

# Package ‘geometa’

November 28, 2017

**Type** Package

**Title** Tools for Reading and Writing ISO/OGC Geographic Metadata

**Version** 0.2-0

**Date** 2017-11-27

**Author** Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Maintainer** Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Description** Provides facilities to handle reading and writing of geographic metadata defined with OGC/ISO 19115, 11119 and 19110 geographic information metadata standards, and encoded using the ISO 19139 (XML) standard. It includes also a facility to check the validity of ISO 19139 XML encoded metadata.

**Depends** R (>= 3.1.0)

**Imports** R6, XML

**Suggests** testthat, roxygen2

**License** MIT + file LICENSE

**URL** <https://github.com/eblondel/geometa/wiki>

**BugReports** <https://github.com/eblondel/geometa/issues>

**LazyLoad** yes

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-11-28 10:21:59 UTC

## R topics documented:

fetchISOCodelists . . . . .	5
geometa . . . . .	6
GMLAbstractCoordinateOperation . . . . .	6
GMLAbstractCoordinateSystem . . . . .	7
GMLAbstractCRS . . . . .	8
GMLAbstractGeneralConversion . . . . .	9
GMLAbstractGeneralDerivedCRS . . . . .	10

GMLAbstractGeneralOperationParameter . . . . .	11
GMLAbstractGML . . . . .	12
GMLAbstractObject . . . . .	13
GMLAbstractSingleCRS . . . . .	14
GMLAbstractSingleOperation . . . . .	14
GMLAbstractTimeGeometricPrimitive . . . . .	15
GMLAbstractTimeObject . . . . .	16
GMLAbstractTimePrimitive . . . . .	17
GMLAffineCS . . . . .	18
GMLBaseUnit . . . . .	18
GMLCartesianCS . . . . .	19
GMLConventionalUnit . . . . .	20
GMLConversion . . . . .	21
GMLCoordinateSystemAxis . . . . .	22
GMLCylindricalCS . . . . .	23
GMLDefinition . . . . .	24
GMLDerivedCRS . . . . .	25
GMLDerivedUnit . . . . .	26
GMLElement . . . . .	27
GMLEllipsoidalCS . . . . .	28
GMLGeodeticCRS . . . . .	28
GMLLinearCS . . . . .	29
GMLObliqueCartesianCS . . . . .	30
GMLOperationMethod . . . . .	31
GMLOperationParameter . . . . .	32
GMLOperationParameterGroup . . . . .	33
GMLPolarCS . . . . .	34
GMLProjectedCRS . . . . .	34
GMLSphericalCS . . . . .	35
GMLTemporalCS . . . . .	36
GMLTimeCS . . . . .	37
GMLTimePeriod . . . . .	37
GMLUnitDefinition . . . . .	39
GMLUserDefinedCS . . . . .	40
GMLVerticalCS . . . . .	40
ISOAbsoluteExternalPositionalAccuracy . . . . .	41
ISOAbstractCatalogue . . . . .	42
ISOAbstractGenericName . . . . .	44
ISOAbstractObject . . . . .	45
ISOAbstractPositionalAccuracy . . . . .	46
ISOAddress . . . . .	47
ISOAnchor . . . . .	48
ISOAngle . . . . .	49
ISOAssociationRole . . . . .	50
ISOBand . . . . .	51
ISOBaseBoolean . . . . .	52
ISOBaseCharacterString . . . . .	53
ISOBaseDate . . . . .	54

ISOBaseDateTime	55
ISOBaseDecimal	56
ISOBaseInteger	57
ISOBaseReal	58
ISOBinding	59
ISOBoundAssociationRole	60
ISOBoundFeatureAttribute	61
ISOBrowseGraphic	62
ISOCarrierOfCharacteristics	63
ISOCellGeometry	64
ISOCharacterSet	65
ISOCitation	66
ISOClassification	67
ISOCodelist	68
ISOCodeListValue	69
ISOConformanceResult	70
ISOConstraint	71
ISOConstraints	72
ISOContact	73
ISOContentInformation	74
ISOCoupledResource	75
ISOCouplingType	76
ISOCoverageContentType	77
ISOCoverageDescription	78
ISODataIdentification	79
ISODataQuality	83
ISODataQualityAbstractElement	85
ISODatatype	86
ISODate	87
ISODateType	88
ISODCPList	89
ISODefinitionReference	90
ISODefinitionSource	91
ISODigitalTransferOptions	92
ISODimension	93
ISODimensionNameType	94
ISODistance	95
ISODistribution	96
ISODistributor	97
ISODomainConsistency	98
ISOElementSequence	99
ISOEvaluationMethodType	100
ISOExtendedElementInformation	101
ISOExtent	103
ISOFeatureAssociation	105
ISOFeatureAttribute	106
ISOFeatureCatalogue	107
ISOFeatureCatalogueDescription	110

ISOFeatureOperation . . . . .	111
ISOFeatureType . . . . .	112
ISOFileName . . . . .	114
ISOFormat . . . . .	115
ISOGeographicBoundingBox . . . . .	116
ISOGeographicDescription . . . . .	117
ISOGeographicExtent . . . . .	118
ISOGeometricObjects . . . . .	119
ISOGeometricObjectType . . . . .	120
ISOGeoreferenceable . . . . .	121
ISOGriddedDataPositionalAccuracy . . . . .	122
ISOGridSpatialRepresentation . . . . .	124
ISOHierarchyLevel . . . . .	125
ISOIdentification . . . . .	126
ISOIdentifier . . . . .	128
ISOInheritanceRelation . . . . .	129
ISOKeywords . . . . .	130
ISOKeywordType . . . . .	131
ISOLanguage . . . . .	132
ISOLegalConstraints . . . . .	133
ISOLength . . . . .	134
ISOLineage . . . . .	135
ISOListedValue . . . . .	136
ISOLocalName . . . . .	138
ISOMaintenanceFrequency . . . . .	138
ISOMaintenanceInformation . . . . .	139
ISOMeasure . . . . .	140
ISOMemberName . . . . .	141
ISOMetadata . . . . .	142
ISOMetadataExtensionInformation . . . . .	148
ISOMetadataNamespace . . . . .	150
ISOMetaIdentifier . . . . .	151
ISOMimeFileType . . . . .	152
ISOMultiplicity . . . . .	153
ISOMultiplicityRange . . . . .	154
ISOObligation . . . . .	155
ISOOnlineResource . . . . .	156
ISOOperationMetadata . . . . .	157
ISOParameter . . . . .	158
ISOParameterDirection . . . . .	159
ISOPresentationForm . . . . .	160
ISOProcessStep . . . . .	161
ISOPropertyType . . . . .	162
ISORangeDimension . . . . .	163
ISORecord . . . . .	164
ISORecordType . . . . .	165
ISOReferenceIdentifier . . . . .	166
ISOReferenceSystem . . . . .	167

ISORepresentativeFraction . . . . . 168

ISOResponsibleParty . . . . . 169

ISORestriction . . . . . 170

ISORole . . . . . 171

ISORoleType . . . . . 172

ISOScale . . . . . 173

ISOScope . . . . . 174

ISOScopedName . . . . . 175

ISOSecurityConstraints . . . . . 175

ISOServiceIdentification . . . . . 177

ISOSource . . . . . 182

ISOSpatialRepresentation . . . . . 183

ISOSpatialRepresentationType . . . . . 184

ISOSpatialTemporalExtent . . . . . 185

ISOStandardOrderProcess . . . . . 186

ISOStatus . . . . . 187

ISOTelephone . . . . . 188

ISOTemporalExtent . . . . . 189

ISOTimePeriod . . . . . 190

ISOTopicCategory . . . . . 190

ISOTopologyLevel . . . . . 191

ISOTypeName . . . . . 192

ISOUnlimitedInteger . . . . . 193

ISOURL . . . . . 194

ISOVectorSpatialRepresentation . . . . . 195

ISOVerticalExtent . . . . . 196

**Index** **197**

---

fetchISOCodelists      *fetchISOCodelists*

---

**Description**

fetchISOCodelists

**Usage**

fetchISOCodelists()

---

geometa

*Tools for Reading and Writing ISO/OGC Geographic Metadata*

---

### Description

Provides facilities to handle reading and writing of geographic metadata defined with OGC/ISO 19115 and 19110 geographic information metadata standards, and encoded using the ISO 19139 (XML) standard.

### Details

Package: geometa  
Type: Package  
Version: 0.2-0  
Date: 2017-11-20  
License: MIT  
LazyLoad: yes

### Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

GMLAbstractCoordinateOperation

*GMLAbstractCoordinateOperation*

---

### Description

GMLAbstractCoordinateOperation

### Usage

GMLAbstractCoordinateOperation

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an GMLAbstractCoordinateOperation

**Fields**

operationVersion  
coordinateOperationAccuracy  
sourceCRS  
targetCRS

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS  
setVersion(version) Sets version  
addAccuracy(accuracy) Adds coordinate operation accuracy, object extending ISOAbstractPositionalAccuracy  
delAccuracy(accuracy) Deletes coordinate operation accuracy, object extending ISOAbstractPositionalAccuracy  
setSourceCRS(sourceCRS) Sets the source CRS, object extending GMLAbstractSingleCRS  
setTargetCRS(targetCRS) Sets the target CRS, object extending GMLAbstractSingleCRS

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractCoordinateSystem  
*GMLAbstractCoordinateSystem*

---

**Description**

GMLAbstractCoordinateSystem

**Usage**

GMLAbstractCoordinateSystem

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractCoordinateSystem

**Fields**

axis

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS

addAxis(axis) Adds an axis, object of class GMLCoordinateSystemAxis

delAxis(axis) Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractCRS

*GMLAbstractCRS*

---

**Description**

GMLAbstractCRS

**Usage**

GMLAbstractCRS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractCRS

**Fields**

scope

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS

addScope(scope) Adds a scope

delScope(scope) Deletes a scope



**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractGeneralConversion  
*GMLAbstractGeneralConversion*

---

**Description**

GMLAbstractGeneralConversion

**Usage**

GMLAbstractGeneralConversion

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractGeneralConversion

**Inherited methods**

from GMLAbstractCoordinateOperation

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractGeneralDerivedCRS

*GMLAbstractGeneralDerivedCRS*

---

**Description**

GMLAbstractGeneralDerivedCRS

**Usage**

GMLAbstractGeneralDerivedCRS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractGeneralDerivedCRS

**Fields**

conversion

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML abstract general derived CRS

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractGeneralOperationParameter

*GMLAbstractGeneralOperationParameter*

---

**Description**

GMLAbstractGeneralOperationParameter

**Usage**

GMLAbstractGeneralOperationParameter

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractGeneralOperationParameter

**Fields**

minimumOccurs

**Inherited methods**

from GMLDefinition

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML AbstractGeneralOperationParameter

`setMinimumOccurs(minimumOccurs)` Sets the minimum occurs, object of class integer

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractGML

*GMLAbstractGML*

---

### Description

GMLAbstractGML

### Usage

GMLAbstractGML

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an GML abstract GML

### Fields

descriptionReference

identifier

name

### Methods

`new(xml, element, attrs, defaults)` This method is used to instantiate a GML abstract GML

`setDescriptionReference(descriptionReference)` Set the descriptionReference

`setIdentifier(identifier)` Set the identifier

`addName(name)` Adds a name

`delName(name)` Deletes a name

### Note

Class used internally by geometa

### Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### References

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractObject	<i>GMLAbstractObject</i>
-------------------	--------------------------

---

**Description**

GMLAbstractObject

**Usage**

GMLAbstractObject

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML abstract object

**Methods**

`new(xml, element, attrs, defaults)` This method is used to instantiate an ISOTemporalPrimitive

**Note**

Class used internally by geometa

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractSingleCRS    *GMLAbstractSingleCRS*

---

**Description**

GMLAbstractSingleCRS

**Usage**

GMLAbstractSingleCRS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractSingleCRS

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract single CRS

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractSingleOperation  
*GMLAbstractSingleOperation*

---

**Description**

GMLAbstractSingleOperation

**Usage**

GMLAbstractSingleOperation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLAbstractSingleOperation

**Inherited methods**

from GMLAbstractCoordinateOperation

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAbstractTimeGeometricPrimitive

*GMLAbstractTimeGeometricPrimitive*

---

**Description**

GMLAbstractTimeGeometricPrimitive

**Usage**

GMLAbstractTimeGeometricPrimitive

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GML abstract temporal primitive

**Methods**

`new(xml, element, namespace, defaults)` This method is used to instantiate an ISOAbstract-TimeGeometricPrimitive

**Note**

Class used internally by geometa

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

GMLAbstractTimeObject *GMLAbstractTimeObject*

---

**Description**

GMLAbstractTimeObject

**Usage**

GMLAbstractTimeObject

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML AbstractTimeObject

**Fields**

remarks

**Methods**

`new(xml, defaults)` This method is used to instantiate a GML AbstractTimeObject

`setId(id)` Sets the id

`addRemark(remark)` Adds a remark

`delRemark(remark)` Deletes a remark

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>



---

GMLAbstractTimePrimitive

*GMLAbstractTimePrimitive*

---

### **Description**

GMLAbstractTimePrimitive

### **Usage**

GMLAbstractTimePrimitive

### **Format**

[R6Class](#) object.

### **Value**

Object of [R6Class](#) for modelling an GML AbstractTimePrimitive

### **Methods**

`new(xml, defaults)` This method is used to instantiate a GML AbstractTimePrimitive

`setId(id)` Sets the id

`addRelatedTime(time)` Adds related time, object of class among GMLTimeInstant, GMLTimePeriod, GMLTimeNode or GMLTimeEdge

`delRelatedTime(time)` Deletes related time

### **Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### **References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLAffineCS

*GMLAffineCS*


---

**Description**

GMLAffineCS

**Usage**

GMLAffineCS

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an GMLAffineCS

**Inherited Methods**

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS

addAxis(axis) Adds an axis, object of class GMLCoordinateSystemAxis

delAxis(axis) Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondell@gmail.com&gt;

**References**ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>


---

GMLBaseUnit

*GMLBaseUnit*


---

**Description**

GMLBaseUnit

**Usage**

GMLBaseUnit

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an GML base unit

**Fields**

unitsSystem

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Base Unit

setUnitsSystem(unitsSystem) Set the unit system

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

**Examples**

```
gml <- GMLBaseUnit$new()
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$setUnitsSystem("somelink")
```

---

GMLCartesianCS

*GMLCartesianCS*

---

**Description**

GMLCartesianCS

**Usage**

GMLCartesianCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLCartesianCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLConventionalUnit    *GMLConventionalUnit*

---

**Description**

GMLConventionalUnit

**Usage**

GMLConventionalUnit

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML derived unit

**Fields**

`conversionToPreferredUnit`

`roughConversionToPreferredUnit`

`derivationUnitTerm`

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Base Unit

`addDerivationUnitTerm(uom, exponent)` Adds a derivation unit term, made of a uom reference, and an exponent which can be negative/positive but not equal to zero.

`delDerivationUnitTerm(uom, exponent)` Deletes a derivation unit term

`setConversionToPreferredUnit(uom, factor, rough)` Sets the conversion to preferred unit. rough is FALSE by default

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

**Examples**

```
gml <- GMLConventionalUnit$new()
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$addDerivationUnitTerm("uomId", 2L)
gml$setConversionToPreferredUnit("uomId", 2L)
```

---

GMLConversion

*GMLConversion*

---

**Description**

GMLConversion

**Usage**

GMLConversion

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLConversion

**Fields**

method  
parameterValue

**Inherited methods**

from GMLAbstractCoordinateOperation

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Conversion  
setMethod(method) Sets the method  
addParameterValue(paramValue) Adds a parameter value  
delParameterValue(paramValue) Deletes a parameter value

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLCoordinateSystemAxis  
*GMLCoordinateSystemAxis*

---

**Description**

GMLCoordinateSystemAxis

**Usage**

GMLCoordinateSystemAxis

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLCoordinateSystemAxis

**Fields**

axisAbbrev  
axisDirection  
minimumValue  
maximumValue  
rangeMeaning

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS  
setAbbrev(abbrev) Sets the axis abbreviation  
setDirection(direction, codeSpace) Sets the axis direction  
setMinimumValue(value) Sets the minimum value  
setMaximumValue(value) Sets the maximum value  
setRangeMeaning(meaning, codeSpace) Sets the range meaning

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLCylindricalCS      *GMLCylindricalCS*

---

**Description**

GMLCylindricalCS

**Usage**

GMLCylindricalCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLCylindricalCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class `GMLCoordinateSystemAxis`

`delAxis(axis)` Deletes an axis, object of class `GMLCoordinateSystemAxis`

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLDefinition

*GMLDefinition*

---

**Description**

GMLDefinition

**Usage**

GMLDefinition

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML definition

**Fields**

remarks

**Methods**

`new(xml, defaults)` This method is used to instantiate a GML Definition

`setId(id)` Sets the id

`addRemark(remark)` Adds a remark

`delRemark(remark)` Deletes a remark



**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
 OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

**Examples**

```
gml <- GMLDefinition$new()
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
```

---

 GMLDerivedCRS

*GMLDerivedCRS*


---

**Description**

GMLDerivedCRS

**Usage**

GMLDerivedCRS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLDerivedCRS

**Fields**

baseCRS  
 derivedCRSType  
 coordinateSystem

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML derived CRS  
`setBaseCRS(crs)` Sets the base CRS, one object of class inherited from GMLAbstractSingleCRS  
`setDerivedCRSType(type, codeSpace)` Sets a derived CRS type  
`setCoordinateSystem(cs)` Sets the coordinate system

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLDerivedUnit

*GMLDerivedUnit*

---

**Description**

GMLDerivedUnit

**Usage**

GMLDerivedUnit

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML derived unit

**Fields**

derivationUnitTerm

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Base Unit

`addDerivationUnitTerm(uom, exponent)` Adds a derivation unit term, made of a uom reference, and an exponent which can be negative/positive but not equal to zero.

