs="3"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="VectorOffsetLinearSRS"
  type="gml:rov:VectorOffsetLinearSRSType"
  substitutionGroup="gml:LinearSRS"/>

<complexType name="VectorType">
  <complexContent>
    <extension base="gml:VectorType">
      <attribute name="offsetUom" type="gml:UomIdentifier"/>
    </extension>
  </complexContent>
</complexType>
```

The element `gml:rov:linearElement` specifies the linear element being measured.

The element `gml:rov:lrm` specifies the linear referencing method of measurement.

The element `gml:rov:offsetVector` specifies the offset vector, that is, the direction in which the offset distance will be measured.

The attribute `srsName`, inherited from `gml:VectorType`, specifies the offset vector Coordinate Reference System.

The `gml:rov:VectorType` extends `gml:VectorType` with an optional attribute `gml:rov:offsetUom`, which specifies the units of measure of the vector distance. It is of type `gml:UomIdentifier`. If none is provided, the `uom` value defaults to the `offsetUnits` value of the Linear Referencing Method With Offset.

This combination of a linear element, a Linear Referencing Method, an offset vector and an optional vector CRS referenced by the `srsName` attribute on the `offsetVector` element, can be used as an SRS. Vector offsets can be accommodated in `gml:pos` or `gml:posList` expressions with the following assumptions:

- 1) the Linear Referencing Method type is either "absolute" or "interpolative"
- 2) the Linear Referencing Method supports offsets
- 3) the first double value in `gml:pos` (or `gml:posList`) is the distance along the linear element

- 4) subsequent double values in `gml:pos` (or `gml:posList`), if present, correspond to the component distance along the offset vector elements in document order as specified in the `gmlrov:VectorOffsetLinearSRSType`.

EXAMPLE 1 Single Offset Vector

A single offset vector can be used to describe the positions P1 to P11 relative to a linear element in the direction of the offset vector v as illustrated in [Figure 7](#).

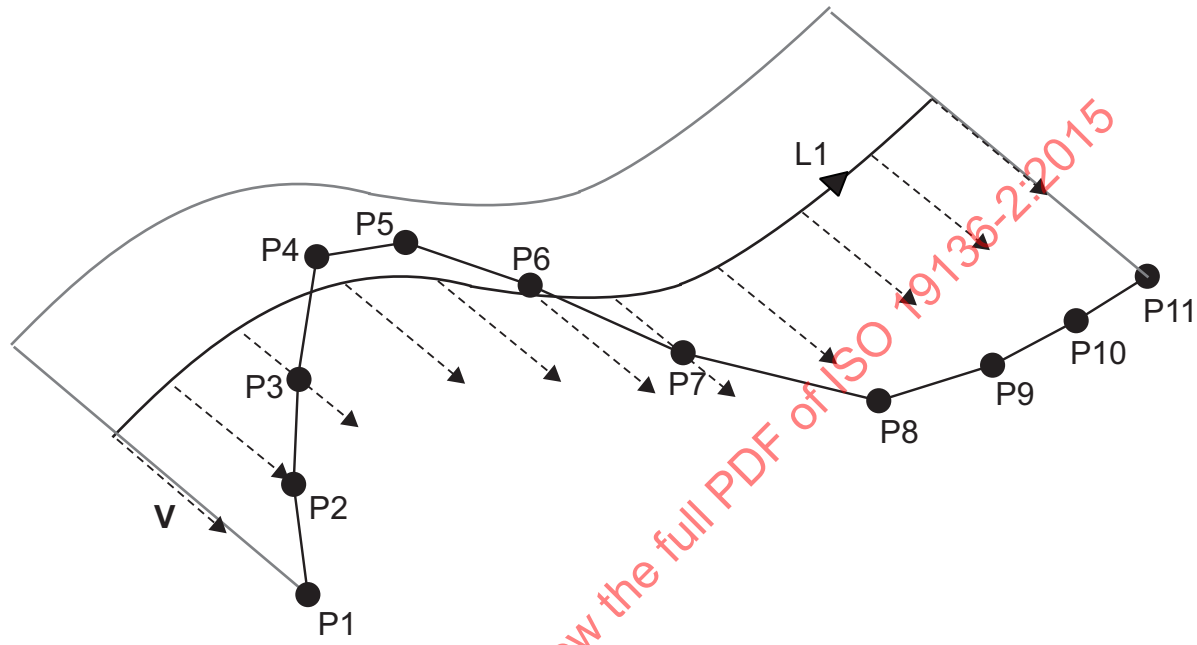


Figure 7— Single offset vector

Sample `gml:Point` encodings of the positions P1 to P11 are shown as follows, where the first ordinate in each `gml:pos` element corresponds to distance along the linear element L1 and the second ordinate in each `gml:pos` element corresponds to the distance (in units specified by `offsetUom`) in the direction of the offset vector v .

```
<gml:Point gml:id="P1" srsDimension="2" srsName="#volsrs001">
  <gml:pos>0 1.7</gml:pos>
</gml:Point>
<gml:Point gml:id="P2" srsDimension="2" srsName="#volsrs001">
  <gml:pos>1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="P3" srsDimension="2" srsName="#volsrs001">
  <gml:pos>2 0.6</gml:pos>
</gml:Point>
<gml:Point gml:id="P4" srsDimension="2" srsName="#volsrs001">
  <gml:pos>3 -0.3</gml:pos>
</gml:Point>
<gml:Point gml:id="P5" srsDimension="2" srsName="#volsrs001">
  <gml:pos>4 -0.3</gml:pos>
</gml:Point>
<gml:Point gml:id="P6" srsDimension="2" srsName="#volsrs001">
  <gml:pos>5 -0.1</gml:pos>
</gml:Point>
<gml:Point gml:id="P7" srsDimension="2" srsName="#volsrs001">
  <gml:pos>6 0.7</gml:pos>
</gml:Point>
<gml:Point gml:id="P8" srsDimension="2" srsName="#volsrs001">
  <gml:pos>7 1.5</gml:pos>
</gml:Point>
<gml:Point gml:id="P9" srsDimension="2" srsName="#volsrs001">
  <gml:pos>8 1.7</gml:pos>
```

```

</gml:Point>
<gml:Point gml:id="P10" srsDimension="2" srsName="#volsrs001">
  <gml:pos>9 1.8</gml:pos>
</gml:Point>
<gml:Point gml:id="P11" srsDimension="2" srsName="#volsrs001">
  <gml:pos>10 1.9</gml:pos>
</gml:Point>

<gml:rov:VectorOffsetLinearSRS gml:id="volsrs001"
  xmlns:gmlr="http://www.opengis.net/gml/3.3/lr"
  xmlns:gmlrov="http://www.opengis.net/gml/3.3/lrov"
  xmlns:gml="http://www.opengis.net/gml/3.2">
  <gml:identifier codeSpace="...">...</gml:identifier>
  <gmlr:linearElement xlink:href="#L1"
  xlink:title="LinearElement"/>
  <gmlr:lrml xlink:href="#lrml0001"
  xlink:title="LinearReferencingMethod"/>
  <gmlrov:offsetVector
  srsName="http://www.opengis.net/def/crs/EPSSG/0/7405"
  offsetUom="m">0 1 0</gmlrov:offsetVector>
</gml:rov:VectorOffsetLinearSRS>