`delDerivationUnitTerm(uom, exponent)` Deletes a derivation unit term

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

**Examples**

```
gml <- GMLDerivedUnit$new()
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$addDerivationUnitTerm("uomId", 2L)
```

---

GMLElement

*GMLElement*

---

**Description**

GMLElement

**Usage**

GMLElement

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML element

**Methods**

`new(xml, element, attrs, defaults)` This method is used to instantiate a GML element

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

GMLEllipsoidalCS      *GMLEllipsoidalCS*

---

**Description**

GMLEllipsoidalCS

**Usage**

GMLEllipsoidalCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLEllipsoidalCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLGeodeticCRS      *GMLGeodeticCRS*

---

**Description**

GMLGeodeticCRS

**Usage**

GMLGeodeticCRS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLGeodeticCRS

**Fields**

ellipsoidalCS

cartesianCS

sphericalCS

geodeticDatum

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract single CRS

`setEllipsoidalCS(cs)` Sets an ellipsoidal CS

`setCartesianCS(cs)` Sets a cartesian CS

`setSphericalCS(cs)` Sets a spherical CS

`setGeodeticDatum(datum)` Sets geodetic datum

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLLinearCS

*GMLLinearCS*

---

**Description**

GMLLinearCS

**Usage**

GMLLinearCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLLinearCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLObliqueCartesianCS *GMLObliqueCartesianCS*

---

**Description**

GMLObliqueCartesianCS

**Usage**

GMLObliqueCartesianCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLObliqueCartesianCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

- ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)
- OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLOperationMethod      *GMLOperationMethod*

---

**Description**

GMLOperationMethod

**Usage**

GMLOperationMethod

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLOperationMethod

**Fields**

formulaCitation  
sourceDimensions  
targetDimensions

**Inherited methods**

from GMLDefinition

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML OperationMethod

`setFormulaCitation(citation)` Sets the formula citation, object of class ISOCitation

`setSourceDimensions(value)` Sets the number of source dimensions, object of class integer

`setTargetDimensions(value)` Sets the number of target dimensions, object of class integer

`addParameter(parameter)` Adds a parameter or parameter group, object of class GMLOperationParameter or GMLOperationParameterGroup

`delParameter(parameter)` Deletes a parameter or parameter group, object of class GMLOperationParameter or GMLOperationParameterGroup

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLOperationParameter *GMLOperationParameter*

---

**Description**

GMLOperationParameter

**Usage**

GMLOperationParameter

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLOperationParameter

**Inherited methods**

GMLAbstractGeneralOperationParameter

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML OperationParameter

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>



---

GMLOperationParameterGroup  
*GMLOperationParameterGroup*

---

**Description**

GMLOperationParameterGroup

**Usage**

GMLOperationParameterGroup

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLOperationParameterGroup

**Fields**

maximumOccurs  
parameter

**Inherited methods**

from GMLAbstractGeneralOperationParameter

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML OperationParameterGroup  
`setMaximumOccurs(maximumOccurs)` Sets the maximum occurs, object of class integer  
`addParameter(parameter)` Adds a parameter or parameter group, object of class GMLOperationParameter  
or GMLOperationParameterGroup  
`delParameter(parameter)` Deletes a parameter or parameter group, object of class GMLOperationParameter  
or GMLOperationParameterGroup

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLPolarCS

*GMLPolarCS*

---

**Description**

GMLPolarCS

**Usage**

GMLPolarCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLPolarCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLProjectedCRS

*GMLProjectedCRS*

---

**Description**

GMLProjectedCRS

**Usage**

GMLProjectedCRS

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an GMLProjectedCRS

**Fields**

baseGeodeticCRS

cartesianCS

**Methods**

new(xml, defaults, id) This method is used to instantiate a GML projected CRS

setBaseGeodeticCRS(crs) Sets the base geodetic CRS, object of class GMLBaseGeodeticCRS

setCartesianCS(cs) Sets the cartesianCS, object of class GMLCartesianCS

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLSphericalCS

*GMLSphericalCS*

---

**Description**

GMLSphericalCS

**Usage**

GMLSphericalCS

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an GMLSphericalCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS  
`addAxis(axis)` Adds an axis, object of class `GMLCoordinateSystemAxis`  
`delAxis(axis)` Deletes an axis, object of class `GMLCoordinateSystemAxis`

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLTemporalCS

*GMLTemporalCS*

---

**Description**

GMLTemporalCS

**Usage**

GMLTemporalCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLTemporalCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS  
`addAxis(axis)` Adds an axis, object of class `GMLCoordinateSystemAxis`  
`delAxis(axis)` Deletes an axis, object of class `GMLCoordinateSystemAxis`

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLTimeCS

*GMLTimeCS*

---

**Description**

GMLTimeCS

**Usage**

GMLTimeCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLTimeCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLTimePeriod

*GMLTimePeriod*

---

**Description**

GMLTimePeriod

**Usage**

GMLTimePeriod

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an GMLTimePeriod

**Fields**

beginPosition

endPosition

duration

**Methods**

new(xml, beginPosition, endPosition) This method is used to instantiate an GMLTimePeriod

setBeginPosition(beginPosition) Sets the begin position (beginning date or date and time of the resource contents), as object of class "POSIXct"/"POSIXt" or "Date"

setEndPosition(endPosition) Sets the end position (ending date or date and time of the resource contents), as object of class "POSIXct"/"POSIXt" or "Date"

computeInterval() Computes the ISO interval string and set as GML id

setId(id) Sets the GML id string.

setDuration(years, months, days, hours, mins, secs) Set duration (Length of time between measurements)

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**Examples**

```
start <- ISOdate(2000, 1, 12, 12, 59, 45)
end <- ISOdate(2010, 8, 22, 13, 12, 43)
md <- GMLTimePeriod$new(beginPosition = start, endPosition = end)
xml <- md$encode()
```

---

GMLUnitDefinition      *GMLUnitDefinition*

---

**Description**

GMLUnitDefinition

**Usage**

GMLUnitDefinition

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GML unit definition

**Fields**

quantityTypeReference  
catalogSymbol

**Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Definition  
`setQuantityTypeReference(ref)` Set the quantity type reference. The content is a reference to a remote value  
`setCatalogSymbol(symbol)` Sets the preferred lexical symbol used for this unit of measure

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)  
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

**Examples**

```
gml <- GMLUnitDefinition$new()
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
```

---

GMLUserDefinedCS      *GMLUserDefinedCS*

---

**Description**

GMLUserDefinedCS

**Usage**

GMLUserDefinedCS

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLUserDefinedCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

GMLVerticalCS      *GMLVerticalCS*

---

**Description**

GMLVerticalCS

**Usage**

GMLVerticalCS



**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLVerticalCS

**Inherited Methods**

`new(xml, defaults, id)` This method is used to instantiate a GML Abstract CRS

`addAxis(axis)` Adds an axis, object of class GMLCoordinateSystemAxis

`delAxis(axis)` Deletes an axis, object of class GMLCoordinateSystemAxis

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19136:2007 Geographic Information – Geographic Markup Language. [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_t](http://www.iso.org/iso/iso_catalogue/catalogue_t)

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

---

ISOAbsoluteExternalPositionalAccuracy

*ISOAbsoluteExternalPositionalAccuracy*

---

**Description**

ISOAbsoluteExternalPositionalAccuracy

**Usage**

ISOAbsoluteExternalPositionalAccuracy

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOAbsoluteExternalPositionalAccuracy

**Inherited methods**

from ISODataQualityAbstractElement

**Methods**

`new(xml)` This method is used to instantiate an ISOAbsoluteExternalPositionalAccuracy

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#encoding
dq <- ISOAbsoluteExternalPositionalAccuracy$new()
dq$addNameOfMeasure("measure")
metaId <- ISOMetaIdentifier$new(code = "measure-id")
dq$setMeasureIdentification(metaId)
dq$setMeasureDescription("description")
dq$setEvaluationMethodDescription("method description")
dq$setEvaluationMethodType("indirect")
dq$setDateTime(ISOdate(2015,1,1,12,10,49))
spec <- ISOCitation$new()
spec$setTitle("specification title")
spec$setAlternateTitle("specification alternate title")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
spec$addDate(d)
dq$setEvaluationProcedure(spec)
result <- ISOConformanceResult$new()
result$setSpecification(spec)
result$setExplanation("some explanation about the conformance")
result$setPass(TRUE)
dq$addResult(result)
xml <- dq$encode()
```

---

ISOAbstractCatalogue    *ISOAbstractCatalogue*

---

**Description**

ISOAbstractCatalogue

**Usage**

ISOAbstractCatalogue

### **Format**

R6Class object.

### **Value**

Object of R6Class for modelling an ISOAbstracCatalogue

### **Fields**

name

scope

fieldOfApplication

versionNumber

versionDate

### **Methods**

new(xml) This method is used to instantiate an ISOAbstractCatalogue

setName(name) Sets the name

addScope(scope) Adds scope (object of class character)

delScope(scope) Deletes scope

addFieldOfApplication(fieldOfApplication) Adds a field of application (object of class character)

delFieldOfApplication(fieldOfApplication) Deletes fieldOfApplication

setVersionNumber(versionNumber) Sets version number (object of class character)

setVersionDate(versionDate) Sets version date

### **Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

### **References**

ISO 19139:2007 Metadata - XML schema implementation

---

ISOAbstractGenericName

*ISOAbstractGenericName*

---

**Description**

ISOAbstractGenericName

**Usage**

ISOAbstractGenericName

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO abstract GenericName

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOLocalName

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOAbstractObject	<i>ISOAbstractObject</i>
-------------------	--------------------------

---

**Description**

ISOAbstractObject

**Usage**

ISOAbstractObject

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Metadata Element

**Static Methods**

`getISOClassByNode(node)` Inherit the ISO class matching an XML document or node

`compare(metadataElement1, metadataElement2)` Compares two metadata elements objects. Returns TRUE if they are equal, FALSE otherwise. The comparison of object is done by comparing the XML representation of the objects (since no R6 object comparison method seems to exist)

**Abstract Methods**

`new(xml, element, namespace, defaults, attrs)` This method is used to instantiate an ISOAbstractObject

`INFO(text)` Logger to report information. Used internally

`WARN(text)` Logger to report warnings. Used internally

`ERROR(text)` Logger to report errors. Used internally

`print()` Provides a custom print output (as tree) of the current class

`decode(xml)` Decodes a ISOMetadata\* R6 object from XML representation

`encode(addNS, validate, strict)` Encodes a ISOMetadata\* R6 object to XML representation. By default, namespace definition will be added to XML root (`addNS = TRUE`), and validation of object will be performed (`validate = TRUE`) prior to its XML encoding. The argument `strict` allows to stop the encoding in case object is not valid, with a default value set to FALSE.

`validate(xml, strict)` Validates the encoded XML against ISO 19139 XML schemas. If `strict` is TRUE, a error will be raised. Default is FALSE.

`getNamespaceDefinition(recursive)` Gets the namespace definition of the current ISO\* class. By default, only the namespace definition of the current element is retrieved (`recursive = FALSE`).

`getClassName()` Gets the class name

`getClass()` Gets the class

`wrapBaseElement(field, fieldObj)` Wraps a base element type

`contains(field, metadataElement)` Indicates if the present class object contains a metadata element object for a particular list-based field.

`addListElement(field, metadataElement)` Adds a metadata element to a list-based field. Returns TRUE if the element has been added, FALSE otherwise. In case an element is already added, the element will not be added and this method will return FALSE.

`delListElement(field, metadataElement)` Deletes a metadata element from a list-based field. Returns TRUE if the element has been deleted, FALSE otherwise. In case an element is absent, this method will return FALSE.

`setAttr(attrKey, attrValue)` Set an attribute

`setId(id, addNS)` Set an id. By default addNS is FALSE (no namespace prefix added).

`setHref(href)` Sets an href reference

`setCodeList(codeList)` Sets a codeList

`setCodeListValue(codeListValue)` Sets a codeList value

`setCodeSpace(codeSpace)` Set a codeSpace

`setValue(value)` Set a value

`isDocument()` Indicates if the object is a metadata document, typically an object of class `ISOMetadata` or `ISOFeatureCatalogue`

`isFieldInheritedFrom(field)` Gives the parent from which the field is inherited, otherwise return NULL.

**Note**

Abstract ISO Metadata class used internally by `geometa`

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

---

ISOAbstractPositionalAccuracy

*ISOAbstractPositionalAccuracy*

---

**Description**

ISOAbstractPositionalAccuracy

**Usage**

ISOAbstractPositionalAccuracy

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOAbstractPositionalAccuracy

**Inherited methods**

from ISODataQualityAbstractElement

**Methods**

`new(xml)` This method is used to instantiate an ISOAbstractPositionalAccuracy

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOAddress

*ISOAddress*

---

**Description**

ISOAddress

**Usage**

ISOAddress

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Address

**Fields**

deliveryPoint

city

postalCode

country

electronicMailAddress

**Methods**

`new(xml)` This method is used to instantiate an ISOAddress  
`setDeliveryPoint(deliveryPoint)` Sets the delivery point  
`setCity(city)` Sets the city  
`setPostalCode(postalCode)` Sets the postal code  
`setCountry(country)` Sets the country  
`setEmail(email)` Sets the electronic Mail address

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOAddress$new()
md$setDeliveryPoint("theaddress")
md$setCity("thecity")
md$setPostalCode("111")
md$setCountry("France")
md$setEmail("someone@theorg.org")
xml <- md$encode()
```

---

ISOAnchor

*ISOAnchor*

---

**Description**

ISOAnchor

**Usage**

ISOAnchor

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Anchor



**Methods**

`new(xml, name, ...)` This method is used to instantiate an ISOAnchor

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19139:2007 Geographic information – XML

**Examples**

```
md <- ISOAnchor$new(name = "some entity name", href = "someentityuri")
xml <- md$encode()
```

---

ISOAngle

*ISOAngle*

---

**Description**

ISOAngle

**Usage**

ISOAngle

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOAngle measure

**Fields**

value

**Methods**

`new(xml, value, uom, useUomURI)` This method is used to instantiate an ISOAngle. The `uom` argument represents the symbol of unit of measure used. The parameter `useUomURI` can be used to set the `uom` as URI, its default value is FALSE.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOAssociationRole      *ISOAssociationRole*

---

**Description**

ISOAssociationRole

**Usage**

ISOAssociationRole

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOAssociationRole

**Fields**

type  
isOrdered  
isNavigable  
relation  
rolePlayer

**Methods**

**new(xml)** This method is used to instantiate an ISOAssociationRole  
**setRoleType(roleType)** Sets the role type, object of class ISORoleType or any character value among ISORoleType\$values().  
**setIsOrdered(isOrdered)** Sets TRUE if ordered, FALSE otherwise  
**setIsNavigable(isNavigable)** Sets TRUE if navigable, FALSE otherwise  
**codesetRelation(relation)** Sets an object of class ISOFeatureAssociation as relation  
**addRolePlayer(rolePlayer)** Adds a role player, object of class ISOFeatureType  
**delRolePlayer(rolePlayer)** Deletes a role player, object of class ISOFeatureType

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISOBand

*ISOBand*

---

**Description**

ISOBand

**Usage**

ISOBand

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOBand

**Fields**

maxValue  
minValue  
units  
peakResponse  
bitsPerValue  
toneGradation  
scaleFactor  
offset

**Methods**

`new(xml)` This method is used to instantiate an ISOBand  
`setMaxValue(maxValue)` Sets the maximum value  
`setMinValue(minValue)` Sets the minimum value  
`setUnits(units)` Sets the unit, object of class `GMLUnitDefinition`  
`setPeakResponse(peakResponse)` Sets the peak response  
`setBitsPerValue(bitsPerValue)` Sets the bits per value  
`setToneGradation` Sets the tone gradation  
`setScaleFactor(scaleFactor)` Sets the scale factor  
`setOffset(offset)` Sets the offset

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Examples**

```
#create band range dimension
md <- ISOBand$new()
md$setSequenceIdentifier(ISOMemberName$new(aName = "name", attributeType = "type"))
md$setDescriptor("descriptor")
md$setMaxValue(10)
md$setMinValue(1)
gml <- GMLBaseUnit$new(id = "ID")
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$setUnitsSystem("somelink")
md$setUnits(gml)
md$setPeakResponse(9)
md$setBitsPerValue(5)
md$setToneGradation(100)
md$setScaleFactor(1)
md$setOffset(4)
xml <- md$encode()
```

---

ISOBaseBoolean

*ISOBaseBoolean*


---

**Description**

ISOBaseBoolean

**Usage**

ISOBaseBoolean

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Boolean

**Fields**

value

**Methods**

`new(xml,value)` This method is used to instantiate an ISOBaseBoolean

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBaseCharacterString

*ISOBaseCharacterString*

---

**Description**

ISOBaseCharacterString

**Usage**

ISOBaseCharacterString

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO BaseCharacterString

**Fields**

value

**Methods**

`new(xml,value)` This method is used to instantiate an ISOBaseCharacterString

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBaseDate

*ISOBaseDate*

---

**Description**

ISOBaseDate

**Usage**

ISOBaseDate

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Date

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOBaseDate

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBaseDateTime	<i>ISOBaseDateTime</i>
-----------------	------------------------