<gmlr:LinearReferencingMethod gml:id="lrml0001"
  xmlns:gmlr="http://www.opengis.net/gml/3.3/lr">
  <gmlr:name>chainage</gmlr:name>
  <gmlr:type>absolute</gmlr:type>
  <gmlr:units>m</gmlr:units>
</gmlr:LinearReferencingMethod>

```

EXAMPLE 2 Two Offset Vectors

A basis of two offset vectors $\mathbf{v1}$ and $\mathbf{v2}$ can be used to describe the positions S1 to S11 relative to a linear element L2 in the offset reference frame as illustrated in Figure 8. Sample gml:Point encodings of the positions S1 to S11 are shown adjacent to the Figure 8, where the first ordinate in each gml:pos element corresponds to distance along the linear element L2, the second ordinate in each gml:pos element corresponds to the component distance along the direction of offset vector $\mathbf{v1}$ and the third ordinate in each gml:pos element corresponds to the component distance along the direction of the offset vector $\mathbf{v2}$.

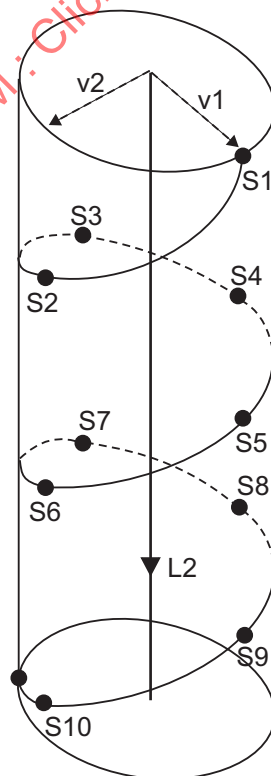


Figure 8 — Two offset vectors

```

<gml:Point gml:id="S1" srsDimension="3" srsName="#volsrs002">
  <gml:pos>0 1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="S2" srsDimension="3" srsName="#volsrs002">
  <gml:pos>1 0 1</gml:pos>
</gml:Point>
<gml:Point gml:id="S3" srsDimension="3" srsName="#volsrs002">
  <gml:pos>2 -1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="S4" srsDimension="3" srsName="#volsrs002">
  <gml:pos>3 0 -1</gml:pos>
</gml:Point>
<gml:Point gml:id="S5" srsDimension="3" srsName="#volsrs002">
  <gml:pos>4 1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="S6" srsDimension="3" srsName="#volsrs002">
  <gml:pos>5 0 1</gml:pos>
</gml:Point>
<gml:Point gml:id="S7" srsDimension="3" srsName="#volsrs002">
  <gml:pos>6 -1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="S8" srsDimension="3" srsName="#volsrs002">
  <gml:pos>7 0 -1</gml:pos>
</gml:Point>
<gml:Point gml:id="S9" srsDimension="3" srsName="#volsrs002">
  <gml:pos>8 1 0</gml:pos>
</gml:Point>
<gml:Point gml:id="S10" srsDimension="3" srsName="#volsrs002">
  <gml:pos>9 0 1</gml:pos>
</gml:Point>
<gml:Point gml:id="S11" srsDimension="3" srsName="#volsrs002">
  <gml:pos>9.3 -0.45 0.89</gml:pos>
</gml:Point>

<gml:rov:VectorOffsetLinearSRS gml:id="volsrs002"
  xmlns:gmlr="http://www.opengis.net/gml/3.3/lr"
  xmlns:gmlrov="http://www.opengis.net/gml/3.3/lrov"
  xmlns:gml="http://www.opengis.net/gml/3.2">
  <gml:identifier codeSpace="..."></gml:identifier>
  <gmlr:linearElement xlink:href="#L2"
    xlink:title="LinearElement"/>
  <gmlr:lrm xlink:href="#lrm0001"
    xlink:title="LinearReferencingMethod"/>
  <gmlrov:offsetVector
    srsName="http://www.opengis.net/def/crs/EPSSG/0/7405"
    offsetUom="m">0 1 0</gmlrov:offsetVector>
  <gmlrov:offsetVector
    srsName="http://www.opengis.net/def/crs/EPSSG/0/7405"
    offsetUom="m">0 0 1</gmlrov:offsetVector>
</gml:rov:VectorOffsetLinearSRS>

```

9.6.6 VectorOffsetLinearSRSPROPERTYTYPE