---

**Description**

ISOBaseDateTime

**Usage**

ISOBaseDateTime

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DateTime

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOBaseDateTime

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBaseDecimal	<i>ISOBaseDecimal</i>
----------------	-----------------------

---

**Description**

ISOBaseDecimal

**Usage**

ISOBaseDecimal

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Decimal

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOBaseDecimal

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language



---

ISOBaseInteger	<i>ISOBaseInteger</i>
----------------	-----------------------

---

**Description**

ISOBaseInteger

**Usage**

ISOBaseInteger

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Integer

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOBaseInteger

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBaseReal

*ISOBaseReal*

---

**Description**

ISOBaseReal

**Usage**

ISOBaseReal

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Real

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOBaseReal

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOBinding

*ISOBinding*

---

**Description**

ISOBinding

**Usage**

ISOBinding

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOBinding

**Fields**

description

globalProperty

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOBinding

`setDescription(description)` Set description of inheritance relation

`setPropertyType(propertyType)` Set global property, object of class ISOPropertyType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISOBoundAssociationRole

*ISOBoundAssociationRole*

---

**Description**

ISOBoundAssociationRole

**Usage**

ISOBoundAssociationRole

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOBoundAssociationRole

**Fields**

rolePlayer

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOBoundAssociationRole

`setFeatureType(featureType)` Set feature type, object of class ISOFeatureType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISOBoundFeatureAttribute

*ISOBoundFeatureAttribute*

---

**Description**

ISOBoundFeatureAttribute

**Usage**

ISOBoundFeatureAttribute

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOBoundFeatureAttribute

**Fields**

valueType

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOBoundFeatureAttribute

`setTypeNames(typeNames)` Set typeNames, object of class ISOTypeNames or character

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

ISOBrowseGraphic      *ISOBrowseGraphic*

---

**Description**

ISOBrowseGraphic

**Usage**

ISOBrowseGraphic

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO BrowseGraphic

**Fields**

fileName

fileDescription

fileType

**Methods**

`new(xml, fileName, fileDescription, fileType)` This method is used to instantiate an ISO-BrowseGraphic

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOBrowseGraphic$new(  
  fileName = "http://www.somefile.org/png",  
  fileDescription = "Map Overview",  
  fileType = "image/png"  
)  
xml <- md$encode()
```

---

ISOCarrierOfCharacteristics  
*ISOCarrierOfCharacteristics*

---

**Description**

ISOCarrierOfCharacteristics

**Usage**

ISOCarrierOfCharacteristics

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOCarrierOfCharacteristics

**Fields**

featureType  
constrainedBy

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOCarrierOfCharacteristics  
`setFeatureType(featureType)` Set a feature type, object of class ISOFeatureType  
`addConstraint(constraint)` Add constraint, object of class ISOConstraint  
`delConstraint(constraint)` Deletes constraint, object of class ISOConstraint

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

ISOCellGeometry

*ISOCellGeometry*

---

**Description**

ISOCellGeometry

**Usage**

ISOCellGeometry

**Format**[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an ISO CellGeometryCode**Fields**

value

**Methods**`new(xml,value, description)` This method is used to instantiate an ISOCellGeometry**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondell@gmail.com&gt;

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOCellGeometry$values(labels = TRUE)

#example of 'point' cell geometry code
pointCode <- ISOCellGeometry$new(value = "point")
```



---

ISOCharacterSet	<i>ISOCharacterSet</i>
-----------------	------------------------

---

**Description**

ISOCharacterSet

**Usage**

ISOCharacterSet

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO CharacterSet

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOCharacterSet

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOCharacterSet$values(labels = TRUE)

#some charset
charset <- ISOCharacterSet$new(value = "utf8")
```

---

ISOCitation

*ISOCitation*

---

### Description

ISOCitation

### Usage

ISOCitation

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO Citation

### Fields

title  
alternateTitle  
date  
edition  
editionDate  
identifier  
presentationForm

### Methods

`new(xml)` This method is used to instantiate an ISOCitation  
`setTitle(title)` Sets the title  
`setAlternateTitle(alternateTitle)` Sets an alternate title  
`addDate(date)` Adds the date (ISODate object containing date and dateType)  
`setEdition(edition)` Sets the edition  
`setEditionDate(editionDate)` Sets the edition date, either an ISODate object containing date and dateType or a simple R date "POSIXct"/"POSIXt" object. For thesaurus citations, an ISODate should be used while for the general citation of ISODataIdentification, a simple R date should be used.  
`setIdentifier(identifier)` Sets the identifier as object of class 'ISOMetaIdentifier'  
`setCitedResponsibleParty(rp)` Sets the cited responsible party  
`setPresentationForm` Sets the presentation form

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create ISOCitation
md <- ISOCitation$new()
md$setTitle("sometitle")
md$setEdition("1.0")
md$setEditionDate(ISOdate(2015,1,1))
md$setIdentifier(ISOMetaIdentifier$new(code = "identifier"))
md$setPresentationForm("mapDigital")

#add a cited responsible party
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
md$setCitedResponsibleParty(rp)
xml <- md$encode()
```

---

ISOClassification

*ISOClassification*

---

**Description**

ISOClassification

**Usage**

ISOClassification

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO Classification

**Fields**

value

**Methods**

new(xml,value, description) This method is used to instantiate an ISOClassification

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOClassification$values(labels = TRUE)

#restricted classification
cl <- ISOClassification$new(value = "restricted")
```

---

ISOCodelist

*ISOCodelist*

---

**Description**

ISOCodelist

**Usage**

ISOCodelist

**Format**

R6Class object.

**Value**

Object of [R6Class](#) for modelling an ISO codelist

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOCodelist

**Note**

Class used by geometa internal codelist XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

---

ISOCodeListValue	<i>ISOCodeListValue</i>
------------------	-------------------------

---

**Description**

ISOCodeListValue

**Usage**

ISOCodeListValue

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Metadata codelist element

**Fields**

value

**Methods**

`new(xml, id, value, description, addCodeListAttrs, addCodeSpaceAttr, setValue)` This method is used to instantiate an ISOCodeListValue. By default, `addCodeListAttrs = TRUE`, to add codelist attributes to root XML. The parameter `addCodeSpaceAttr = TRUE` by default, and ignored if the value of `addCodeListAttrs` is set to `FALSE`. The argument `setValue` sets the value as node text (default is `TRUE`).

`getAcceptedValues()` This method allows to get the codelist accepted values

**Note**

Abstract ISO codelist class used internally by geometa

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOConformanceResult *ISOConformanceResult*

---

**Description**

ISOConformanceResult

**Usage**

ISOConformanceResult

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ConformanceResult

**Fields**

result

**Methods**

`new(xml, value)` This method is used to instantiate an ISOConformanceResult  
`setSpecification(specification)` Sets the specification (an ISOCitation object)  
`setExplanation(explanation)` Sets the explanation  
`setPass(pass)` Sets if passing the conformance or not (logical value)

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOConformanceResult$new()
spec <- ISOCitation$new()
spec$setTitle("specification title")
spec$setAlternateTitle("specification alternate title")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
spec$addDate(d)
md$setSpecification(spec)
md$setExplanation("some explanation about the conformance")
md$setPass(TRUE)
xml <- md$encode()
```

---

ISOConstraint

*ISOConstraint*

---

**Description**

ISOConstraint

**Usage**

ISOConstraint

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOConstraint

**Fields**

signature

formalDefinition

**Methods**

`new(xml, description)` This method is used to instantiate an ISOConstraint

`setDescription(description)` Sets the description

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
md <- ISOConstraint$new(description = "description")
xml <- md$encode()
```

---

ISOConstraints

*ISOConstraints*

---

**Description**

ISOConstraints

**Usage**

ISOConstraints

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO abstract Constraints

**Fields**

useLimitation

**Methods**

`new(xml)` This method is used to instantiate an ISOLegalConstraints  
`addUseLimitation(useLimitation)` Adds a use limitation  
`setUseLimitation(useLimitation)` Sets a use limitation  
`delUseLimitation(useLimitation)` Deletes a use limitation

**Note**

Abstract ISO class

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata



---

ISOContact

*ISOContact*

---

**Description**

ISOContact

**Usage**

ISOContact

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Contact

**Fields**

phone

address

onlineResource

**Methods**

`new(xml)` This method is used to instantiate an ISOContact

`setPhone(phone)` Sets the phone contact

`setAddress(address)` Sets the address contact

`setOnlineResource(onlineResource)` Sets the online resource

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
md$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
md$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
md$setOnlineResource(res)
xml <- md$encode()
```

---

ISOContentInformation *ISOContentInformation*

---

**Description**

ISOContentInformation

**Usage**

ISOContentInformation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOContentInformation

**Abstract Methods**

`new(xml)` This method is used to instantiate an ISOContentInformation

**Note**

Abstract class. Used internally by **geometa**

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOCoupledResource     *ISOCoupledResource*

---

**Description**

ISOCoupledResource

**Usage**

ISOCoupledResource

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOCoupledResource

**Fields**

operationName  
identifier

**Methods**

`new(xml)` This method is used to instantiate an ISOCoupledResource  
`setOperationName(operationName)` Set the operation name  
`setIdentifier(identifier)` Set the identifier

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Services

**Examples**

```
md <- ISOCoupledResource$new()  
md$setOperationName("name")  
md$setIdentifier("identifier")  
xml <- md$encode()
```

---

ISOCouplingType	<i>ISOCouplingType</i>
-----------------	------------------------

---

**Description**

ISOCouplingType

**Usage**

ISOCouplingType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOCouplingType

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOCouplingType

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Services

**Examples**

```
#possible values
values <- ISOCouplingType$values(labels = TRUE)

#couplingType
couplingType <- ISOCouplingType$new(value = "loose")
```

---

ISOCoverageContentType

*ISOCoverageContentType*

---

## Description

ISOCoverageContentType

## Usage

ISOCoverageContentType

## Format

[R6Class](#) object.

## Value

Object of [R6Class](#) for modelling an ISO CoverageContentType

## Fields

value

## Methods

`new(xml,value, description)` This method is used to instantiate an ISOCoverageContentType

## Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

## References

ISO 19115:2003 - Geographic information – Metadata

## Examples

```
#possible values
values <- ISOCoverageContentType$values(labels = TRUE)

#example of CoverageContentType
modelResultType <- ISOCoverageContentType$new(value = "modelResult")
```

---

ISOCoverageDescription

*ISOCoverageDescription*

---

**Description**

ISOCoverageDescription

**Usage**

ISOCoverageDescription

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOCoverageDescription

**Fields**

attributeDescription

contentType

dimension

**Methods**

`new(xml)` This method is used to instantiate an ISOCoverageDescription

`setAttributeDescription` Sets the attribute description, as object of class ISORecordType or character

`setContentType(contentType)` Sets the content Type, as object of class ISOCoverageContentType or any character value listed in ISOCoverageContentType\$values()

`addDimension(dimension)` Adds a dimension, object of class (or subclass of) ISORangeDimension

`delDimension(dimension)` Deletes a dimension, object of class (or subclass of) ISORangeDimension

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create coverage description
md <- ISOCoverageDescription$new()
md$setAttributeDescription("test")
md$setContentTypes("modelResult")

#adding 3 arbitrary dimensions
for(i in 1:3){
  band <- ISOBand$new()
  mn <- ISOMemberName$new(aName = sprintf("name %s",i), attributeType = sprintf("type %s",i))
  band$setSequenceIdentifier(mn)
  band$setDescriptor("descriptor")
  band$setMaxValue(10)
  band$setMinValue(1)
  gml <- GMLBaseUnit$new(id = sprintf("ID%s",i))
  gml$setDescriptionReference("someref")
  gml$setIdentifier("identifier", "codespace")
  gml$addName("name1", "codespace")
  gml$addName("name2", "codespace")
  gml$setQuantityTypeReference("someref")
  gml$setCatalogSymbol("symbol")
  gml$setUnitsSystem("somelink")
  band$setUnits(gml)
  band$setPeakResponse(9)
  band$setBitsPerValue(5)
  band$setToneGradation(100)
  band$setScaleFactor(1)
  band$setOffset(4)
  md$addDimension(band)
}
xml <- md$encode()
```

---

ISODataIdentification *ISODataIdentification*

---

**Description**

ISODataIdentification

**Usage**

ISODataIdentification

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DataIdentification

**Fields**

spatialRepresentationType  
 spatialResolution  
 language  
 characterSet  
 topicCategory  
 environmentDescription  
 extent  
 supplementalInformation

**Inherited methods**

setCitation(citation) Sets an object of class ISOCitation  
 setAbstract(abstract) Sets an abstract (object of class "character")  
 setPurpose(purpose) Sets a purpose (object of class "character")  
 addCredit(credit) Adds a credit (object of class "character")  
 delCredit(credit) Deletes a credit (object of class "character")  
 addStatus(status) Adds a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().  
 delStatus(status) Deletes a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().  
 addPointOfContact(pointOfContact) Adds an object of class ISOResponsibleParty  
 delPointOfContact(pointOfContact) Deletes an object of class ISOResponsibleParty  
 addResourceMaintenance(resourceMaintenance) Adds a resource maintenance information as object of class ISOMaintenanceInformation.  
 setResourceMaintenance(resourceMaintenance) Sets a resource maintenance information as object of class ISOMaintenanceInformation.  
 delResourceMaintenance(resourceMaintenance) Deletes a resource maintenance information as object of class ISOMaintenanceInformation.  
 addGraphicOverview(graphicOverview) Adds an object of class ISOBrowseGraphic  
 setGraphicOverview(graphicOverview) Sets an object of class ISOBrowseGraphic  
 delGraphicOverview(graphicOverview) Deletes an object of class ISOBrowseGraphic  
 addKeywords(keywords) Adds a set of keywords as object of class ISOKeywords  
 setKeywords(keywords) Sets a set of keywords as object of class ISOKeywords  
 delKeywords(keywords) Deletes a set of keywords as object of class ISOKeywords  
 addResourceConstraints(resourceConstraints) Adds an object of class ISOLegalConstraints  
 setResourceConstraints(resourceConstraints) Sets an object of class ISOLegalConstraints  
 delResourceConstraints(resourceConstraints) Deletes an object of class ISOLegalConstraints



**Methods**

`new(xml, value)` This method is used to instantiate an `ISODataIdentification`

`addSpatialRepresentationType(spatialRepresentationType)` Adds a spatial representation type, as object of class "character" or class `ISOSpatialRepresentationType`. If an object of class "character" is specified, it must match the accepted values listed by `ISOSpatialRepresentationType$values()`.

`setSpatialRepresentationType(spatialRepresentationType)` Sets a spatial representation type, as object of class "character" or class `ISOSpatialRepresentationType`. If an object of class "character" is specified, it must match the accepted values listed by `ISOSpatialRepresentationType$values()`.

`delSpatialRepresentationType(spatialRepresentationType)` Deletes a spatial representation type, as object of class "character" or class `ISOSpatialRepresentationType`. If an object of class "character" is specified, it must match the accepted values listed by `ISOSpatialRepresentationType$values()`.

`addLanguage(locale)` Adds a language, as object of class "character" or class `ISOLanguage`. If an object of class "character" is specified, it must match the accepted language values `ISOLanguage$values()`.

`setLanguage(locale)` Sets a language, as object of class "character" or class `ISOLanguage`. If an object of class "character" is specified, it must match the accepted language values `ISOLanguage$values()`.

`delLanguage(locale)` Deletes a language, as object of class "character" or class `ISOLanguage`. If an object of class "character" is specified, it must match the accepted language values `ISOLanguage$values()`.

`addCharacterSet(charset)` Adds a character set, as object of class "character" or class `ISOCharacterSet`. If an object of class "character" is specified, it must match the accepted charset values `ISOCharacterSet$values()`.

`setCharacterSet(charset)` Sets a character set, as object of class "character" or class `ISOCharacterSet`. If an object of class "character" is specified, it must match the accepted charset values `ISOCharacterSet$values()`.

`delCharacterSet(charset)` Deletes a character set, as object of class "character" or class `ISOCharacterSet`. If an object of class "character" is specified, it must match the accepted charset values `ISOCharacterSet$values()`.

`addTopicCategory(topicCategory)` Adds a character set, as object of class "character" or class `ISOTopicCategory`. If an object of class "character" is specified, it must match the accepted topic category values `ISOTopicCategory$values()`.

`setTopicCategory(topicCategory)` Sets a character set, as object of class "character" or class `ISOTopicCategory`. If an object of class "character" is specified, it must match the accepted topic category values `ISOTopicCategory$values()`.

`delTopicCategory(topicCategory)` Deletes a character set, as object of class "character" or class `ISOTopicCategory`. If an object of class "character" is specified, it must match the accepted topic category values `ISOTopicCategory$values()`.

`setEnvironmentDescription(environmentDescription)` Sets the environment description

`addExtent(extent)` Adds an object of class `ISOExtent`.

`setExtent(extent)` Sets an object of class `ISOExtent`.

`delExtent(extent)` Deletes an object of class `ISOExtent`.

`setSupplementalInformation(supplementalInformation)` Sets supplemental information

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

## References

ISO 19115:2003 - Geographic information – Metadata

## Examples

```
#create dataIdentification
md <- ISODataIdentification$new()
md$setAbstract("abstract")
md$setPurpose("purpose")
md$setLanguage("eng")
md$setCharacterSet("utf8")
md$addTopicCategory("biota")
md$addTopicCategory("oceans")

#adding a point of contact
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
md$addPointOfContact(rp)

#citation
ct <- ISOCitation$new()
ct$setTitle("sometitle")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
ct$addDate(d)
ct$setEdition("1.0")
ct$setEditionDate(ISOdate(2015, 1, 1, 1))
ct$setIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$setPresentationForm("mapDigital")
ct$setCitedResponsibleParty(rp)
md$setCitation(ct)
```

```
#graphic overview
go <- ISOBrowseGraphic$new(
  fileName = "http://www.somefile.org/png",
  fileDescription = "Map Overview",
  fileType = "image/png"
)
md$setGraphicOverview(go)

#maintenance information
mi <- ISOMaintenanceInformation$new()
mi$setMaintenanceFrequency("daily")
md$setResourceMaintenance(mi)

#adding legal constraints
lc <- ISOLegalConstraints$new()
lc$addUseLimitation("limitation1")
lc$addUseLimitation("limitation2")
lc$addUseLimitation("limitation3")
lc$addAccessConstraint("copyright")
lc$addAccessConstraint("license")
lc$addUseConstraint("copyright")
lc$addUseConstraint("license")
md$setResourceConstraints(lc)

#adding extent
extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
md$setExtent(extent)

#add keywords
kwds <- ISOKeywords$new()
kwds$addKeyword("keyword1")
kwds$addKeyword("keyword2")
kwds$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle("General")
th$addDate(d)
kwds$setThesaurusName(th)
md$addKeywords(kwds)

#supplementalInformation
md$setSupplementalInformation("some additional information")

xml <- md$encode()
```

**Description**

ISODataQuality

**Usage**

ISODataQuality

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DataQuality

**Fields**

scope  
report  
lineage

**Methods**

`new(xml)` This method is used to instantiate an ISODataQuality  
`setScope(scope)` Sets the scope  
`addReport(report)` Adds a report  
`setLineage(lineage)` Sets the lineage

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create dataQuality object
dq <- ISODataQuality$new()

#add scope
scope <- ISOScope$new()
scope$setLevel("dataset")
dq$setScope(scope)

#add report
dc <- ISODomainConsistency$new()
result <- ISOConformanceResult$new()
```

```
spec <- ISOCitation$new()
spec$setTitle("specification title")
spec$setAlternateTitle("specification alternate title")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
spec$addDate(d)
result$setSpecification(spec)
result$setExplanation("some explanation about the conformance")
result$setPass(TRUE)
dc$addResult(result)
dq$addReport(dc)

#add lineage
lineage <- ISOLineage$new()
lineage$setStatement("statement")
dq$setLineage(lineage)

#xml
xml <- dq$encode()
```

---

ISODataQualityAbstractElement

*ISODataQualityAbstractElement*

---

## Description

ISODataQualityAbstractElement

## Usage

ISODataQualityAbstractElement

## Format

[R6Class](#) object.