```

<complexType name="VectorOffsetLinearSRSPROPERTYTYPE">
  <sequence minOccurs="0">
    <element ref="gml:rov:VectorOffsetLinearSRS"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
</complexType>

```

9.7 Requirements classes

[Table 8](#) specifies the requirements class "Linear Referencing Systems".

Table 8 — Requirements class “Linear Referencing Systems”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/lr	
Target type	Data instance
Name	Linear Referencing Systems
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/lr/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lr schema SHALL be well-formed and valid.

[Table 9](#) specifies the requirements class “Linear Referencing Towards Referents”.

Table 9 — Requirements class “Linear Referencing Towards Referents”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/lrtr	
Target type	Data instance
Name	Linear Referencing Towards Referents
Dependency	http://www.opengis.net/spec/GML/3.3/req/lr
Requirement	http://www.opengis.net/spec/GML/3.3/req/lrtr/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lrtr schema SHALL be well-formed and valid.

[Table 10](#) specifies the requirements class “Linear Referencing Offsets”.

Table 10 — Requirements class “Linear Referencing Offsets”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/lro	
Target type	Data instance
Name	Linear Referencing Offsets
Dependency	http://www.opengis.net/spec/GML/3.3/req/lr
Requirement	http://www.opengis.net/spec/GML/3.3/req/lro/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lro schema SHALL be well-formed and valid.

[Table 11](#) specifies the requirements class “Linear Referencing Vector Offsets”.

Table 11 — Requirements class “Linear Referencing Vector Offsets”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/lrov	
Target type	Data instance
Name	Linear Referencing Vector Offsets
Dependency	http://www.opengis.net/spec/GML/3.3/req/lro
Requirement	http://www.opengis.net/spec/GML/3.3/req/lrov/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lrov schema SHALL be well-formed and valid.

9.8 Conformance

[Table 12](#) specifies the conformance class “Linear Referencing Systems”.

Table 12 — Conformance class “Linear Referencing Systems”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/lr	
Requirements	http://www.opengis.net/spec/GML/3.3/req/lr
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Test	http://www.opengis.net/spec/GML/3.3/conf/lr/valid
	Requirement http://www.opengis.net/spec/GML/3.3/req/lr/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lr schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/lr.xsd . Pass if no errors reported. Fail otherwise.
	Test type Basic

[Table 13](#) specifies the conformance class “Linear Referencing Towards Referents”.

Table 13 — Conformance class “Linear Referencing Towards Referents”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/lrtr	
Requirements	http://www.opengis.net/spec/GML/3.3/req/lrtr
Dependency	http://www.opengis.net/spec/GML/3.3/conf/lr
Test	http://www.opengis.net/spec/GML/3.3/conf/lrtr
	Requirement http://www.opengis.net/spec/GML/3.3/req/lrtr/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lrtr schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/lrtr.xsd . Pass if no errors reported. Fail otherwise.
	Test type Capability

[Table 14](#) specifies the conformance class “Linear Referencing Offsets”.

Table 14 — Conformance class “Linear Referencing Offsets”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/lro	
Requirements	http://www.opengis.net/spec/GML/3.3/req/lro
Dependency	http://www.opengis.net/spec/GML/3.3/conf/lr
Test	http://www.opengis.net/spec/GML/3.3/conf/lro
	Requirement http://www.opengis.net/spec/GML/3.3/req/lro/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lro schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/lro.xsd . Pass if no errors reported. Fail otherwise.
	Test type Capability

[Table 15](#) specifies the conformance class “Linear Referencing Vector Offsets”.

Table 15 — Conformance class “Linear Referencing Vector Offsets”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/lrov	
Requirements	http://www.opengis.net/spec/GML/3.3/req/lrov
Dependency	http://www.opengis.net/spec/GML/3.3/conf/lro
Test	http://www.opengis.net/spec/GML/3.3/conf/lrov
	Requirement http://www.opengis.net/spec/GML/3.3/req/lrov/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/lrov schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/lrov.xsd . Pass if no errors reported. Fail otherwise.
	Test type Capability

10 ReferenceableGrid

10.1 Target namespace

All schema components specified in [Clause 10](#) are in the target namespace:

<http://www.opengis.net/gml/3.3/rgrid>

10.2 Clarifications to ISO 19123:2005

The CV_ReferenceableGrid class in ISO 19123:2005(see [8.10](#) and [Figure 17](#)) assume the existence of a coordinate transformation, as defined by ISO 19111:2007 by which CV_ReferenceableGrid is referenced. However, this grid may be, and often is, referenced by a concatenated coordinate operation, as defined by ISO 19111:2007

Of course, multiple coordinate transformations or concatenated operations may exist for the same image, that are referenced to more than one other CRS, and/or are referenced to the same other CRS by multiple Transformation versions (with different accuracy characteristics). However, 5.3.10 of Topic 6 requires that if a coverage domain is defined in a different coordinate reference system, the domain is different so the coverage shall be different. Therefore, if multiple coordinate transformation or concatenated operations exist for the same image range values, these are encoded as different grid coverages.

10.3 AbstractReferenceableGrid

A referenceable grid is associated with a transform between grid coordinates and coordinates in an external coordinate reference system. Unlike a rectified grid, this transformation is usually not characterised through an affine transformation. The grid curves in the external coordinate reference system need not be straight or orthogonal, but they should retain a grid topology (i.e. the sets of curves defining the grid network should not intersect within themselves). Grid cells may be of different shapes and sizes.

NOTE 1 Notwithstanding some ambiguity in ISO 19123, a `gml:Grid` represents a geometry, and is always associated with an 'internal' grid coordinate system (with integer coordinates and a zero origin). A grid that is also defined in terms of an 'external' coordinate reference system is a rectified grid (GML 3.2 19.2.3), if there is an affine relationship between the internal and external coordinates, or a referenceable grid as specified in 10.3 otherwise. The external coordinate reference system for both these subclasses is provided through the inherited `gml:SRSReferenceGroup` (including the `srsName` attribute). The use of `gml:SRSReferenceGroup` for `gml:Grid` is not defined since it has no analogue in ISO 19123.

In ISO 19123, the external coordinate reference system for a `CV_ReferenceableGrid` is defined (ISO 19123:2005, 8.10.2) through an explicit association *Coordinate Reference System* with role name *crs* to an ISO 19111 SC_CRS coordinate reference system object. For a `CV_RectifiedGrid`, it is defined implicitly (ISO 19123:2005, 8.9.2) as the coordinate reference system of the origin.

NOTE 2 A grid topology is retained automatically through the affine transformation of a `RectifiedGrid`.

EXAMPLE [Figure 9](#) shows an example of a referenceable grid.

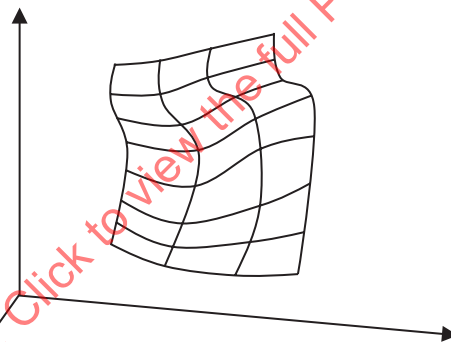


Figure 9 — ReferenceableGrid example

The transformation between grid coordinates and 'external' coordinates is usually non-affine and may take different forms. It may be described analytically (e.g. through parametrised mathematical image rectification models); or it may be provided in a table, for instance, relating the grid points to coordinates in the external coordinate reference system (ISO 19123:2005, 8.2.1). The method of transformation is an implementation detail not specified by ISO 19123.

The encoding of `CV_ReferenceableGrid` therefore uses an abstract element, `gmlrgrid:AbstractReferenceableGrid`, as the head of a substitution group that may include various concrete implementations of the transformation.

```
<complexType abstract="true" name="AbstractReferenceableGridType">
  <complexContent>
    <extension base="gml:GridType">
      <sequence>
        <element ref="gmlrgrid:gridCRS" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element abstract="true" name="AbstractReferenceableGrid"
  substitutionGroup="gml:Grid">
```

```
type="gmlrgrid:AbstractReferenceableGridType"/>
```

The attribute group `gml:SRSReferenceGroup` inherited from `gml:AbstractGeometry` shall link the referenceable grid to the external coordinate reference system to which it is referenceable (represented in ISO 19123 as an explicit association *Coordinate Reference System* with role name *crs* to an ISO 19111 SC_CRS coordinate reference system object).

The optional `gridCRS` property is a composition association to the definition of the grid CRS for this `AbstractReferenceableGrid`. This definition should be included here when not otherwise known to all data receivers, especially when extended by `ReferenceableGridByTransformation` which uses a coordinate Transformation or Concatenated Operation that references a `gridCRS`.

A property type for the `gmlrgrid:AbstractReferenceableGrid` is also defined:

```
<complexType name="ReferenceableGridPropertyType">
  <sequence>
    <element ref="gmlrgrid:AbstractReferenceableGrid"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
  <attributeGroup ref="gml:OwnershipAttributeGroup"/>
</complexType>

<element name="referenceableGridProperty"
type="gmlrgrid:ReferenceableGridPropertyType"/>
```

10.4 ReferenceableGridByArray

The `gmlrgrid:ReferenceableGridByArrayType` defines a referenceable grid by listing an array of grid point locations explicitly, as a sequence of direct positions in a defined sequence order over the grid. It is defined as follows:

```
<complexType name="ReferenceableGridByArrayType">
  <complexContent>
    <extension base="gmlrgrid:AbstractReferenceableGridType">
      <sequence>
        <group ref="gml:geometricPositionListGroup"/>
        <element name="sequenceRule"
          type="gml:SequenceRuleType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="ReferenceableGridByArray"
  substitutionGroup="gmlrgrid:AbstractReferenceableGrid"
  type="gmlrgrid:ReferenceableGridByArrayType"/>
```

The element group `gml:geometricPositionListGroup` specifies the array of grid point locations in the external coordinate reference system (through either a `gml:posList` element, or a sequence of `gml:pos` elements or `gml:Point` objects). The `gml:sequenceRule` element specifies the sequence order of these grid point locations over the grid.

EXAMPLE The referenceable grid of [Figure 10](#) can be represented by specifying an array of grid point locations explicitly as follows (note that coordinates are specified as (Lat, Long) in EPSG 4326):

```
<gmlrgrid:ReferenceableGridByArray gml:id="ex" dimension="2"
srsName="http://www.opengis.net/def/crs/EPSPG/0/4326">
  <gml:limits>
    <gml:GridEnvelope>
      <gml:low>0 0</gml:low>
      <gml:high>4 3</gml:high>
    </gml:GridEnvelope>
  </gml:limits>
  <gml:axisLabels>x y</gml:axisLabels>
  <gml:posList>
    2 8 3 10 6 12 8 14 10 18
    4 6 6 8 8 12 10 14 12 16
```

```

        6 2 7  4  9  6 10  8 13 12
        8 2 8  3 10  5 11  8 13 10
    </gml:posList>
    <gml:sequenceRule axisOrder="+1 +2">Linear</gml:sequenceRule>
</gmlrgrid:ReferenceableGridByArray>

```

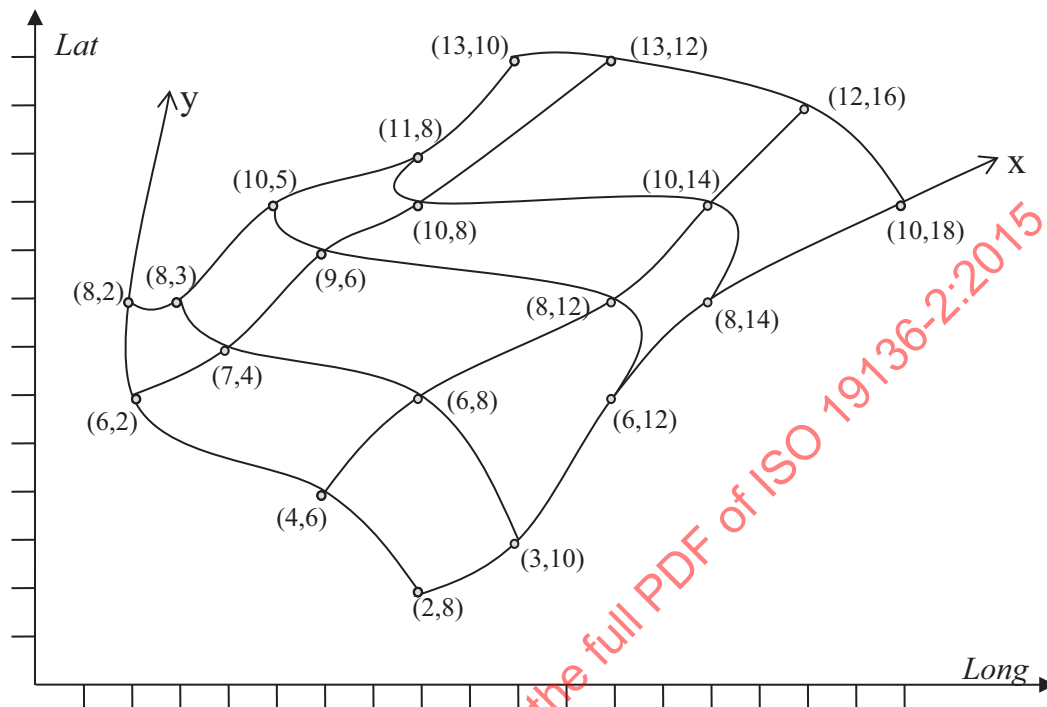


Figure 10 — ReferenceableGridByArray – grid point locations listed explicitly

10.5 ReferenceableGridByVectorsType, ReferenceableGridByVectors

The `gmlrgrid:ReferenceableGridByVectorsType` defines a referenceable grid by specifying an origin and offset vectors, with coefficients that may vary over the grid. This generalises the mechanism used for the RectifiedGrid (GML 3.2 19.2.3), which uses offset vectors with implicit scalar (unit) coefficients.

In fact, a hierarchy of grid types may be represented using this mechanism (Figure 11). A RectifiedGrid corresponds to offset vectors with scalar (unit integer) coefficients; an irregular but rectilinear grid corresponds to real-valued coefficient vectors; while a completely irregular (curvilinear) grid corresponds to coefficient arrays.

NOTE 1 The latter two cases are both examples of an ISO 19123 CV_ReferenceableGrid.

NOTE 2 The conceptual hierarchy of Figure 11 is not required by the model presented here, and is different to ISO 19123, where CV_RectifiedGrid is not regarded as a specialisation of CV_ReferenceableGrid. There may be merit in revisiting the relationship between these classes in any future revision of ISO 19123.

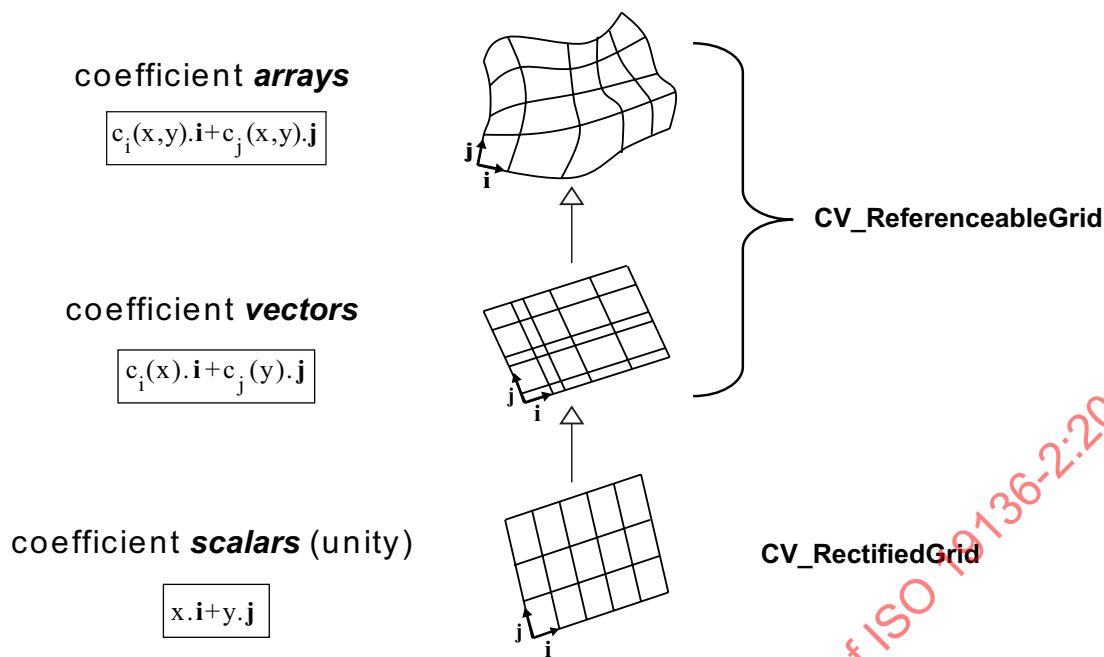


Figure 11 — Hierarchy of grids with different offset vector coefficients (x and y range over the integer grid point indices)

The `gmlrgrid:ReferenceableGridByVectorsType` is defined as follows:

```
<complexType name="ReferenceableGridByVectorsType">
  <complexContent>
    <extension base="gmlrgrid:AbstractReferenceableGridType">
      <sequence>
        <element name="origin" type="gml:PointPropertyType"/>
        <element maxOccurs="unbounded" name="generalGridAxis"
          type="gmlrgrid:GeneralGridAxisPropertyType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<element name="ReferenceableGridByVectors"
substitutionGroup="gmlrgrid:AbstractReferenceableGrid"
  type="gmlrgrid:ReferenceableGridByVectorsType"/>
```

The element `gmlrgrid:origin` represents the origin of the referenceable grid in the external coordinate reference system.

The set of `gmlrgrid:generalGridAxis` elements provide offset vectors (with corresponding coefficients) that together span the (sub-)space containing the convex hull of the referenceable grid. A grid point location in the external coordinate reference system is calculated from the sum of the offset vectors from all `gmlrgrid:generalGridAxis` elements weighted by their respective coefficients for the grid point concerned.

The `gmlrgrid:GeneralGridAxisType` specifies an offset vector and array of coefficients over all grid points. Both an element and property-type are defined in order to allow the GML by-reference pattern (GML 3.2, 7.2.3.3):

```
<complexType name="GeneralGridAxisType">
  <sequence>
    <element name="offsetVector" type="gml:VectorType"/>
    <element name="coefficients" type="gml:doubleList"/>
    <element name="gridAxesSpanned" type="gml:NCNameList"/>
    <element name="sequenceRule" type="gml:SequenceRuleType"/>
  </sequence>
```