## Value

Object of [R6Class](#) for modelling an ISODataQualityAbstractElement

## Fields

nameOfMeasure  
measureIdentification  
measureDescription  
evaluationMethodType  
evaluationMethodDescription

evaluationProcedure  
 dateTime  
 result

### Methods

new(xml) This method is used to instantiate an ISODataQuality  
 addNameOfMeasure(name) Add name  
 delNameOfMeasure(name) Deletes name  
 setMeasureIdentification(identification) Sets measure identifier, an object of class ISOMetaIdentifier  
 setMeasureDescription(description) Sets measure description  
 setEvaluationMethodType(type) Sets the type of evaluation method, an object of class ISOEvaluationMethodType  
 or any character value among codeISOEvaluationMethodType\$values()  
 setEvaluationMethodDescription(description) Sets the description of evaluation method  
 setEvaluationMethodProcedure(procedure) Sets the procedure as an object of class ISOCitation  
 setDateTime(dateTime) Sets datetime, object class 'POSIXct'/'POSIXt'  
 addResult(result) Sets conformance result, object of class ISOConformanceResult  
 delResult(result) Deletes conformance result, object of class ISOConformanceResult

### Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### References

ISO 19115:2003 - Geographic information – Metadata

---

ISODatatype

*ISODatatype*

---

### Description

ISODatatype

### Usage

ISODatatype

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO Datatype

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISODatatype

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISODatatype$values(labels = TRUE)

#string Datatype
stringType <- ISODatatype$new(value = "characterString")
```

---

ISODate

*ISODate*

---

**Description**

ISODate

**Usage**

ISODate

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Date

**Fields**

date

dateType

**Methods**

`new(xml)` This method is used to instantiate an ISODate  
`setDate(date)` Sets the date  
`setDateType(dateType)` Sets the date type

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISODate$new()
d <- ISOBaseDate$new(value = ISOdate(2015, 1, 1, 1))
md$setDate(d)
md$setDateType("publication")
xml <- md$encode()
```

---

ISODateType

*ISODateType*


---

**Description**

ISODateType

**Usage**

ISODateType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DateType

**Fields**

value

**Methods**

`new(xml, value, description)` This method is used to instantiate an ISODateType



**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISODateType$values(labels = TRUE)

#creation datatype
creation <- ISODateType$new(value = "creation")
```

---

ISODCPList

*ISODCPList*

---

**Description**

ISODCPList

**Usage**

ISODCPList

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DCPList

**Fields**

value

**Methods**

`new(xml, value, description)` This method is used to instantiate an ISODCPList

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Service

**Examples**

```
#possible values
values <- ISODCPList$values(labels = TRUE)

#example
javaDCP <- ISODCPList$new(value = "JAVA")
```

---

ISODefinitionReference

*ISODefinitionReference*

---

**Description**

ISODefinitionReference

**Usage**

ISODefinitionReference

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISODefinitionReference

**Fields**

sourceIdentifier  
definitionSource

**Methods**

`new(xml)` This method is used to instantiate an ISODefinitionReference  
`setSourceIdentifier(identifier)` Sets the source identifier as object of class character  
`setDefinitionSource(source)` Sets the definition source as object of class ISODefinitionSource  
or directly using a ISOCitation

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISODefinitionSource    *ISODefinitionSource*

---

**Description**

ISODefinitionSource

**Usage**

ISODefinitionSource

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISODefinitionSource

**Fields**

source

**Methods**

`new(xml, source)` This method is used to instantiate an ISODefinitionSource

`setSource(source)` Sets the source as object of class ISOCitation

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISODigitalTransferOptions

*ISODigitalTransferOptions*

---

### **Description**

ISODigitalTransferOptions

### **Usage**

ISODigitalTransferOptions

### **Format**

[R6Class](#) object.

### **Value**

Object of [R6Class](#) for modelling an ISO DigitalTransferOptions

### **Fields**

onLine

### **Methods**

`new(xml, value)` This method is used to instantiate an ISODigitalTransferOptions

`setUnitsOfDistribution(unit)` Sets the units of distribution

`setTransferSize(transferSize)` Sets the transfer Size

`addOnlineResource(onlineResource)` Adds an object of class ISOnlineResource

`setOnlineResource(onlineResource)` Sets an object of class ISOnlineResource

`delOnlineResource(onlineResource)` Deletes an object of class ISOnlineResource

### **Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### **References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISODigitalTransferOptions$new()

or <- ISOOnlineResource$new()
or$setLinkage("http://somelink")
or$setName("name")
or$setDescription("description")
or$setProtocol("WWW:LINK-1.0-http--link")
md$addOnlineResource(or)

xml <- md$encode()
```

---

ISODimension

*ISODimension*


---

**Description**

ISODimension

**Usage**

ISODimension

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Dimension

**Fields**

dimensionName  
dimensionSize  
resolution

**Methods**

`new(xml, value)` This method is used to instantiate an ISODimension  
`setName(name)` Sets the dimension name. Object of class ISODimensionNameType or any value from ISODimensionNameType\$values()  
`setSize(size)` Sets the dimension size, object of class integer  
`setResolution(resolution)` Sets the resolution ie. object of class ISOMeasure or any subclass ISOLength, ISODistance, ISOAngle, ISOScale

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create dimension
md <- ISODimension$new()
md$setName("row")
md$setSize(1)
md$setResolution(ISOLength$new(value=1,uom="m"))
xml <- md$encode()
```

---

ISODimensionNameType    *ISODimensionNameType*

---

**Description**

ISODimensionNameType

**Usage**

ISODimensionNameType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO DimensionNameType

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISODimensionNameType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISODimensionNameType$values(labels = TRUE)

#row DimensionNameType
rowType <- ISODimensionNameType$new(value = "row")
```

---

ISODistance

*ISODistance*

---

**Description**

ISODistance

**Usage**

ISODistance

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Distance measure

**Fields**

value

**Methods**

`new(xml,value, uom, useUomURI)` This method is used to instantiate an ISODistance. The `uom` argument represents the symbol of unit of measure used. The parameter `useUomURI` can be used to set the `uom` as URI, its default value is FALSE.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISODistribution	<i>ISODistribution</i>
-----------------	------------------------

---

**Description**

ISODistribution

**Usage**

ISODistribution

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Distribution

**Fields**

transferOptions

**Methods**

`new(xml)` This method is used to instantiate an ISODistribution

`addFormat(format)` Adds a distribution format, object of class ISOFormat

`delFormat(format)` Deletes a distribution format, object of class ISOFormat

`addDistributor(distributor)` Adds a distributor, object of class ISODistributor

`delDistributor(distributor)` Deletes a distributor, object of class ISODistributor

`setDigitalTransferOptions(options)` Sets the digital transfer options

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata



**Examples**

```
md <- ISODistribution$new()

dto <- ISODigitalTransferOptions$new()
for(i in 1:3){
  or <- ISOOnlineResource$new()
  or$setLinkage(paste0("http://somelink",i))
  or$setName(paste0("name",i))
  or$setDescription(paste0("description",i))
  or$setProtocol("WWW:LINK-1.0-http--link")
  dto$addOnlineResource(or)
}
md$setDigitalTransferOptions(dto)

xml <- md$encode()
```

---

ISODistributor

*ISODistributor*

---

**Description**

ISODistributor

**Usage**

ISODistributor

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISODistributor

**Fields**

distributorContact

distributorFormat

**Methods**

`new(xml)` This method is used to instantiate an ISODistributor  
`setContact(contact)` Sets the contact ISOResponsibleParty  
`addFormat(format)` Adds a distributor format ISOFormat  
`delFormat(format)` Deletes a distributor format ISOFormat

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISODistributor$new()
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("Data manager")

contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
rp$setRole("author")
md$setContact(rp)

format <- ISOFormat$new()
format$setName("name")
format$setVersion("1.0")
format$setAmendmentNumber("2")
format$setSpecification("specification")
md$addFormat(format)

xml <- md$encode()
```

---

ISODomainConsistency    *ISODomainConsistency*

---

**Description**

ISODomainConsistency

**Usage**

ISODomainConsistency

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO DomainConsistency

**Fields**

result

**Methods**

new(xml) This method is used to instantiate an ISODomainConsistency  
addResult(result) Adds a result

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOElementSequence      *ISOElementSequence*

---

**Description**

ISOElementSequence

**Usage**

ISOElementSequence

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISOElementSequence

**Fields**

aName  
attributeType

**Methods**

new(xml, ...) This method is used to instantiate an element sequence

**Note**

This class is used internally by geometa to deal with simple type not handled by proper class element. e.g. name property of ISOParameter class from ISO 19119:2005

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOEvaluationMethodType

*ISOEvaluationMethodType*

---

**Description**

ISOEvaluationMethodType

**Usage**

ISOEvaluationMethodType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO EvaluationMethodType

**Fields**

value

**Methods**

new(xml, value, description) This method is used to instantiate an ISOEvaluationMethodType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOEvaluationMethodType$values(labels = TRUE)

#example of EvaluationMethodType
indirect <- ISOEvaluationMethodType$new(value = "indirect")
```

---

ISOExtendedElementInformation  
*ISOExtendedElementInformation*

---

**Description**

ISOExtendedElementInformation

**Usage**

ISOExtendedElementInformation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ExtendedElementInformation

**Fields**

name  
shortName  
domainCode  
definition  
obligation  
condition  
dataType  
maximumOccurrence

domainValue  
parentEntity  
rule  
rationale  
source

### Methods

**new(xml)** This method is used to instantiate an ISOExtendedElementInformation

**setName(name)** Sets the element name, object of class Character

**setShortName(shortName)** Sets the element shortname, object of class character

**setDomainCode(domainCode)** Sets the element domain code, object of class integer

**setDefinition(definition)** Sets the element definition, object of class character

**setObligation(obligation)** Sets an obligation, as object of class character or class ISOobligation.  
If an object of class "character" is specified, it must match the accepted obligation values  
ISOobligation\$values().

**setCondition(condition)** Sets the element condition, object of class character

**setDatatype(dataType)** Sets the element datatype, as object of class character or class ISODatatype.  
If an object of class "character" is specified, it must match the accepted datatype values  
ISODatatype\$values().

**setMaximumOccurrence(maximumOccurrence)** Sets the element maximum occurrence, object  
of class character

**setDomainValue(domainValue)** Sets the element domain value, object of class character

**addParentEntity(parentEntity)** Adds a parent Entity, object of class character

**delParentEntity(parentEntity)** Deletes a parent Entity, object of class character

**setRule(rule)** Sets a rule, object of class character

**addRationale(rationale)** Adds a rationale, object of class character

**delRationale(rationale)** Deletes a rationale, object of class character

**addSource(source)** Adds a source, object of class ISOResponsibleParty

**delSource(source)** Deletes a source, object of class ISOResponsibleParty

### Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### References

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```

md <- ISOExtendedElementInformation$new()
md$setName("name")
md$setShortName("shortName")
md$setDomainCode("1L")
md$setDefinition("some definition")
md$setObligation("mandatory")
md$setCondition("no condition")
md$setDatatype("characterString")
md$setMaximumOccurrence("string")
md$setDomainValue("value")
md$addParentEntity("none")
md$setRule("rule")
md$addRationale("rationale")

#adding a source
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)

md$addSource(rp)

xml <- md$encode()

```

---

ISOExtent

*ISOExtent*


---

**Description**

ISOExtent

**Usage**

ISOExtent

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Extent

**Fields**

geographicElement

temporalElement

verticalElement

**Methods**

`new(xml)` This method is used to instantiate an ISOExtent

`addGeographicElement(extent)` Adds an object extending ISOGeographicExtent

`setGeographicElement(extent)` Sets an object extending ISOGeographicExtent

`delGeographicElement(extent)` Deletes an object extending ISOGeographicExtent

`addTemporalElement(extent)` Adds an object extending ISOTemporalExtent

`setTemporalElement(extent)` Sets an object extending ISOTemporalExtent

`delTemporalElement(extent)` Deletes an object extending ISOTemporalExtent

`addVerticalElement(extent)` Adds an object extending ISOVerticalExtent

`setVerticalElement(extent)` Sets an object extending ISOVerticalExtent

`delVerticalElement(extent)` Deletes an object extending ISOVerticalExtent

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata



---

ISOFeatureAssociation *ISOFeatureAssociation*

---

**Description**

ISOFeatureAssociation

**Usage**

ISOFeatureAssociation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOFeatureAssociation

**Fields**

roleName

**Methods**

`new(xml)` This method is used to instantiate an ISOFeatureAssociation

`addRoleName(associationRole)` Adds an association role, object of class ISOAssociationRole

`delRoleName(associationRole)` Deletes an association role, object of class ISOAssociationRole

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISOFeatureAttribute    *ISOFeatureAttribute*

---

**Description**

ISOFeatureAttribute

**Usage**

ISOFeatureAttribute

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOFeatureAttribute

**Fields**

code  
valueMeasurementUnit  
valueType  
listedValue

**Methods**

`new(xml)` This method is used to instantiate an ISOFeatureAttribute  
`setCode(code)` Sets the code  
`setValueMeasurementUnit(uom)` Sets the value measurement unit, an object of class GMLUnitDefinition  
`setValueType(typeName)` Sets the value type  
`addListedValue(value)` Adds a listed value (object of class ISOListedValue)  
`delListedValue(value)` Deletes a listed value (object of class ISOListedValue)

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
md <- ISOFeatureAttribute$new()
md$setMemberName("name")
md$setDefinition("definition")
md$setCardinality(lower=1,upper=1)
md$setCode("code")

gml <- GMLBaseUnit$new(id = "ID")
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$setUnitsSystem("somelink")
md$setValueMeasurementUnit(gml)

val1 <- ISOListedValue$new()
val1$setCode("code1")
val1$setLabel("label1")
val1$setDefinition("definition1")
md$addListedValue(val1)
val2 <- ISOListedValue$new()
val2$setCode("code2")
val2$setLabel("label2")
val2$setDefinition("definition2")
md$addListedValue(val2)
md$setValueType("typeName")
```

---

ISOFeatureCatalogue    *ISOFeatureCatalogue*

---

**Description**

ISOFeatureCatalogue

**Usage**

ISOFeatureCatalogue

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO FeatureCatalogue

**Fields**

name

**Inherited Methods from ISOAbstractCatalogue**

setName(name) Sets the name  
 addScope(scope) Adds scope (object of class character)  
 delScope(scope) Deletes scope  
 addFieldOfApplication(fieldOfApplication) Adds a field of application (object of class character)  
 delFieldOfApplication(fieldOfApplication) Deletes fieldOfApplication  
 setVersionNumber(versionNumber) Sets version number (object of class character)  
 setVersionDate(versionDate) Sets version date

**Methods**

new(xml, uuid) This method is used to instantiate an ISOFeatureCatalogue  
 setProducer(producer) Sets an object of class ISOResponsibleParty as producer  
 setFunctionalLanguage(functionalLanguage) Sets the functional language  
 addFeatureType(featureType) Adds a feature type, object of class ISOFeatureType  
 delFeatureType(featureType) Deletes a feature type, object of class ISOFeatureType  
 addDefinitionSource(source) Adds a definition source, object of class ISODefinitionSource  
 or ISOCitation  
 delDefinitionSource(source) Deletes a definition source, object of class ISODefinitionSource  
 or ISOCitation

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
fc <- ISOFeatureCatalogue$new(uuid = "my-fc-identifier")
fc$setName("name")
fc$addScope("scope1")
fc$addScope("scope2")
fc$addFieldOfApplication("field1")
fc$addFieldOfApplication("field2")
fc$setVersionNumber("1.0")
fc$setVersionDate(ISOdate(2015, 1, 1, 1))

producer <- ISOResponsibleParty$new()
producer$setIndividualName("someone")
```

```

fc$setProducer(producer)
fc$setFunctionalLanguage("eng")

cit <- ISOCitation$new()
cit$setTitle("some citation title")
fc$addDefinitionSource(cit)
#' #add featureType
ft <- ISOFeatureType$new()
ft$setType("typeName")
ft$setDefinition("definition")
ft$setCode("code")
ft$setIsAbstract(FALSE)
ft$addAlias("alias1")
ft$addAlias("alias2")

#add feature attributes
for(i in 1:3){
  #create attribute
  fat <- ISOFeatureAttribute$new()
  fat$setMemberName(sprintf("name %s",i))
  fat$setDefinition(sprintf("definition %s",i))
  fat$setCardinality(lower=1,upper=1)
  fat$setCode(sprintf("code %s",i))

  gml <- GMLBaseUnit$new(id = sprintf("ID%s",i))
  gml$setDescriptionReference("someref")
  gml$setIdentifier("identifier", "codespace")
  gml$addName("name1", "codespace")
  gml$addName("name2", "codespace")
  gml$setQuantityTypeReference("someref")
  gml$setCatalogSymbol("symbol")
  gml$setUnitsSystem("somelink")
  fat$setValueMeasurementUnit(gml)

  #add listed values
  val1 <- ISOListedValue$new()
  val1$setCode("code1")
  val1$setLabel("label1")
  val1$setDefinition("definition1")
  fat$addListedValue(val1)
  val2 <- ISOListedValue$new()
  val2$setCode("code2")
  val2$setLabel("label2")
  val2$setDefinition("definition2")
  fat$addListedValue(val2)
  fat$setValueType("typeName")

  #add feature attribute as carrierOfCharacteristic
  ft$addCharacteristic(fat)
}
#add featureType to catalogue
fc$addFeatureType(ft)

```

```
xml <- fc$encode()
```

---

ISOFeatureCatalogueDescription  
*ISOFeatureCatalogueDescription*

---

**Description**

ISOFeatureCatalogueDescription

**Usage**

ISOFeatureCatalogueDescription

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOFeatureCatalogue

**Fields**

complianceCode  
language  
includedWithDataset  
featureCatalogueCitation

**Methods**

`new(xml)` This method is used to instantiate an ISOFeatureCatalogueDescription  
`setComplianceCode(compliance)` Sets the compliance. TRUE if compliant, FALSE otherwise  
`addLanguage(lang)` Adds a language  
`delLanguage(lang)` Deletes a language  
`setIncludedWithDataset(include)` Sets TRUE if included with dataset, FALSE otherwise  
`addFeatureCatalogueCitation(citation)` Adds an object of class ISOCitation referencing the link to Feature Catalogue  
`delFeatureCatalogueCitation(citation)` Deletes an object of class ISOCitation referencing the link to Feature Catalogue

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

## References

ISO 19115:2003 - Geographic information – Metadata

## Examples

```
md <- ISOFeatureCatalogueDescription$new()
md$setComplianceCode(FALSE)
md$addLanguage("eng")
md$setIncludedWithDataset(FALSE)

cit = ISOCitation$new()
contact = ISOContact$new()
fcLink <- ISOOnlineResource$new()
fcLink$setLinkage("http://somelink/featurecatalogue")
contact$setOnlineResource(fcLink)
rp = ISOResponsibleParty$new()
rp$setContactInfo(contact)
cit$setCitedResponsibleParty(rp)
md$addFeatureCatalogueCitation(cit)
```

---

ISOFeatureOperation    *ISOFeatureOperation*

---

## Description

ISOFeatureOperation

## Usage

ISOFeatureOperation

## Format

[R6Class](#) object.

## Value

Object of [R6Class](#) for modelling an ISOFeatureOperation

## Fields

signature  
formalDefinition

## Methods

`new(xml)` This method is used to instantiate an ISOFeatureOperation  
`setSignature(signature)` Sets the signature  
`setFormalDefinition(formalDefinition)` Sets the formal definition

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
md <- ISOFeatureOperation$new()
md$setMemberName("name")
md$setDefinition("definition")
md$setCardinality(lower=1,upper=1)
md$setSignature("signature")
md$setFormalDefinition("def")
```

---

ISOFeatureType

*ISOFeatureType*

---

**Description**

ISOFeatureType

**Usage**

ISOFeatureType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO FeatureType

**Fields**

typeName  
definition  
code  
isAbstract  
aliases  
inheritsFrom  
inheritsTo  
featureCatalogue  
constrainedBy  
definitionReference  
carrierOfCharacteristics



**Methods**

new(xml) This method is used to instantiate an ISOFeatureType  
 setTypeNames(typeNames) Sets the type name. Object of class ISOLocalName or "character"  
 setDefinition(definition) Sets the definition  
 setCode(code) Sets the code  
 setIsAbstract(isAbstract) Sets TRUE/FALSE if the feature type is abstract or not  
 addAlias(alias) Set alias name. Object of class ISOLocalName or "character"  
 delAlias(alias) Deletes alias name  
 setFeatureCatalogue(fc) Sets a feature catalogue, object of class ISOFeatureCatalogue  
 addConstraint(constraint) Adds a constraint, object of class ISOConstraint or character  
 delConstraint(constraint) Deletes a constraint, object of class ISOConstraint or character  
 setDefinitionReference(definitionReference) Sets the definition Reference, object of class  
 ISODefinitionReference  
 addCharacteristic(characteristic) Adds a characteristic as object of class ISOPropertyType  
 or subclass  
 delCharacteristic(characteristic) Deletes a characteristic as object of class ISOPropertyType  
 or subclass

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```

#featuretype
md <- ISOFeatureType$new()
md$setTypeNames("typeName")
md$setDefinition("definition")
md$setCode("code")
md$setIsAbstract(FALSE)
md$addAlias("alias1")
md$addAlias("alias2")

#add feature attributes
for(i in 1:3){
  #create attribute
  fat <- ISOFeatureAttribute$new()
  fat$setMemberName(sprintf("name %s",i))
  fat$setDefinition(sprintf("definition %s",i))
  fat$setCardinality(lower=1,upper=1)
  fat$setCode(sprintf("code %s",i))
}

```

```

#add measurement unit
gml <- GMLBaseUnit$new(id = "ID%")
gml$setDescriptionReference("someref")
gml$setIdentifier("identifier", "codespace")
gml$addName("name1", "codespace")
gml$addName("name2", "codespace")
gml$setQuantityTypeReference("someref")
gml$setCatalogSymbol("symbol")
gml$setUnitsSystem("somelink")
fat$setValueMeasurementUnit(gml)

#add listed values
val1 <- ISOListedValue$new()
val1$setCode("code1")
val1$setLabel("label1")
val1$setDefinition("definition1")
fat$addListedValue(val1)
val2 <- ISOListedValue$new()
val2$setCode("code2")
val2$setLabel("label2")
val2$setDefinition("definition2")
fat$addListedValue(val2)
fat$setValueType("typeName")

#add feature attribute as carrierOfCharacteristic
md$addCharacteristic(fat)
}
xml <- md$encode()

```

---

ISOFileName

*ISOFileName*


---

### Description

ISOFileName

### Usage

ISOFileName

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO FileName

### Methods

`new(xml, file, name)` This method is used to instantiate an ISOFileName

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19139:2007 Geographic information – XML

**Examples**

```
md <- ISOFileName$new(file = "someuri", name = "filename")
xml <- md$encode()
```

---

ISOFormat

*ISOFormat*

---

**Description**

ISOFormat

**Usage**

ISOFormat

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOFormat

**Fields**

name  
version  
amendmentNumber  
specification  
fileDecompressionTechnique

**Methods**

`new(xml)` This method is used to instantiate an ISOFormat  
`setName(name)` Sets the format name  
`setVersion(version)` Sets the format version  
`setAmendmentNumber(amendmentNumber)` Sets an admenment number  
`setSpecification(specification)` Sets the format specification  
`setFileDecompressionTechnique(technique)` Sets the file decompression technique  
`addDistributor(distributor)` Adds a distributor, object of class ISODistributor  
`delDistributor(distributor)` Deletes a distributor, object of class ISODistributor

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOFormat$new()
md$setName("name")
md$setVersion("1.0")
md$setAmendmentNumber("2")
md$setSpecification("specification")
```

---

ISOGeographicBoundingBox  
*ISOGeographicBoundingBox*

---

**Description**

ISOGeographicBoundingBox

**Usage**

ISOGeographicBoundingBox

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GeographicBoundingBox

**Fields**

westBoundLongitude  
eastBoundLongitude  
southBoundLatitude  
northBoundLatitude

**Methods**

`new(xml, minx, miny, maxx, maxy, bbox)` This method is used to instantiate an ISOGeographicBoundingBox

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
xml <- md$encode()
```

---

ISOGeographicDescription

*ISOGeographicDescription*

---

**Description**

ISOGeographicDescription

**Usage**

ISOGeographicDescription

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GeographicDescription

**Fields**

geographicIdentifier

**Methods**

`new(xml)` This method is used to instantiate an ISOGeographicDescription

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOGeographicDescription$new()
md$setGeographicIdentifier(ISOMetaIdentifier$new(code = "identifiant"))
xml <- md$encode()
```

---

ISOGeographicExtent    *ISOGeographicExtent*

---

**Description**

ISOGeographicExtent

**Usage**

ISOGeographicExtent

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO abstract geographicExtent

**Fields**

extentTypeCode

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOGeographicExtent

**Note**

abstract class

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOGeometricObjects    *ISOGeometricObjects*

---

**Description**

ISOGeometricObjects

**Usage**

ISOGeometricObjects

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GeometricObjects

**Fields**

geometricObjectType  
geometricObjectCount

**Methods**

`new(xml)` This method is used to instantiate an ISOGeometricObjects  
`setGeometricObjectType(geometricObjectType)` Sets the type of geometric object  
`setGeometricObjectCount(geometricObjectCount)` Sets the count of geometric objects

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOGeometricObjects$new()
md$setGeometricObjectType("surface")
md$setGeometricObjectCount(5L)
xml <- md$encode()
```

---

ISOGeometricObjectType

*ISOGeometricObjectType*

---

**Description**

ISOGeometricObjectType

**Usage**

ISOGeometricObjectType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GeometricObjectType

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOGeometricObjectType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOGeometricObjectType$values(labels = TRUE)

#point type
pt <- ISOGeometricObjectType$new(value = "point")
```



---

ISOGeoreferenceable *ISOGeoreferenceable*

---

**Description**

ISOGeoreferenceable

**Usage**

ISOGeoreferenceable

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Georeferenceable

**Fields**

controlPointAvailability  
orientationParameterAvailability  
orientationParameterDescription  
georeferencedParameters  
parameterCitation

**Inherited Methods**

setNumberOfDimensions Sets the number of dimensions (value of class integer)  
addDimension(dimension) Adds a dimension. Object of class ISODimension  
delDimension(dimension) Deletes a dimension;  
setCellGeometry(cellGeometry) Sets the cell geometry. Object of class ISOCellGeometry or any value from ISOCellGeometry\$values()  
setTransformationParameterAvailability(availability) Sets the transformation parameter availability

**Methods**

new(xml) This method is used to instantiate an ISOGeoreferenceable  
setControlPointAvailability(availability) Sets the control point availability. TRUE/FALSE  
setOrientationParameterAvailability(availability) Sets the orientation parameter availability. TRUE/FALSE  
setOrientationParameterDescription(description) Sets the orientation parameter description

setGeoreferencedParameters(record) Sets the georeferenced parameter (object of class ISORecord)  
 addParameterCitation(citation) Adds a parameter citation  
 delParameterCitation(citation) Deletes a parameter citation

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOGeoreferenceable$new()

#inherited methods from ISOGridSpatialRepresentation
md$setNumberOfDimensions(1)
dim1 <- ISODimension$new()
dim1$setName("row")
dim1$setSize(100)
dim1$setResolution(ISOMeasure$new(value=1,uom="m"))
md$addDimension(dim1)
md$setCellGeometry("area")

#parameters
md$setControlPointAvailability(TRUE)
md$setOrientationParameterAvailability(TRUE)
md$setOrientationParameterDescription("description")
md$setGeoreferencedParameters("record")
ct <- ISOCitation$new()
ct$setTitle("citation")
md$addParameterCitation(ct)

xml <- md$encode()
```

---

ISOGriddedDataPositionalAccuracy

*ISOGriddedDataPositionalAccuracy*

---

**Description**

ISOGriddedDataPositionalAccuracy

**Usage**

ISOGriddedDataPositionalAccuracy

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISOGriddedDataPositionalAccuracy

**Inherited methods**

from ISODataQualityAbstractElement

**Methods**

`new(xml)` This method is used to instantiate an ISOGriddedDataPositionalAccuracy

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#encoding
dq <- ISOGriddedDataPositionalAccuracy$new()
dq$addNameOfMeasure("measure")
metaId <- ISOMetaIdentifier$new(code = "measure-id")
dq$setMeasureIdentification(metaId)
dq$setMeasureDescription("description")
dq$setEvaluationMethodDescription("method description")
dq$setEvaluationMethodType("indirect")
dq$setDateTime(ISOdate(2015,1,1,12,10,49))
spec <- ISOCitation$new()
spec$setTitle("specification title")
spec$setAlternateTitle("specification alternate title")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
spec$addDate(d)
dq$setEvaluationProcedure(spec)
result <- ISOConformanceResult$new()
result$setSpecification(spec)
result$setExplanation("some explanation about the conformance")
result$setPass(TRUE)
dq$addResult(result)
xml <- dq$encode()
```

---

ISOGridSpatialRepresentation  
*ISOGridSpatialRepresentation*

---

**Description**

ISOGridSpatialRepresentation

**Usage**

ISOGridSpatialRepresentation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO GridSpatialRepresentation

**Fields**

numberOfDimensions  
axisDimensionProperties  
cellGeometry  
transformationParameterAvailability

**Methods**

`new(xml)` This method is used to instantiate an ISOGridSpatialRepresentation  
`setNumberOfDimensions` Sets the number of dimensions (value of class integer)  
`addDimension(dimension)` Adds a dimension. Object of class ISODimension  
`delDimension(dimension)` Deletes a dimension;  
`setCellGeometry(cellGeometry)` Sets the cell geometry. Object of class ISOCellGeometry or any value from ISOCellGeometry\$values()  
`setTransformationParameterAvailability(availability)` Sets the transformation parameter availability

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOGridSpatialRepresentation$new()
md$setNumberOfDimensions(1)
dim1 <- ISODimension$new()
dim1$setName("row")
dim1$setSize(100)
dim1$setResolution(ISOMeasure$new(value=1,uom="m"))
md$addDimension(dim1)
md$setCellGeometry("area")
xml <- md$encode()
```

---

ISOHierarchyLevel      *ISOHierarchyLevel*

---

**Description**

ISOHierarchyLevel

**Usage**

ISOHierarchyLevel

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO HierarchyLevel

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOHierarchyLevel

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOHierarchyLevel$values(labels = TRUE)

#dataset scope
ds <- ISOHierarchyLevel$new(value = "dataset")
```

---

ISOIdentification      *ISOIdentification*

---

**Description**

ISOIdentification

**Usage**

ISOIdentification

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Identification

**Fields**

citation  
abstract  
purpose  
credit  
status  
pointOfContact  
resourceMaintenance  
graphicOverview  
descriptiveKeywords  
resourceConstraints  
resourceFormat

**Methods**

`new(xml, element, namespace, defaults)` This method is used to instantiate an ISOIdentification

`setCitation(citation)` Sets an object of class ISOCitation

`setAbstract(abstract)` Sets an abstract (object of class "character")

`setPurpose(purpose)` Sets a purpose (object of class "character")

`addCredit(credit)` Adds a credit (object of class "character")

`delCredit(credit)` Deletes a credit (object of class "character")

`addStatus(status)` Adds a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().

`delStatus(status)` Deletes a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().

`addPointOfContact(pointOfContact)` Adds an object of class ISOResponsibleParty

`delPointOfContact(pointOfContact)` Deletes an object of class ISOResponsibleParty

`addResourceMaintenance(resourceMaintenance)` Adds a resource maintenance information as object of class ISOMaintenanceInformation.

`setResourceMaintenance(resourceMaintenance)` Sets a resource maintenance information as object of class ISOMaintenanceInformation.

`delResourceMaintenance(resourceMaintenance)` Deletes a resource maintenance information as object of class ISOMaintenanceInformation.