```

</complexType>

<element name="GeneralGridAxis" type="gmlrgrid:GeneralGridAxisType"/>

<complexType name="GeneralGridAxisPropertyType">
  <sequence minOccurs="0">
    <element ref="gmlrgrid:GeneralGridAxis"/>
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup"/>
  <attributeGroup ref="gml:OwnershipAttributeGroup"/>
</complexType>

```

The element `gmlrgrid:offsetVector` specifies a vector in the external coordinate reference system. The element `gmlrgrid:coefficients` specifies a corresponding set of weighting coefficients over the grid points. In some cases these coefficients will be uniform along one or more axes of the grid, and for efficiency reasons the coefficients need only be specified along those axes on which they vary. The grid axes over which the coefficients vary is indicated with the `gmlrgrid:gridAxesSpanned` element, and the order of the coefficients over those grid axes is indicated using the `gmlrgrid:sequenceRule` element.

NOTE 3 A referenceable grid with each 'general grid axis' having unit coefficients over one grid axis is equivalent to a `RectifiedGrid`.

NOTE 4 A referenceable grid with each 'general grid axis' having a coefficient vector (i.e. coefficients varying over just one axis of the grid) is 'rectilinear' (i.e. parallel grid lines in the external coordinate reference system).

NOTE 5 Any referenceable grid may be embedded within a space of dimension greater than, or equal to, its grid dimension. For example, a two-dimensional grid may be embedded within a three-dimensional volume – either by lying in an oblique plane ([Figure 12](#)), or by lying in a surface curved within three dimensions ([Figure 13](#)). The latter case is defined by specifying three offset vectors. In general, the following constraints apply for any referenceable grid (analogous to those for rectified grids, ISO 19123:2005, 8.9.6):

$$(\text{grid dimension}) \leq (\text{number of offset vectors}) \leq (\text{CRS dimension})$$

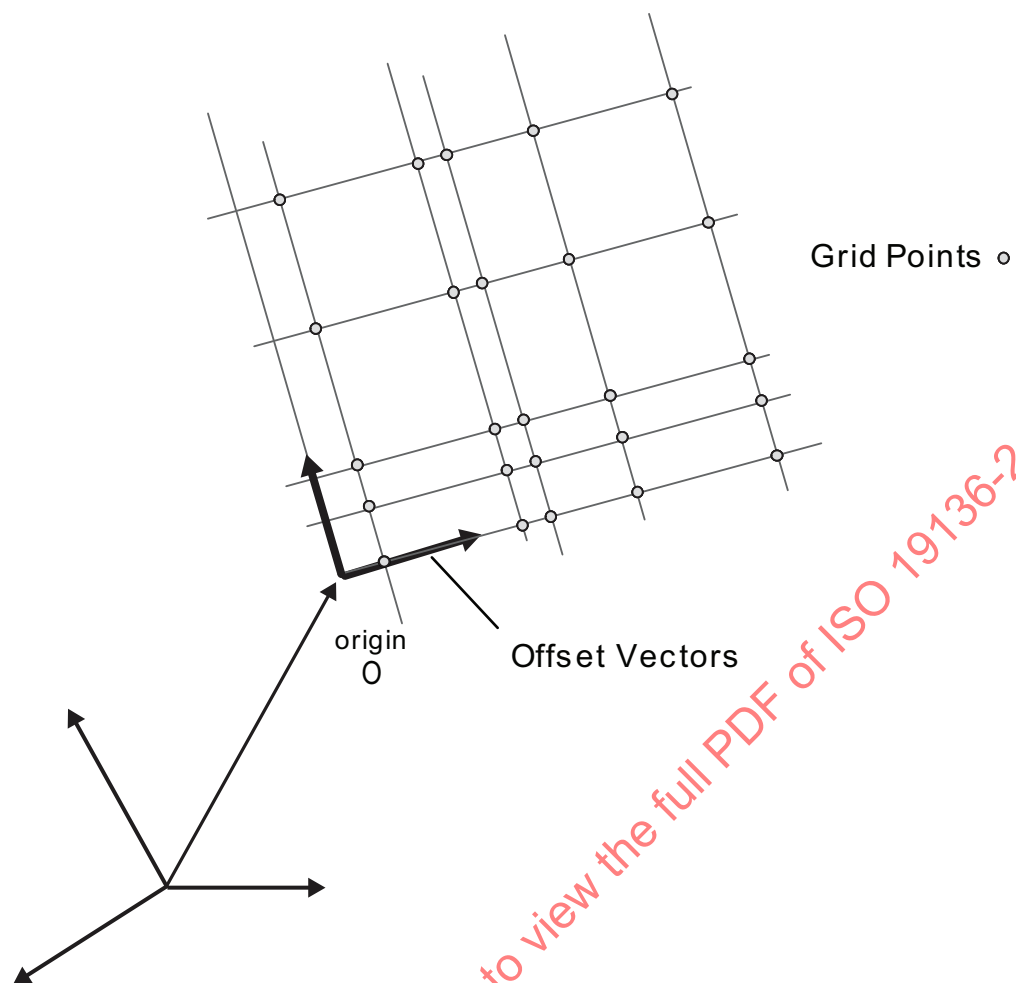


Figure 12 — ReferenceableGrid embedded in oblique plane (two offset vectors)

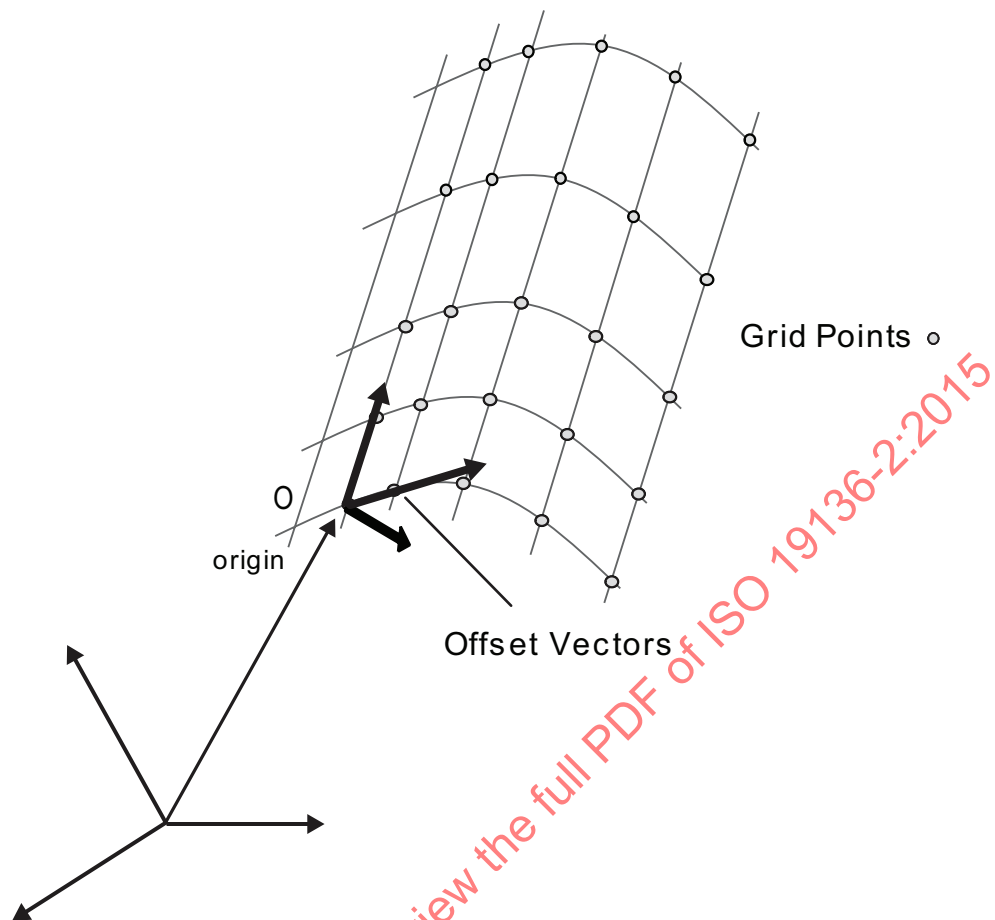


Figure 13 — ReferenceableGrid embedded in curved surface (three offset vectors)

EXAMPLE 1 The (rectilinear) referenceable grid of [Figure 14](#) requires coefficient vectors along each grid axis. It is represented as follows:

```
<ReferenceableGridByVectors gml:id="ex1" dimension="2" srsName="http://www.opengis.
net/def/crs/EPSG/0/4326">
  <limits>
    <GridEnvelope>
      <low>0 0</low>
      <high>4 3</high>
    </GridEnvelope>
  </limits>
  <axisLabels>x y</axisLabels>
  <origin>
    <gml:Point gml:id="o1.1">
      <pos>2 8</pos>
    </gml:Point>
  </origin>
  <generalGridAxis>
    <GeneralGridAxis>
      <offsetVector>1 2</offsetVector>
      <coefficients>0 1 3 6 7</coefficients>
      <gridAxesSpanned>x</gridAxesSpanned>
      <sequenceRule axisOrder="+1">Linear</sequenceRule>
    </GeneralGridAxis>
  </generalGridAxis>
  <generalGridAxis>
    <GeneralGridAxis>
      <offsetVector>1 -1</offsetVector>
      <coefficients>0 1 4 6</coefficients>
      <gridAxesSpanned>y</gridAxesSpanned>
      <sequenceRule axisOrder="+1">Linear</sequenceRule>
    </GeneralGridAxis>
  </generalGridAxis>
</ReferenceableGridByVectors>
```

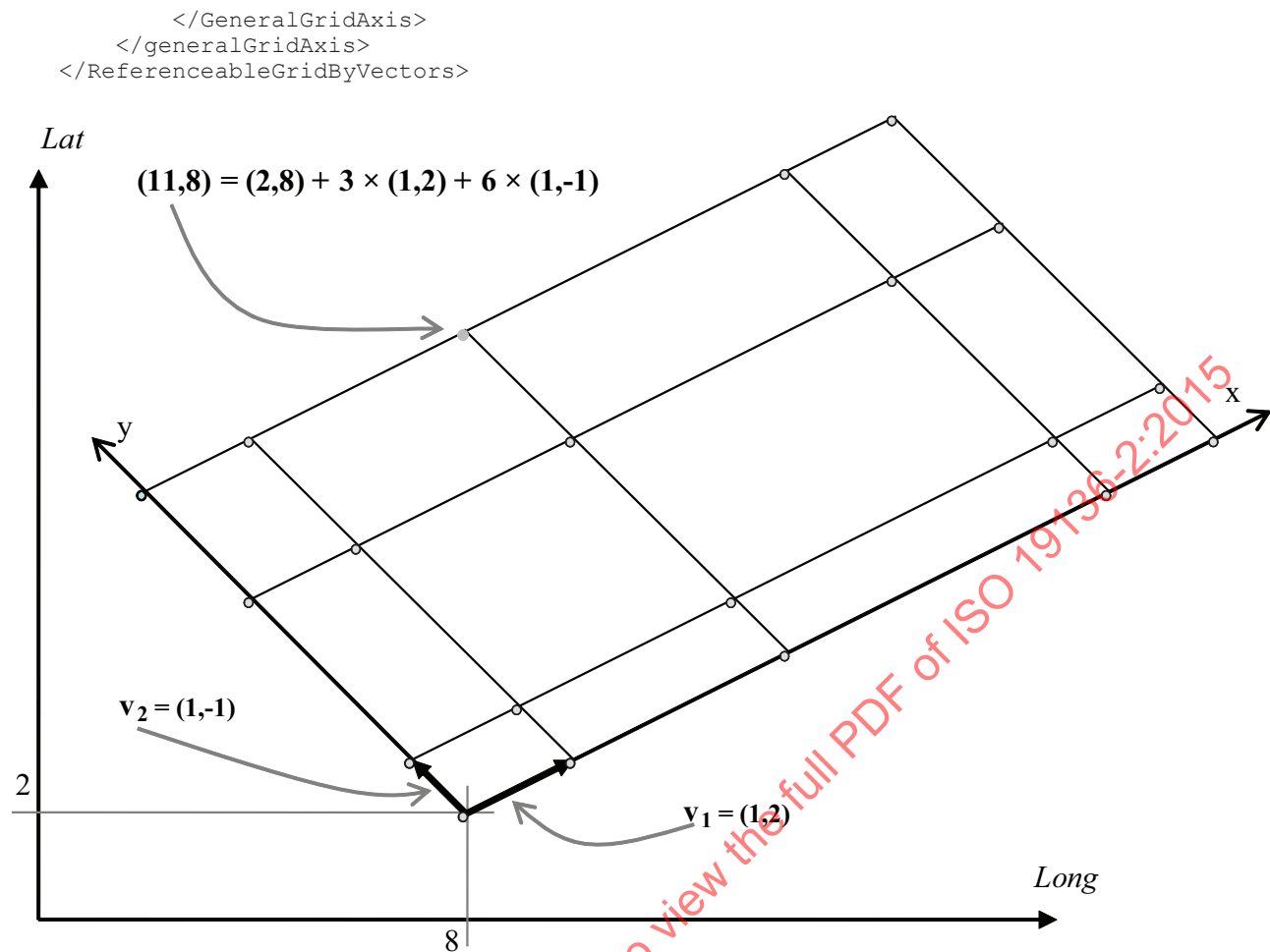


Figure 14 — ReferenceableGridByVectors - rectilinear grid with coefficient vectors

EXAMPLE 2 The (curvilinear) referenceable grid of [Figure 10](#) may be specified using coefficient arrays across both axes of the grid. It is represented as follows ([Figure 15](#)):

```

<ReferenceableGridByVectors gml:id="ex2" dimension="2" srsName="http://www.opengis.
net/def/crs/EPSG/0/4326">
  <limits>
    <GridEnvelope>
      <low>0 0</low>
      <high>4 3</high>
    </GridEnvelope>
  </limits>
  <axisLabels>x y</axisLabels>
  <origin>
    <gml:Point gml:id="o1.2">
      <pos>2 8</pos>
    </gml:Point>
  </origin>
  <generalGridAxis>
    <GeneralGridAxis>
      <offsetVector>0.333 0.667</offsetVector>
      <coefficients>
        0      3      8      12      18
        0      4      10     14      18
        -2     1      5      8       15
        0      1      5      9       13
      </coefficients>
      <gridAxesSpanned>x y</gridAxesSpanned>
      <sequenceRule axisOrder="+1 +2">Linear</sequenceRule>
    </GeneralGridAxis>
  </generalGridAxis>

```

```

</generalGridAxis>
<generalGridAxis>
  <GeneralGridAxis>
    <offsetVector>0.667 -0.667</offsetVector>
    <coefficients>
      0    0    2    3    3
      3    4    4    5    6
      7    7    8    8    9
      9    8.5  9.5  9    10
    </coefficients>
    <gridAxesSpanned>x y</gridAxesSpanned>
    <sequenceRule axisOrder="+1 +2">Linear</sequenceRule>
  </GeneralGridAxis>
</generalGridAxis>
</ReferenceableGridByVectors>

```

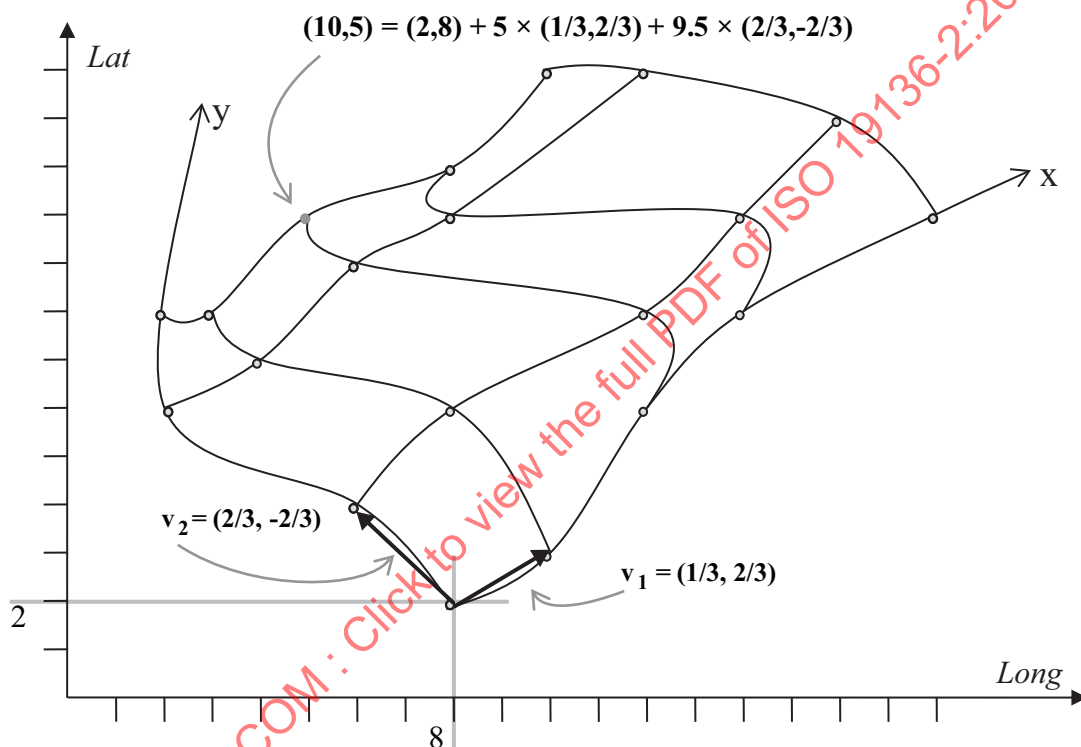


Figure 15 — ReferenceableGridByVectors – curvilinear grid with coefficient arrays

10.6 ReferenceableGridByTransformation

```

<element name="ReferenceableGridByTransformation"
type="gmlrgrid:ReferenceableGridByTransformationType"
substitutionGroup="gmlrgrid:AbstractReferenceableGrid"/>

<complexType name="ReferenceableGridByTransformationType">
  <complexContent>
    <extension base="gmlrgrid:AbstractReferenceableGridType">
      <choice>
        <element name="transformation"
          type="gml:TransformationPropertyType"/>
        <element name="concatenatedOperation"
          type="gml:ConcatenatedOperationPropertyType"/>
      </choice>
    </extension>
  </complexContent>
</complexType>

```

ReferenceableGridByTransformation encodes one type of CV_ReferenceableGrid as defined in ISO 19123, using a `gml:Transformation` or `gml:ConcatenatedOperation` to specify the relationship between positions in the grid and corresponding positions in another CRS.

The choice in ReferenceableGridByTransformation encodes an association to a coordinate Transformation or Concatenated Operation that (geo)references this ReferenceableGrid. This Transformation or Concatenated Operation shall reference the grid CRS for this `gml:AbstractReferenceableGrid`, as either its sourceCRS or targetCRS. If a grid is referenced to more than one other CRS, or is referenced to the same other CRS by multiple Transformation versions, different grids and different coverages shall be used.

10.7 gridCRS