`addGraphicOverview(graphicOverview)` Adds an object of class ISOBrowseGraphic

`setGraphicOverview(graphicOverview)` Sets an object of class ISOBrowseGraphic

`delGraphicOverview(graphicOverview)` Deletes an object of class ISOBrowseGraphic

`addFormat(format)` Adds a resource format, object of class ISOFormat

`delFormat(format)` Deletes a resource format, object of class ISOFormat

`addKeywords(keywords)` Adds a set of keywords as object of class ISOKeywords

`setKeywords(keywords)` Sets a set of keywords as object of class ISOKeywords

`delKeywords(keywords)` Deletes a set of keywords as object of class ISOKeywords

`addResourceConstraints(resourceConstraints)` Adds an object extending ISOConstraints, either an object of class ISOLegalConstraints or ISOSecurityConstraints

`setResourceConstraints(resourceConstraints)` Sets an object extending ISOConstraints, either an object of class ISOLegalConstraints or ISOSecurityConstraints

`delResourceConstraints(resourceConstraints)` Deletes an object extending ISOConstraints, either an object of class ISOLegalConstraints or ISOSecurityConstraints

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOIdentifier

*ISOIdentifier*

---

**Description**

ISOIdentifier

**Usage**

ISOIdentifier

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Identifier

**Fields**

authority

code

**Methods**

`new(xml, code, codeSpace)` This method is used to instantiate an ISOIdentifier

`setAuthority(authority)` Sets an authority object of class ISOCitation

**Note**

Abstract ISO class

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata



---

ISOInheritanceRelation

*ISOInheritanceRelation*

---

**Description**

ISOInheritanceRelation

**Usage**

ISOInheritanceRelation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOInheritanceRelation

**Fields**

name  
description  
uniqueInstance  
subtype  
supertype

**Methods**

`new(xml, defaults)` This method is used to instantiate an ISOInheritanceRelation  
`setName(name)` Set name of inheritance relation  
`setDescription(description)` Set description of inheritance relation  
`setUniqueInstance(uniqueInstance)` Set TRUE if it's a unique instance, FALSE otherwise  
`setSubtype(featureType)` Set subtype, object of class ISOFeatureType  
`setSupertype(featureType)` Set supertype, object of class ISOFeatureType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

ISOKeywords

*ISOKeywords*

---

**Description**

ISOKeywords

**Usage**

ISOKeywords

**Format**[R6Class](#) object.**Value**Object of [R6Class](#) for modelling a ISO set of keywords**Fields**

keyword

type

**Methods**`new(xml)` This method is used to instantiate an ISOKeywords`addKeyword(keyword)` Adds a keyword`delKeyword(keyword)` Deletes a keyword`setKeywordType(keywordType)` Sets the keyword type`setThesaurusName(thesaurusName)` Sets the thesaurus name**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondel1@gmail.com&gt;

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#a basic keyword set
md <- ISOKeywords$new()
md$addKeyword("keyword1")
md$addKeyword("keyword2")
md$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle("General")
md$setThesaurusName(th)
xml <- md$encode()

#a keyword set with anchors
md <- ISOKeywords$new()
kwd1 <- ISOAnchor$new(
  name = "keyword1",
  href = "http://myvocabulary.geometa/keyword1"
)
md$addKeyword(kwd1)
kwd2 <- ISOAnchor$new(
  name = "keyword2",
  href = "http://myvocabulary.geometa/keyword2"
)
md$addKeyword(kwd2)
md$setKeywordType("theme")
xml <- md$encode()
```

ISOKeywordType

*ISOKeywordType***Description**

ISOKeywordType

**Usage**

ISOKeywordType

**Format**[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an ISO KeywordType**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOKeywordType

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOKeywordType$values(labels = TRUE)

#place keywordType
place <- ISOKeywordType$new(value = "place")
```

---

ISOLanguage

*ISOLanguage*

---

**Description**

ISOLanguage

**Usage**

ISOLanguage

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Language

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOLanguage

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

## References

ISO 19115:2003 - Geographic information – Metadata

## Examples

```
#possible values
values <- ISOLanguage$values(labels = TRUE)

#english language
eng <- ISOLanguage$new(value = "eng")
```

---

ISOLegalConstraints    *ISOLegalConstraints*

---

## Description

ISOLegalConstraints

## Usage

ISOLegalConstraints

## Format

[R6Class](#) object.

## Value

Object of [R6Class](#) for modelling an ISO LegalConstraints

## Fields

accessConstraints  
useConstraints  
otherConstraints

## Methods

`new(xml)` This method is used to instantiate an ISOLegalConstraints

`addAccessConstraint(constraint)` Adds an access constraint, as object of class "character" or class ISORestriction. If an object of class "character" is specified, it must match the accepted values given by ISORestriction\$values().

`delAccessConstraint(constraint)` Deletes an access constraint, as object of class "character" or class ISORestriction. If an object of class "character" is specified, it must match the accepted values given by ISORestriction\$values().

`addUseConstraint(constraint)` Adds a use constraint, as object of class "character" or class `ISORestriction`. If an object of class "character" is specified, it must match the accepted values given by `ISORestriction$values()`.

`delUseConstraint(constraint)` Deletes a use constraint, as object of class "character" or class `ISORestriction`. If an object of class "character" is specified, it must match the accepted values given by `ISORestriction$values()`.

`addOtherConstraint(constraint)` Adds an other constraint as object of class "character"

`delOtherConstraint(constraint)` Deletes an other constraint as object of class "character"

### Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

### References

ISO 19115:2003 - Geographic information – Metadata

### Examples

```
#create object
md <- ISOLegalConstraints$new()
md$addUseLimitation("limitation1")
md$addUseLimitation("limitation2")
md$addUseLimitation("limitation3")
md$addAccessConstraint("copyright")
md$addAccessConstraint("license")
md$addUseConstraint("copyright")
md$addUseConstraint("license")

xml <- md$encode()
```

---

ISOLength

*ISOLength*

---

### Description

ISOLength

### Usage

ISOLength

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO Length measure

**Fields**

value

**Methods**

`new(xml,value,uom,useUomURI)` This method is used to instantiate an ISOLength. The `uom` argument represents the symbol of unit of measure used. The parameter `useUomURI` can be used to set the uom as URI, its default value is FALSE.

**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondel1@gmail.com&gt;

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

ISOLineage

*ISOLineage***Description**

ISOLineage

**Usage**

ISOLineage

**Format**[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an ISO Lineage**Fields**

statement

**Methods**`new(xml)` This method is used to instantiate an ISOLineage`setStatement(statement)` Sets the statement`addProcessStep(processStep)` Adds a process step (object of class ISOProcessStep)`delProcessStep(processStep)` Deletes a process step`addSource(source)` Adds a source (object of class ISOSource)`delSource(source)` Deletes a source

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```

lineage <- ISOLineage$new()
  lineage$setState("statement")

#add a process step
ps <- ISOProcessStep$new()
ps$setDescription("description")
ps$setRationale("rationale")
ps$setDateTime(ISOdate(2015, 1, 1, 23, 59, 59))
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone") #and more responsible party properties..
ps$addProcessor(rp)
lineage$addProcessStep(ps)

#add a source
src <- ISOSource$new()
src$setDescription("description")
src$setScaleDenominator(1L)
rs <- ISOReferenceSystem$new()
rsId <- ISOReferenceIdentifier$new(code = "4326", codeSpace = "EPSG")
rs$setReferenceSystemIdentifier(rsId)
src$setReferenceSystem(rs)
cit <- ISOCitation$new()
cit$setTitle("sometitle") #and more citation properties...
src$setCitation(cit)
extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
src$addExtent(extent)
lineage$addSource(src)

xml <- lineage$encode()

```

---

ISOListedValue

*ISOListedValue*

---

**Description**

ISOListedValue



**Usage**

ISOListedValue

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOListedValue

**Fields**

code

**Methods**

`new(xml)` This method is used to instantiate an ISOListedValue

`setLabel(label)` Sets the label

`setCode(code)` Sets the code

`setDefinition(definition)` Sets the definition

`setDefinitionReference(definitionReference)` Sets the definition reference

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
val <- ISOListedValue$new()
val$setCode("code1")
val$setLabel("label1")
val$setDefinition("definition1")
xml <- val$encode()
```

ISOLocalName

*ISOLocalName*

---

**Description**

ISOLocalName

**Usage**

ISOLocalName

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO LocalName

**Fields**

value

**Methods**

new(xml, value) This method is used to instantiate an ISOLocalName

**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondel1@gmail.com&gt;

**References**ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOMaintenanceFrequency

*ISOMaintenanceFrequency*

---

**Description**

ISOMaintenanceFrequency

**Usage**

ISOMaintenanceFrequency

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO MaintenanceFrequency

**Fields**

value

**Methods**

new(xml, value, description) This method is used to instantiate an ISOMaintenanceFrequency

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOMaintenanceFrequency$values(labels = TRUE)

#daily frequency
daily <- ISOMaintenanceFrequency$new(value = "daily")
```

---

ISOMaintenanceInformation

*ISOMaintenanceInformation*

---

**Description**

ISOMaintenanceInformation

**Usage**

ISOMaintenanceInformation

**Format**

R6Class object.

**Value**

Object of [R6Class](#) for modelling an ISO MaintenanceInformation

**Fields**

maintenanceAndUpdateFrequency

**Methods**

`new(xml)` This method is used to instantiate an ISOMaintenanceInformation  
`setMaintenanceFrequency(frequency)` Sets the maintenance and update frequency

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOMaintenanceInformation$new()
md$setMaintenanceFrequency("daily")
xml <- md$encode()
```

---

ISOMeasure

*ISOMeasure*

---

**Description**

ISOMeasure

**Usage**

ISOMeasure

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Measure

**Fields**

value

**Methods**

`new(xml,value, uom, useUomURI)` This method is used to instantiate an ISOMeasure. The `uom` argument represents the symbol of unit of measure used. The parameter `useUomURI` can be used to set the uom as URI, its default value is FALSE.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOMemberName

*ISOMemberName*

---

**Description**

ISOMemberName

**Usage**

ISOMemberName

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOMemberName

**Fields**

`aName`

`attributeType`

**Methods**

`new(xml, aName, attributeType)` This method is used to instantiate an ISOMemberName

`setName(aName)` Set the `aName`, object of class character

`setAttributeType(attributeType)` Set the attribute type, object of class ISOTypeName or character

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOMetadata

*ISOMetadata*

---

**Description**

ISOMetadata

**Usage**

ISOMetadata

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Metadata

**Fields**

fileIdentifier  
language  
characterSet  
parentIdentifier  
hierarchyLevel  
hierarchyLevelName  
contact  
dateStamp  
metadataStandardName  
metadataStandardVersion  
dataSetURI  
spatialRepresentationInfo  
referenceSystemInfo  
metadataExtensionInfo  
identificationInfo  
contentInfo  
distributionInfo  
dataQualityInfo  
metadataMaintenance

**Methods**

`new(xml)` This method is used to instantiate an ISOMetadata

`setFileIdentifier(fileIdentifier)` Sets the file identifier

`setLanguage{locale}` Sets the locale

`setCharacterSet(charset)` Sets the character set

`setParentIdentifier(parentIdentifier)` Sets the parentIdentifier

`addHierarchyLevel(level)` Adds the hierarchy level

`setHierarchyLevel(level)` Sets the hierarchy level

`delHierarchyLevel(level)` Deletes the hierarchy level

`addHierarchyLevelName(levelName)` Adds the hierarchy level name

`delHierarchyLevelName(levelName)` Deletes the hierarchy level name

`addContact(contact)` Adds a contact as object of class ISOResponsibleParty

`delContact(contact)` Deletes a contact as object of class ISOResponsibleParty

`setDateStamp(date)` Sets the date stamp

`setMetadataStandardName(name)` Sets the metadata standard name

`setMetadataStandardVersion(version)` Sets the metadata standard version

`setDataSetURI(dataSetURI)` Sets the metadata dataSet URI

`addSpatialRepresentationInfo(spatialRepresentationInfo)` Adds a spatial representation

`setSpatialRepresentationInfo(spatialRepresentationInfo)` Sets a spatial representation

`delSpatialRepresentationInfo(spatialRepresentationInfo)` Deletes a spatial representation

`addReferenceSystemInfo(referenceSystemInfo)` Adds a reference system

`setReferenceSystemInfo(referenceSystemInfo)` Sets the reference system

`delReferenceSystemInfo(referenceSystemInfo)` Deletes a reference system

`addMetadataExtensionInfo(extensionInfo)` Adds extension info, object of class ISOMetadataExtensionInformation

`delMetadataExtensionInfo(extensionInfo)` Deletes extension info, object of class ISOMetadataExtensionInformation

`addIdentificationInfo(identificationInfo)` Adds a data identification

`setIdentificationInfo(identificationInfo)` Sets the data identification

`delIdentificationInfo(identificationInfo)` Deletes a data identification

`addContentInfo(contentInfo)` Adds a content info, either an object of class ISOCoverageDescription for coverage data, or ISOFeatureCatalogueDescription for vector data.

`delContentInfo(contentInfo)` Deletes a content info, either an object of class ISOCoverageDescription for coverage data, or ISOFeatureCatalogueDescription for vector data.

`setDistributionInfo(distributionInfo)` Sets the distribution

`addDataQualityInfo(dataQualityInfo)` Adds a data quality

`setDataQualityInfo(dataQualityInfo)` Sets the data quality

`delDataQualityInfo(dataQualityInfo)` Deletes a data quality

`setMetadataMaintenance(metadataMaintenance)` Sets a metadata maintenance as object of class ISOMaintenanceInformation

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#example 1 - WRITE: Create an ISO metadata and encode it as XML
#####
md = ISOMetadata$new()
md$setFileIdentifier("my-metadata-identifier")
md$setParentIdentifier("my-parent-metadata-identifier")
md$setCharacterSet("utf8")
md$setLanguage("eng")
md$setDateStamp(ISOdate(2015, 1, 1, 1))
md$setMetadataStandardName("ISO 19115:2003/19139")
md$setMetadataStandardVersion("1.0")
md$setDataSetURI("my-dataset-identifier")

#add 3 contacts
for(i in 1:3){
  rp <- ISOResponsibleParty$new()
  rp$setIndividualName(paste0("someone",i))
  rp$setOrganisationName("somewhere")
  rp$setPositionName(paste0("someposition",i))
  rp$setRole("pointOfContact")
  contact <- ISOContact$new()
  phone <- ISOTelephone$new()
  phone$setVoice(paste0("myphonenumber",i))
  phone$setFacsimile(paste0("myfacsimile",i))
  contact$setPhone(phone)
  address <- ISOAddress$new()
  address$setDeliveryPoint("theaddress")
  address$setCity("thecity")
  address$setPostalCode("111")
  address$setCountry("France")
  address$setEmail("someone@theorg.org")
  contact$setAddress(address)
  res <- ISOOnlineResource$new()
  res$setLinkage("http://somelink")
  res$setName("someresourcename")
  contact$setOnlineResource(res)
  rp$setContactInfo(contact)
  md$addContact(rp)
}

#VectorSpatialRepresentation
vsr <- ISOVectorSpatialRepresentation$new()
vsr$setTopologyLevel("geometryOnly")
geomObject <- ISOGeometricObjects$new()
```



```
geomObject$setGeometricObjectType("surface")
geomObject$setGeometricObjectCount(5L)
vsr$setGeometricObjects(geomObject)
md$addSpatialRepresentationInfo(vsr)

#ReferenceSystem
rs <- ISOReferenceSystem$new()
rsId <- ISOReferenceIdentifier$new(code = "4326", codeSpace = "EPSG")
rs$setReferenceSystemIdentifier(rsId)
md$setReferenceSystemInfo(rs)

#data identification
ident <- ISODataIdentification$new()
ident$setAbstract("abstract")
ident$setPurpose("purpose")
ident$addCredit("credit1")
ident$addCredit("credit2")
ident$addCredit("credit3")
ident$addStatus("completed")
ident$setLanguage("eng")
ident$setCharacterSet("utf8")
ident$addTopicCategory("biota")
ident$addTopicCategory("oceans")

#adding a point of contact
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenummer")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://somelink")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
ident$addPointOfContact(rp)

#citation
ct <- ISOCitation$new()
ct$setTitle("sometitle")
d <- ISODate$new()
```

```

d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
ct$addDate(d)
ct$setEdition("1.0")
ct$setEditionDate(as.Date(ISOdate(2015, 1, 1, 1)))
ct$setIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$setPresentationForm("mapDigital")
ct$setCitedResponsibleParty(rp)
ident$setCitation(ct)

#graphic overview
go1 <- ISOBrowseGraphic$new(
  fileName = "http://www.somefile.org/png1",
  fileDescription = "Map Overview 1",
  fileType = "image/png"
)
go2 <- ISOBrowseGraphic$new(
  fileName = "http://www.somefile.org/png2",
  fileDescription = "Map Overview 2",
  fileType = "image/png"
)
ident$addGraphicOverview(go1)
ident$addGraphicOverview(go2)

#maintenance information
mi <- ISOMaintenanceInformation$new()
mi$setMaintenanceFrequency("daily")
ident$setResourceMaintenance(mi)

#adding legal constraints
lc <- ISOLegalConstraints$new()
lc$addUseLimitation("limitation1")
lc$addUseLimitation("limitation2")
lc$addUseLimitation("limitation3")
lc$addAccessConstraint("copyright")
lc$addAccessConstraint("license")
lc$addUseConstraint("copyright")
lc$addUseConstraint("license")
ident$addResourceConstraints(lc)

#adding security constraints
sc <- ISOSecurityConstraints$new()
sc$setClassification("secret")
sc$setUserNote("ultra secret")
sc$setClassificationSystem("no classification in particular")
sc$setHandlingDescription("description")
ident$addResourceConstraints(sc)

#adding extent
extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
ident$setExtent(extent)

```

```

#add keywords
kwds <- ISOKeywords$new()
kwds$addKeyword("keyword1")
kwds$addKeyword("keyword2")
kwds$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle("General")
th$addDate(d)
kwds$setThesaurusName(th)
ident$addKeywords(kwds)

#supplementalInformation
ident$setSupplementalInformation("some additional information")

#spatial representation type
ident$addSpatialRepresentationType("vector")

md$setIdentificationInfo(ident)