```
<element name="gridCRS" type="gmlrgrid:GridCRSPropertyType"/>

<complexType name="GridCRSPropertyType">
  <sequence>
    <element ref="gml:AbstractCRS"/>
  </sequence>
</complexType>
```

A grid coordinate reference system for positions in a ReferenceableGrid that uses a defined coordinate system congruent with the coordinate system described by the GridEnvelope and axisLabels of `gml:GridType`. That is, this defined coordinate system shall have the same grid axis positions and origin as the GridEnvelope, with the same axisLabels, but need not define the same or any limits on the grid size. In addition to this defined coordinate system, a gridCRS shall have a defined datum, whose origin may be a specific grid point in a specific quadrilateral grid coverage that uses this gridCRS.

The gridCRS is a composition association to the definition of the CRS for this AbstractReferenceableGrid. This CRS may be a DerivedCRS, EngineeringCRS, ImageCRS, or CompoundCRS.

NOTE The proposed OGC Best Practices Paper 09-085r1 recommends that this gridCRS always be a DerivedCRS.

10.8 Coverages using ReferenceableGrid

The OGC standard "GML Application Schema – Coverages", version 1.0, OGC document 09-146r1, provides an element ReferenceableGridCoverage as a GML implementation of CV_DiscreteGridPointCoverage where the domain is one of the GML implementations of CV_ReferenceableGrid specified in [Clause 10](#). Since such an element is provided already in that GML application schema for coverages, no new coverage element is provided in this part of ISO 19136.

[Table 16](#) extends GML 3.2, ISO 19136:2007, Table D.8, to include ISO 19123 CV_ReferenceableGrid.

Table 16 — Description of the profile of ISO 19123 (GML 3.3 extension)

ISO 19123 construct	GML construct
CV_ReferenceableGrid	ReferenceableGrid
coordTransform (operation)	GridPointList and GeneralGridAxis

10.9 Requirements classes

[Table 17](#) specifies the requirements class "Referenceable Grids (Base)".

Table 17 — Requirements class “Referenceable Grids (Base)”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/grids	
Target type	Data instance
Name	Referenceable Grids (Base)
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/grids/srs In <code>gml:Grid</code> the attribute group <code>gml:SRSReferenceGroup</code> inherited from <code>gml:AbstractGeometry</code> SHALL not be used. In <code>gml:RectifiedGrid</code> the attribute group <code>gml:SRSReferenceGroup</code> inherited from <code>gml:AbstractGeometry</code> SHALL link the rectified grid to the external coordinate reference system to which it is referenceable (identified in ISO 19123 through the coordinate reference system of the origin). For consistency with ISO 19123, the <code>gml:SRSReferenceGroup</code> of <code>gml:RectifiedGrid</code> MAY be omitted in favour of providing it with the <code>gml:origin</code>. If both are supplied, then they SHALL be identical.
Requirement	http://www.opengis.net/spec/GML/3.3/req/grids/gridcov <code>gml:GridCoverage</code> SHALL not be used.

Table 18 specifies the requirements class “Referenceable Grids”.

Table 18 — Requirements class “Referenceable Grids”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/rgrid	
Target type	Data instance
Name	Referenceable Grids
Dependency	http://www.opengis.net/spec/GML/3.3/req/grids
Requirement	http://www.opengis.net/spec/GML/3.3/req/rgrid/valid Any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/rgrid schema SHALL be well-formed and valid.

10.10 Conformance

Table 19 specifies the conformance class “Referenceable Grids (Base)”.

Table 19 — Conformance class “Referenceable Grids (Base)”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/grids	
Requirements	http://www.opengis.net/spec/GML/3.3/req/grids
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4

Table 19 (continued)

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/grids	
Test	http://www.opengis.net/spec/GML/3.3/conf/grids/srs
	Requirement http://www.opengis.net/spec/GML/3.3/req/grids/srs
	Test purpose Inspect the XML instance to verify that all requirements are met.
	Test method If the XML document contains a <code>gml:Grid</code> element verify that no attribute of the attribute group <code>gml:SRSReferenceGroup</code> including <code>srsName</code> is provided. If the XML document contains a <code>gml:RectifiedGrid</code> element verify that the attribute <code>@srsName</code> or <code>gml:origin/*/srsName</code> is provided. Verify that the value identifies the external coordinate reference system to which it is referenceable. If both are provided, verify that the values are identical.
	Test type Capability
Test	http://www.opengis.net/spec/GML/3.3/conf/grids/gridcov
	Requirement http://www.opengis.net/spec/GML/3.3/req/grids/gridcov
	Test purpose Inspect the XML instance to verify that all requirements are met.
	Test method Verify that the XML document does not contain a <code>gml:GridCoverage</code> element.
	Test type Capability

Table 20 specifies the conformance class “Referenceable Grids”.

Table 20 — Conformance class “Referenceable Grids”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/rgrid	
Requirements	http://www.opengis.net/spec/GML/3.3/req/rgrid
Dependency	http://www.opengis.net/spec/GML/3.3/conf/grids
Test	http://www.opengis.net/spec/GML/3.3/conf/rgrid/valid
	Requirement http://www.opengis.net/spec/GML/3.3/req/rgrid/valid
	Test purpose Verify that any XML node whose content model is specified using schema components in the http://www.opengis.net/gml/3.3/rgrid schema is well-formed and valid.
	Test method Validate the XML document using the XML schema document http://schemas.opengis.net/gml/3.3/referenceableGrids.xsd . Pass if no errors reported. Fail otherwise.
	Test type Basic

11 Code lists, dictionaries and definitions

11.1 Overview

Definition and Dictionary encoding is part of the GML schema as a stop-gap, pending the availability of a suitable general purpose dictionary model. Since the GML Dictionary schema was developed, standards on this topic within the semantic web community have emerged and matured. In particular best-practice is to generally use URIs for referring to items in vocabularies, and RDF (OWL, SKOS) for encoding their descriptions. The GML dictionary schemas for units of measures and reference systems are implementations of specific dictionaries and unaffected by the changes described in [Clause 11](#).

Within GML 3.2 `gml:ReferenceType` and `gml:CodeType` are both used to refer to an external item. Furthermore, the pattern for using `gml:CodeType` to refer to a resource identified by URI is

not standardized: in particular the two-component value `codeSpace/value` is not consistent with contemporary web practice which is to use the URI for referencing. In the context of GML 3.3, the scope of `gml:CodeType` is clarified to be name assignment, not cross-referencing.

Table 21 lists all clarifications on the use of `gml:CodeType` in the context of this part of ISO 19136. In the context of ISO 19136:2007, the GML 3.2 conformance classes, `gml:CodeType` may still be used as specified in ISO 19136:2007.

Table 21 — Clarifications for code list values and definitions

Reference in ISO 19136:2007	Clarification in the context of the conformance class specified in Clause 11
8.2.3.5 <code>CodeType</code> , <code>CodeWithAuthorityType</code>	<code>gml:CodeType</code> is a generalized type to be used for <i>assigning</i> a term, keyword or name. NOTE 1 Elements with type=<code>'gml:CodeType'</code> are used to assign a name to a feature or other resource. <code>gml:ReferenceType</code> is used to hold a reference to another resource. The use of <code>CodeType</code> to reference code list entries is deprecated.
8.2.4.2 <code>CodeListType</code> , <code>CodeOrNilReasonListType</code>	These types are deprecated.
15.1 Overview 15.2.4 Using definitions and dictionaries 21.10 Schemas defining dictionaries and definitions	Certain types of definition are specifically required to support geographic information. GML provides encodings for the following: coordinate operations and dependent objects (GML 3.2, Clause 12), coordinate reference systems and dependent objects (GML 3.2, Clause 12), temporal reference systems (GML 3.2, Clause 14), and units of measure (GML 3.2, Clause 16). The elements <code>gml:Definition</code> and <code>gml:Dictionary</code> provide a common basis for the GML definition types and collections. The GML Schema for definitions and dictionaries was previously used for generic definitions and code lists but is now deprecated for these purposes. NOTE 2 Definitions provided by external authorities may already be packaged for delivery in various ways, both online and offline. In order that they may be referred to from GML documents it is merely necessary that a URI be available to identify for each definition. Applications that do not have existing dictionaries, vocabularies or ontologies may choose any suitable encoding. Semantic web representations based on the RDF model should be considered for new dictionaries, or to encode existing dictionaries in a form compatible with contemporary web technologies.

11.2 Requirements class

Table 22 specifies the requirements class “Code lists, dictionaries and definitions”.

Table 22 — Requirements class “Code lists, dictionaries and definitions”

Requirements Class	
http://www.opengis.net/spec/GML/3.3/req/definitions	
Target type	Data instance
Name	Code lists, dictionaries and definitions
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Requirement	http://www.opengis.net/spec/GML/3.3/req/definitions/codetype Property elements with a content model of <code>gml:CodeType</code> SHALL only occur to assign a name to an object, not for references to another resource. Property elements with a content model of <code>gml:CodeListType</code> or <code>gml:CodeOrNilReasonListType</code> SHALL not occur.
Requirement	http://www.opengis.net/spec/GML/3.3/req/definitions/definition <code>gml:Definition</code> SHALL not occur in the instance document.

11.3 Conformance

Table 23 specifies the conformance class “Code lists, dictionaries and definitions”.

Table 23 — Conformance class “Code lists, dictionaries and definitions”

Conformance Class	
http://www.opengis.net/spec/GML/3.3/conf/definitions	
Requirements	http://www.opengis.net/spec/GML/3.3/req/definitions
Dependency	http://www.opengis.net/doc/IS/GML/3.2/clause/2.4
Test	http://www.opengis.net/spec/GML/3.3/conf/definitions/codetype
	Requirement http://www.opengis.net/spec/GML/3.3/req/definitions/codetype
	Test purpose Inspect the XML instance to verify that all requirements are met.
	Test method Verify using the schemas directly or indirectly referenced from the XML document that no property element of the XML document is of type <code>gml:CodeType</code> unless the documentation of the schema clarifies that this property is used to assign a name to an object, not to reference another resource. Verify using the schemas directly or indirectly referenced from the XML document that no property element of the XML document is of type <code>gml:CodeListType</code> or <code>gml:CodeOrNilReasonListType</code> .
	Test type Capability
Test	http://www.opengis.net/spec/GML/3.3/conf/definitions/definition
	Requirement http://www.opengis.net/spec/GML/3.3/req/definitions/definition
	Test purpose Inspect the XML instance to verify that all requirements are met.
	Test method Verify that the XML document does not contains a <code>gml:Definition</code> element.
	Test type Capability

12 Encoding rule

12.1 Target namespace

All schema components specified in 12.4 are in the target namespace:

<http://www.opengis.net/gml/3.3/exr>

12.2 Improved conversion rule

12.2.1 Conversion rule changes

The conversion rule in ISO 19136:2007, E.2.4, (GML 3.2) is improved as specified in Table 24.

Table 24 — Improvements to the encoding rule for code lists

Reference in ISO 19136:2007	Clarification in the context of the conformance class specified in Clause 12
Table E.1 — Schema encoding overview	A <<CodeList>> is by default converted to an external dictionary, within which each item is identified by a URI. An alternative mapping is a union of an enumeration and a pattern. NOTE This changes the default conversion for code lists.