#Distribution
distrib <- ISODistribution$new()
dto <- ISODigitalTransferOptions$new()
for(i in 1:3){
  or <- ISOOnlineResource$new()
  or$setLinkage(paste0("http://somelink",i))
  or$setName(paste0("name",i))
  or$setDescription(paste0("description",i))
  or$setProtocol("WWW:LINK-1.0-http--link")
  dto$addOnlineResource(or)
}
distrib$setDigitalTransferOptions(dto)
md$setDistributionInfo(distrib)

#Data Quality
dq <- ISODataQuality$new()
scope <- ISOScope$new()
scope$setLevel("dataset")
dq$setScope(scope)

#add report
dc <- ISODomainConsistency$new()
result <- ISOConformanceResult$new()
spec <- ISOCitation$new()
spec$setTitle("specification title")
spec$setAlternateTitle("specification alternate title")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
spec$addDate(d)
result$setSpecification(spec)
result$setExplanation("some explanation about the conformance")
result$setPass(TRUE)

```

```

dc$addResult(result)
dq$addReport(dc)

#add lineage
lineage <- ISOLineage$new()
lineage$setStatement("statement")
dq$setLineage(lineage)

md$setDataQualityInfo(dq)

#Content Information
#-----
#add a feature catalogue description
fcd <- ISOFeatureCatalogueDescription$new()
fcd$setComplianceCode(FALSE)
fcd$addLanguage("eng")
fcd$setIncludedWithDataset(FALSE)
cit = ISOCitation$new()
contact = ISOContact$new()
fcLink <- ISOOnlineResource$new()
fcLink$setLinkage("http://somelink/featurecatalogue")
contact$setOnlineResource(fcLink)
rp = ISOResponsibleParty$new()
rp$setContactInfo(contact)
cit$setCitedResponsibleParty(rp)
fcd$addFeatureCatalogueCitation(cit)
md$addContentInfo(fcd)

#XML representation of the ISOMetadata
xml <- md$encode()

#example 2 - READ: Create an ISO metadata reading from XML
#####
require(XML)
xmlfile <- system.file("extdata/examples", "metadata.xml", package = "geometa")
xml <- xmlParse(xmlfile)
md <- ISOMetadata$new(xml = xml)

```

---

ISOMetadataExtensionInformation

*ISOMetadataExtensionInformation*

---

## Description

ISOMetadataExtensionInformation

## Usage

ISOMetadataExtensionInformation

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO MetadataExtensionInformation

**Fields**

extensionOnLineResource  
extendedElementInformation

**Methods**

new(xml) This method is used to instantiate an ISOMetadataExtensionInformation  
setOnlineResource(onlineResource) Sets an online resource (object of class ISOOnlineResource)  
addElement(element) Adds an element (object of class ISOExtendedElementInformation)  
delElement(element) Deletes an element (object of class ISOExtendedElementInformation)

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create an extended element information
elem <- ISOExtendedElementInformation$new()
elem$setName("name")
elem$setShortName("shortName")
elem$setDomainCode(1L)
elem$setDefinition("some definition")
elem$setObligation("mandatory")
elem$setCondition("no condition")
elem$setDatatype("characterString")
elem$setMaximumOccurrence("string")
elem$setDomainValue("value")
elem$addParentEntity("none")
elem$setRule("rule")
elem$addRationale("rationale")
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
contact <- ISOContact$new()
phone <- ISOTelephone$new()
```

```
phone$setVoice("myphonenumbr")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somenam")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
elem$addSource(rp)

md <- ISOMetadataExtensionInformation$new()
md$addElement(elem)

xml <- md$encode()
```

---

ISOMetadataNamespace *ISOMetadataNamespace*

---

## Description

ISOMetadataNamespace

## Usage

ISOMetadataNamespace

## Format

[R6Class](#) object.

## Value

Object of [R6Class](#) for modelling an ISO Metadata Namespace

## Methods

`new(id, uri)` This method is used to instantiate an ISOMetadata

## Note

ISO class used internally by geometa for specifying XML namespaces

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

ISOMetaIdentifier      *ISOMetaIdentifier*

---

**Description**

ISOMetaIdentifier

**Usage**

ISOMetaIdentifier

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO MetaIdentifier

**Methods**

`new(xml, code)` This method is used to instantiate an ISOMetaIdentifier

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOMetaIdentifier$new(code = "identifier")
xml <- md$encode()
```

---

ISOMimeType	<i>ISOMimeType</i>
-------------	--------------------

---

**Description**

ISOMimeType

**Usage**

ISOMimeType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO MimeFileType

**Methods**

`new(xml, type, name)` This method is used to instantiate an ISOMimeType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19139:2007 Geographic information – XML

**Examples**

```
md <- ISOMimeType$new(type = "somemimetype", name = "Mime type name")
xml <- md$encode()
```



---

ISOMultiplicity	<i>ISOMultiplicity</i>
-----------------	------------------------

---

**Description**

ISOMultiplicity

**Usage**

```
ISOMultiplicity
```

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOMultiplicity

**Fields**

value

**Methods**

`new(xml, lower, upper)` This method is used to instantiate an ISOMultiplicity. The range is specified by two arguments lower and upper.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

**Examples**

```
md <- ISOMultiplicity$new(lower = 1, upper = Inf)
xml <- md$encode()
```

---

ISOMultiplicityRange *ISOMultiplicityRange*

---

**Description**

ISOMultiplicityRange

**Usage**

ISOMultiplicityRange

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO MultiplicityRange

**Fields**

value

**Methods**

`new(xml, lower, upper)` This method is used to instantiate an ISOMultiplicityRange

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

**Examples**

```
md <- ISOMultiplicityRange$new(lower = 1, upper = Inf)
xml <- md$encode()
```

---

ISOobligation

*ISOobligation*

---

**Description**

ISOobligation

**Usage**

ISOobligation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Obligation

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOobligation

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOobligation$values(labels = TRUE)

#mandatory value
mandatory <- ISOobligation$new(value = "mandatory")
```

---

ISOOnlineResource      *ISOOnlineResource*

---

**Description**

ISOOnlineResource

**Usage**

ISOOnlineResource

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Online Resource

**Fields**

linkage  
protocol  
name  
description

**Methods**

`new(xml)` This method is used to instantiate an ISOOnlineResource  
`setLinkage(linkage)` Sets the linkage (URL), an object of class character or ISOUr1  
`setProtocol(protocol)` Sets the protocol  
`setName(name)` Sets the name  
`setDescription(description)` Sets the description

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOOnlineResource$new()
md$setLinkage("http://somelink")
md$setName("name")
md$setDescription("description")
md$setProtocol("protocol")
xml <- md$encode()
```

---

ISOOperationMetadata *ISOOperationMetadata*

---

**Description**

ISOOperationMetadata

**Usage**

ISOOperationMetadata

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOOperationMetadata

**Fields**

operationName  
DCP  
operationDescription  
invocationName  
parameters  
connectPoint  
dependsOn

**Methods**

`new(xml)` This method is used to instantiate an ISOOperationMetadata  
`setOperationName(operationName)` Set the operation name  
`addDCP(dcp)` Add a DCP  
`delDCP(dcp)` Deletes a DCP  
`setOperationDescription(operationDescription)` Set the operation description

`setInvocationName(invocationName)` Set the invocation name  
`addParameter(parameter)` Add a parameter, object of class `ISOPParameter`  
`delParameter(parameter)` Deletes a parameter, object of class `ISOPParameter`  
`addConnectPoint(connectPoint)` Add a connect point, object of class `ISOOnlineResource`  
`delConnectPoint(connectPoint)` Deletes a connect point, object of class `ISOOnlineResource`  
`addDependentOperationMetadata(operationMetadata)` Add dependent operation metadata  
`delDependentOperationMetadata(operationMetadata)` Deletes dependent operation metadata

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Services

**Examples**

```
md <- ISOOperationMetadata$new()
xml <- md$encode()
```

---

ISOPParameter

*ISOPParameter*

---

**Description**

ISOPParameter

**Usage**

ISOPParameter

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOPParameter

**Fields**

name  
 direction  
 description  
 optionality  
 repeatability  
 valueType

**Methods**

`new(xml)` This method is used to instantiate an ISOParameter

`setName(name, attributeType)` Sets the parameter name (character) and attributeType (ISOTypeName or character)

`setDirection(direction)` Sets the direction, an object of class ISOParameterDirection or any character value among ISOParameterDirection\$values()

`setDescription(description)` Sets the parameter description

`setOptionality(optional)` Set whether the parameter is optional (TRUE), FALSE otherwise

`setRepeatability(repeatable)` Set whether the parameter is repeatable (TRUE), FALSE otherwise

`setValueType(valueType)` Sets the type of parameter value, object of class ISOTypeName or character

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Services

**Examples**

```
md <- ISOParameter$new()
md$setName("name", "attType")
md$setDirection("in")
md$setDescription("description")
md$setOptionality(FALSE)
md$setRepeatability(FALSE)
md$setValueType("CharacterString")
xml <- md$encode()
```

---

ISOParameterDirection *ISOParameterDirection*

---

**Description**

ISOParameterDirection

**Usage**

ISOParameterDirection

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOPParameterDirection

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOPParameterDirection

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19119:2005 - Geographic information – Services

**Examples**

```
#possible values
values <- ISOPParameterDirection$values(labels = TRUE)

#paramDir
paramDir <- ISOPParameterDirection$new(value = "in")
```

---

ISOPresentationForm    *ISOPresentationForm*

---

**Description**

ISOPresentationForm

**Usage**

ISOPresentationForm

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO PresentationForm

**Fields**

value



**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOPresentationForm

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOPresentationForm$values(labels = TRUE)

#mapDigital type
map <- ISOPresentationForm$new(value = "mapDigital")
```

---

ISOProcessStep

*ISOProcessStep*

---

**Description**

ISOProcessStep

**Usage**

ISOProcessStep

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ProcessStep

**Fields**

description  
rationale  
dateTime  
processor

**Methods**

new(xml) This method is used to instantiate an ISOProcessStep  
 setDescription(description) Sets the process step description  
 setRationale(rationale) Sets the process step rationale  
 setDateTime(dateTime) Sets the date time  
**addProcessor(processor)** Adds a processor (object of class ISOResponsibleParty)  
**delProcessor(processor)** Deletes a processor  
**addSource(source)** Adds a source (object of class ISOSource)  
**delSource(source)** Deletes a source

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```

ps <- ISOProcessStep$new()
ps$setDescription("description")
ps$setRationale("rationale")
ps$setDateTime( ISOdate(2015, 1, 1, 23, 59, 59))
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone") #and more responsible party properties..
ps$addProcessor(rp)
xml <- ps$encode()

```

---

ISOPropertyType

*ISOPropertyType*

---

**Description**

ISOPropertyType

**Usage**

ISOPropertyType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOPropertyType

**Fields**

memberName  
definition  
cardinality  
definitionReference

**Inherited methods from** ISOCarrierOfCharacteristics

setFeatureType(featureType) Sets a feature type, object of class ISOFeatureType  
addConstraint(constraint) Adds a constraint, object of class ISOConstraint or character  
delConstraint(constraint) Deletes a constraint, object of class ISOConstraint or character

**Methods**

new(xml, defaults) This method is used to instantiate an ISOPropertyType  
setMemberName(memberName) Sets the member name. Object of class ISOLocalName or "character"  
setDefinition(definition) Sets the definition  
setCardinality(lower, upper) Sets the cardinality boundaries lower and upper of class numeric  
setDefinitionReference(definitionReference) Sets the definition Reference, object of class ISODefinitionReference

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

---

ISORangeDimension      *ISORangeDimension*

---

**Description**

ISORangeDimension

**Usage**

ISORangeDimension

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISORangeDimension

**Fields**

sequenceIdentifier  
descriptor

**Methods**

new(xml) This method is used to instantiate an ISORangeDimension  
setSequenceIdentifier(memberName) Sets the sequence identifier, object of class ISOMemberName  
setDescription(descriptor) Sets the descriptor, object of class character

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create dimension
md <- ISORangeDimension$new()
md$setSequenceIdentifier(ISOMemberName$new(aName = "name", attributeType = "type"))
md$setDescription("descriptor")
xml <- md$encode()
```

---

ISORecord

*ISORecord*

---

**Description**

ISORecord

**Usage**

ISORecord

**Format**

R6Class object.

**Value**

Object of [R6Class](#) for modelling an ISORecord

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISORecord

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISORecordType

*ISORecordType*

---

**Description**

ISORecordType

**Usage**

ISORecordType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISORecordType

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISORecordType

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOReferenceIdentifier

*ISOReferenceIdentifier*

---

**Description**

ISOReferenceIdentifier

**Usage**

ISOReferenceIdentifier

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ReferenceIdentifier

**Methods**

`new(xml, code, codeSpace)` This method is used to instantiate an ISOReferenceIdentifier

`setCodeSpace(codeSpace)` Sets a codeSpace

`setVersion(version)` Sets a version

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOReferenceIdentifier$new(code = "4326", codeSpace = "EPSG")
xml <- md$encode()
```

---

ISOReferenceSystem     *ISOReferenceSystem*

---

**Description**

ISOReferenceSystem

**Usage**

ISOReferenceSystem

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ReferenceSystem

**Fields**

referenceSystemIdentifier

**Methods**

`new(xml, value)` This method is used to instantiate an ISOReferenceSystem

`setReferenceSystemIdentifier(code, codeSpace)` Sets the reference system identifier

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOReferenceSystem$new()
rsId <- ISOReferenceIdentifier$new(code = "4326", codeSpace = "EPSG")
md$setReferenceSystemIdentifier(rsId)
xml <- md$encode()
```

ISORepresentativeFraction

*ISORepresentativeFraction*

---

**Description**

ISORepresentativeFraction

**Usage**

ISORepresentativeFraction

**Format**[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an ISO RepresentativeFraction**Fields**

denominator

**Methods**`new(xml, denominator)` This method is used to instantiate an ISORepresentativeFraction`setDenominator(denominator)` Sets the denominator**Author(s)**

Emmanuel Blondel &lt;emmanuel.blondel1@gmail.com&gt;

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
fr <- ISORepresentativeFraction$new(denominator = 1L)
xml1 <- fr$encode()
fr$setDenominator(2L)
xml2 <- fr$encode()
```



---

ISOResponsibleParty    *ISOResponsibleParty*

---

**Description**

ISOResponsibleParty

**Usage**

ISOResponsibleParty

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ResponsibleParty

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOResponsibleParty

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create a responsible party element
md <- ISOResponsibleParty$new()
md$setIndividualName("someone")
md$setOrganisationName("somewhere")
md$setPositionName("someposition")
md$setRole("pointOfContact")

#add contact
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
```

```

contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
md$setContactInfo(contact)

xml <- md$encode()

```

---

ISORestriction

*ISOHierarchyLevel*


---

### Description

ISOHierarchyLevel

### Usage

ISORestriction

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO Restriction

### Fields

value

### Methods

`new(xml, value, description)` This method is used to instantiate an ISORestriction

### Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

### References

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISORole$values(labels = TRUE)

#copyright restriction
cr <- ISORole$new(value = "copyright")
```

---

ISORole

*ISORole*

---

**Description**

ISORole

**Usage**

ISORole

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Role

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISORole

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISORole$values(labels = TRUE)

#publisher restriction
role <- ISORole$new(value = "publisher")
```

---

ISORoleType

*ISORoleType*

---

**Description**

ISORoleType

**Usage**

ISORoleType

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO RoleType

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISORoleType

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19110:2005 Methodology for Feature cataloguing

**Examples**

```
#possible values
values <- ISORoleType$values(labels = TRUE)

#some charset
ordinaryType <- ISORoleType$new(value = "ordinary")
```

---

ISOScale

*ISOScale*

---

**Description**

ISOScale

**Usage**

ISOScale

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOScale measure

**Fields**

value

**Methods**

`new(xml, value, uom, useUomURI)` This method is used to instantiate an ISOScale. The `uom` argument represents the symbol of unit of measure used. The parameter `useUomURI` can be used to set the `uom` as URI, its default value is FALSE.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOScope

*ISOScope*

---

**Description**

ISOScope

**Usage**

ISOScope

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Scope

**Fields**

level

**Methods**

`new(xml)` This method is used to instantiate an ISOScope

`setLevel(level)` Sets the scope level

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOScope$new()
md$setLevel("dataset")
xml <- md$encode()
```

---

ISOScopedName	<i>ISOScopedName</i>
---------------	----------------------

---

**Description**

ISOScopedName

**Usage**

ISOScopedName

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ScopedName

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOScopedName

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOSecurityConstraints	<i>ISOSecurityConstraints</i>
------------------------	-------------------------------

---

**Description**

ISOSecurityConstraints

**Usage**

ISOSecurityConstraints

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO SecurityConstraints

**Fields**

classification  
userNote  
classificationSystem  
handlingDescription

**Methods**

new(xml) This method is used to instantiate an ISOSecurityConstraints  
setClassification(classification) Adds a classification, as object of class "character" or class ISOClassification. If an object of class "character" is specified, it must match the accepted values given by ISOClassification\$values().  
setUserNote(userNote) Sets a user note as object of class "character"  
setClassificationSystem(classificationSystem) Sets a classification system as object of class "character"  
setHandlingDescription(handlingDescription) Sets a handling description as object of class "character"

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create object
md <- ISOSecurityConstraints$new()
md$setClassification("secret")
md$setUserNote("ultra secret")
md$setClassificationSystem("no classification in particular")
md$setHandlingDescription("description")

xml <- md$encode()
```



---

ISOServiceIdentification  
*ISOServiceIdentification*

---

**Description**

ISOServiceIdentification

**Usage**

ISOServiceIdentification

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO ServiceIdentification

**Fields**

serviceType  
serviceTypeVersion  
accessProperties  
restrictions  
keywords  
extent  
coupledResource  
couplingType  
containsOperations  
operatesOn

**Inherited methods**

setCitation(citation) Sets an object of class ISOCitation  
setAbstract(abstract) Sets an abstract (object of class "character")  
setPurpose(purpose) Sets a purpose (object of class "character")  
addCredit(credit) Adds a credit (object of class "character")  
delCredit(credit) Deletes a credit (object of class "character")  
addStatus(status) Adds a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().

delStatus(status) Deletes a status, as object of class "character" or class ISOStatus. If an object of class "character" is specified, it must match the accepted progress status values ISOStatus\$values().

addPointOfContact(pointOfContact) Adds an object of class ISOResponsibleParty

delPointOfContact(pointOfContact) Deletes an object of class ISOResponsibleParty

addResourceMaintenance(resourceMaintenance) Adds a resource maintenance information as object of class ISOMaintenanceInformation.

setResourceMaintenance(resourceMaintenance) Sets a resource maintenance information as object of class ISOMaintenanceInformation.

delResourceMaintenance(resourceMaintenance) Deletes a resource maintenance information as object of class ISOMaintenanceInformation.

addGraphicOverview(graphicOverview) Adds an object of class ISOBrowseGraphic

setGraphicOverview(graphicOverview) Sets an object of class ISOBrowseGraphic

delGraphicOverview(graphicOverview) Deletes an object of class ISOBrowseGraphic

addKeywords(keywords) Adds a set of keywords as object of class ISOKeywords

setKeywords(keywords) Sets a set of keywords as object of class ISOKeywords

delKeywords(keywords) Deletes a set of keywords as object of class ISOKeywords

addResourceConstraints(resourceConstraints) Adds an object of class ISOLegalConstraints

setResourceConstraints(resourceConstraints) Sets an object of class ISOLegalConstraints

delResourceConstraints(resourceConstraints) Deletes an object of class ISOLegalConstraints

## Methods

new(xml, value) This method is used to instantiate an ISOServiceIdentification

## Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

## References

ISO 19115:2003 - Geographic information – Services

## Examples

```
#encoding
md <- ISOServiceIdentification$new()
md$setAbstract("abstract")
md$setPurpose("purpose")

#adding a point of contact
rp <- ISOResponsibleParty$new()
rp$setIndividualName("someone")
rp$setOrganisationName("somewhere")
rp$setPositionName("someposition")
rp$setRole("pointOfContact")
```

```
contact <- ISOContact$new()
phone <- ISOTelephone$new()
phone$setVoice("myphonenumber")
phone$setFacsimile("myfacsimile")
contact$setPhone(phone)
address <- ISOAddress$new()
address$setDeliveryPoint("theaddress")
address$setCity("thecity")
address$setPostalCode("111")
address$setCountry("France")
address$setEmail("someone@theorg.org")
contact$setAddress(address)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
md$addPointOfContact(rp)

#citation
ct <- ISOCitation$new()
ct$setTitle("sometitle")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
ct$addDate(d)
ct$setEdition("1.0")
ct$setEditionDate(ISOdate(2015,1,1))
ct$setIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$setPresentationForm("mapDigital")
ct$setCitedResponsibleParty(rp)
md$setCitation(ct)

#graphic overview
go <- ISOBrowseGraphic$new(
  fileName = "http://www.somefile.org/png",
  fileDescription = "Map Overview",
  fileType = "image/png"
)
md$setGraphicOverview(go)

#maintenance information
mi <- ISOMaintenanceInformation$new()
mi$setMaintenanceFrequency("daily")
md$setResourceMaintenance(mi)

#adding legal constraints
lc <- ISOLegalConstraints$new()
lc$addUseLimitation("limitation1")
lc$addUseLimitation("limitation2")
lc$addUseLimitation("limitation3")
lc$addAccessConstraint("copyright")
lc$addAccessConstraint("license")
```

```

lc$addUseConstraint("copyright")
lc$addUseConstraint("license")
md$setResourceConstraints(lc)

#specific elements to service identification
md$setServiceType("Fishery data harmonization process")
md$addServiceTypeVersion("1.0")
orderProcess <- ISOStandardOrderProcess$new()
orderProcess$setFees("fees")
orderProcess$setPlannedAvailableDateTime(ISOdate(2017,7,5,12,0,0))
orderProcess$setOrderingInstructions("instructions")
orderProcess$setTurnaround("turnaround")
md$setAccessProperties(orderProcess)
md$setRestrictions(lc)

kwds <- ISOKeywords$new()
kwds$addKeyword("keyword1")
kwds$addKeyword("keyword2")
kwds$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle("General")
th$addDate(d)
kwds$setThesaurusName(th)
md$addKeywords(kwds)

#adding extent
extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
md$addExtent(extent)

#coupling type
#(here "tight" associated with a particular dataset "my-dataset-identifier")
#see ISOCouplingType$values(labels = T) for other values
md$setCouplingType("tight")
coupledDataset1 <- ISOCoupledResource$new()
coupledDataset1$setOperationName("Rscript")
coupledDataset1$setIdentifier("my-dataset-identifier")
coupledDataset2 <- ISOCoupledResource$new()
coupledDataset2$setOperationName("WPS:Execute")
coupledDataset2$setIdentifier("my-dataset-identifier")
md$addCoupledResource(coupledDataset1)
md$addCoupledResource(coupledDataset2)

#add operation metadata 1 (Rscript)
scriptOp <- ISOOperationMetadata$new()
scriptOp$setOperationName("Rscript")
scriptOp$addDCP("WebServices")
scriptOp$setOperationDescription("WPS Execute")
scriptOp$setInvocationName("identifier")
for(i in 1:3){
  param <- ISOParameter$new()
  param$setName(sprintf("name%s",i), "xs:string")
}

```

```

    param$setDirection("in")
    param$setDescription(sprintf("description%s",i))
    param$setOptionality(FALSE)
    param$setRepeatability(FALSE)
    param$setValueType("xs:string")
    scriptOp$addParameter(param)
  }
  outParam <- ISOPParameter$new()
  outParam$setName("outputname", "xs:string")
  outParam$setDirection("out")
  outParam$setDescription("outputdescription")
  outParam$setOptionality(FALSE)
  outParam$setRepeatability(FALSE)
  outParam$setValueType("xs:string")
  scriptOp$addParameter(outParam)
  or <- ISOOnlineResource$new()
  or$setLinkage("http://somelink/myrscript.R")
  or$setName("R script name")
  or$setDescription("R script description")
  or$setProtocol("protocol")
  scriptOp$addConnectPoint(or)
  md$addOperationMetadata(scriptOp)
  #add operation metadata 1 (WPS)
  wpsOp <- ISOOperationMetadata$new()
  wpsOp$setOperationName("WPS:Execute")
  wpsOp$addDCP("WebServices")
  wpsOp$setOperationDescription("WPS Execute")
  invocationName <- "mywpsidentifier"
  wpsOp$setInvocationName(invocationName)
  for(i in 1:3){
    param <- ISOPParameter$new()
    param$setName(sprintf("name%s",i), "xs:string")
    param$setDirection("in")
    param$setDescription(sprintf("description%s",i))
    param$setOptionality(FALSE)
    param$setRepeatability(FALSE)
    param$setValueType("xs:string")
    wpsOp$addParameter(param)
  }
  outParam <- ISOPParameter$new()
  outParam$setName("outputname", "xs:string")
  outParam$setDirection("out")
  outParam$setDescription("outputdescription")
  outParam$setOptionality(FALSE)
  outParam$setRepeatability(FALSE)
  outParam$setValueType("xs:string")
  wpsOp$addParameter(outParam)
  or1 <- ISOOnlineResource$new()
  or1$setLinkage(
    sprintf("http://somelink/wps?request=Execute&version=1.0.0&Identifier=%s",
      invocationName)
  )
  or1$setName("WPS process name")

```

```

or1$setDescription("WPS process description")
or1$setProtocol("protocol")
wpsOp$addConnectPoint(or1)
or2 <- ISOonlineResource$new()
or2$setLinkage("http://somelink/myrscript.R")
or2$setName("Source R script name")
or2$setDescription("Source R script description")
or2$setProtocol("protocol")
wpsOp$addConnectPoint(or2)
md$addOperationMetadata(wpsOp)
xml <- md$encode()

```

---

ISOSource

*ISOSource*


---

### Description

ISOSource

### Usage

ISOSource

### Format

[R6Class](#) object.

### Value

Object of [R6Class](#) for modelling an ISO Source

### Fields

description  
scaleDenominator  
sourceReferenceSystem  
sourceCitation  
sourceExtent  
sourceStep

### Methods

new(xml) This method is used to instantiate an ISOSource  
setDescription(description) Sets the source description  
setScaleDenominator(denominator) Sets the scale denominator (object of class ISORepresentativeFraction or an integer value or other value coercable to integer)

setReferenceSystem(referenceSystem) Sets the source reference system (object of class ISOResourceSystem)  
 setCitation(citation) Sets the source citation (object of class ISOCitation)  
 addExtent(extent) Adds the source extent (object of class ISOExtent)  
 delExtent(extent) Deletes a source extent  
 addProcessStep(processStep) Adds a source process step (object of class ISOProcessStep)  
 delProcessStep(processStep) Deletes a source processStep

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```

src <- ISOSource$new()
src$setDescription("description")
src$setScaleDenominator(1L)

rs <- ISOResourceSystem$new()
rsId <- ISOResourceIdentifier$new(code = "4326", codeSpace = "EPSG")
rs$setResourceSystemIdentifier(rsId)
src$setResourceSystem(rs)

cit <- ISOCitation$new()
cit$setTitle("sometitle") #and more citation properties...
src$setCitation(cit)

extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
src$addExtent(extent)
xml <- src$encode()

```

---

ISOSpatialRepresentation

*ISOSpatialRepresentation*

---

**Description**

ISOSpatialRepresentation

**Usage**

ISOSpatialRepresentation

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO abstract SpatialRepresentation

**Methods**

new(xml) This method is used to instantiate an ISOSpatialRepresentation

**Note**

abstract class

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOSpatialRepresentationType

*ISOSpatialRepresentationType*

---

**Description**

ISOSpatialRepresentationType

**Usage**

ISOSpatialRepresentationType

**Format**

R6Class object.

**Value**

Object of R6Class for modelling an ISO SpatialRepresentationType

**Fields**

value



**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOSpatialRepresentation-Type

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOSpatialRepresentationType$values(labels = TRUE)

#vector example
vectorRep <- ISORestriction$new(value = "vector")
```

---

ISOSpatialTemporalExtent  
*ISOSpatialTemporalExtent*

---

**Description**

ISOSpatialTemporalExtent

**Usage**

ISOSpatialTemporalExtent

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO SpatialTemporalExtent

**Methods**

`new(xml)` This method is used to instantiate an ISOTemporalExtent  
`addSpatialExtent(spatialExtent)` Adds an object of class ISOGeographicExtent  
`setSpatialExtent(spatialExtent)` Sets an object of class ISOGeographicExtent  
`delSpatialExtent(spatialExtent)` Deletes an object of class ISOGeographicExtent

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#create object
md <- ISOSpatialTemporalExtent$new()
start <- ISOdate(2000, 1, 12, 12, 59, 45)
end <- ISOdate(2010, 8, 22, 13, 12, 43)
tp <- GMLTimePeriod$new(beginPosition = start, endPosition = end)
md$setTimePeriod(tp)
spatialExtent <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
md$setSpatialExtent(spatialExtent)

xml <- md$encode()
```

---

ISOStandardOrderProcess

*ISOStandardOrderProcess*

---

**Description**

ISOStandardOrderProcess

**Usage**

ISOStandardOrderProcess

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO StandardOrderProcess

**Fields**

fees  
plannedAvailableDateTime  
orderingInstructions  
turnaround

**Methods**

`new(xml)` This method is used to instantiate an `ISOStandardOrderProcess`  
`setFees(fees)` Sets fees, object of class character  
`setPlannedAvailableDateTime(dateTime)` Sets planned available datetime, object of class `c('POSIXct', 'POSIXlt')`  
`setOrderingInstructions(instructions)` Sets ordering instructions, object of class character  
`setTurnaround(turnaround)` Sets turnaround, object of class character

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```

md <- ISOStandardOrderProcess$new()
md$setFees("fees")
md$setPlannedAvailableDateTime(ISOdate(2017,7,5,12,0,0))
md$setOrderingInstructions("instructions")
md$setTurnaround("turnaround")
xml <- md$encode()
  
```

---

ISOStatus

*ISOStatus*

---

**Description**

ISOStatus

**Usage**

ISOStatus

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO progress status

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOStatus

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOStatus$values(labels = TRUE)

#pending status
pending <- ISOStatus$new(value = "pending")
```

---

ISOTelephone

*ISOTelephone*

---

**Description**

ISOTelephone

**Usage**

ISOTelephone

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO Telephone

**Fields**

voice  
facsimile

**Methods**

`new(xml, value)` This method is used to instantiate an ISOTelephone

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOTelephone$new()  
md$setVoice("myphonenumber")  
md$setFacsimile("myfacsimile")  
xml <- md$encode()
```

---

ISOTemporalExtent	<i>ISOTemporalExtent</i>
-------------------	--------------------------

---

**Description**

ISOTemporalExtent

**Usage**

ISOTemporalExtent

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO TemporalExtent

**Methods**

`new(xml)` This method is used to instantiate an ISOTemporalExtent  
`setTimeInstant(timeInstant)` Sets a time instant  
`setTimePeriod(timePeriod)` Sets a time period

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
te <- ISOTemporalExtent$new()
start <- ISOdate(2000, 1, 12, 12, 59, 45)
end <- ISOdate(2010, 8, 22, 13, 12, 43)
tp <- GMLTimePeriod$new(beginPosition = start, endPosition = end)
te$setTimePeriod(tp)
```

---

ISOTimePeriod	<i>ISOTimePeriod</i>
---------------	----------------------

---

**Description**

ISOTimePeriod

**Usage**

ISOTimePeriod

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an GMLTimePeriod

**Warning**

Deprecated class, use GMLTimePeriod instead

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

ISOTopicCategory	<i>ISOTopicCategory</i>
------------------	-------------------------

---

**Description**

ISOTopicCategory

**Usage**

ISOTopicCategory

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO TopicCategory

**Fields**

value

**Methods**

`new(xml, value, description)` This method is used to instantiate an ISOTopicCategory

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOTopicCategory$values(labels = TRUE)

#biota topic
biota <- ISOTopicCategory$new(value = "biota")
```

---

ISOTopologyLevel	<i>ISOTopologyLevel</i>
------------------	-------------------------

---

**Description**

ISOTopologyLevel

**Usage**

ISOTopologyLevel

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO TopologyLevel

**Fields**

value

**Methods**

`new(xml,value, description)` This method is used to instantiate an ISOTopologyLevel

**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
#possible values
values <- ISOTopologyLevel$values(labels = TRUE)

#geomOnly
geomOnly <- ISOTopologyLevel$new(value = "geometryOnly")
```

---

ISOTypeName

*ISOTypeName*

---

**Description**

ISOTypeName

**Usage**

ISOTypeName

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOTypeName

**Fields**

aName



**Methods**

`new(xml, aName)` This method is used to instantiate an `ISOTypeName`  
`setName(aName)` Sets the `aName`

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

**Examples**

```
typeName <- ISOTypeName$new(aName = "name")
xml <- typeName$encode()
```

---

ISOUnlimitedInteger    *ISOUnlimitedInteger*

---

**Description**

ISOUnlimitedInteger

**Usage**

ISOUnlimitedInteger

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO UnlimitedInteger

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an `ISOUnlimitedInteger`

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO/TS 19103:2005 Geographic information – Conceptual schema language

---

ISOURL

*ISOURL*

---

**Description**

ISOURL

**Usage**

ISOURL

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISOURL

**Fields**

value

**Methods**

`new(xml, value)` This method is used to instantiate an ISOURL

**Note**

Class used by geometa internal XML decoder/encoder

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

---

ISOVectorSpatialRepresentation  
*ISOVectorSpatialRepresentation*

---

**Description**

ISOVectorSpatialRepresentation

**Usage**

ISOVectorSpatialRepresentation

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO VectorSpatialRepresentation

**Fields**

topologyLevel  
geometricObjects

**Methods**

`new(xml)` This method is used to instantiate an ISOVectorSpatialRepresentation  
`setTopologyLevel(topologyLevel)` Sets the topology level  
`addGeometricObject(geometricObjects)` Adds the geometricObjects  
`setGeometricObject(geometricObjects)` Sets the geometricObjects  
`delGeometricObject(geometricObjects)` Deletes the geometricObjects

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
md <- ISOVectorSpatialRepresentation$new()
md$setTopologyLevel("geometryOnly")
geomObject1 <- ISOGeometricObjects$new()
geomObject1$setGeometricObjectType("surface")
geomObject1$setGeometricObjectCount(5L)
md$addGeometricObjects(geomObject1)
xml <- md$encode()
```

---

ISOVerticalExtent      *ISOVerticalExtent*

---

**Description**

ISOVerticalExtent

**Usage**

ISOVerticalExtent

**Format**

[R6Class](#) object.

**Value**

Object of [R6Class](#) for modelling an ISO VerticalExtent

**Methods**

`new(xml)` This method is used to instantiate an ISOVerticalExtent

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**References**

ISO 19115:2003 - Geographic information – Metadata

**Examples**

```
ve <- ISOVerticalExtent$new()
ve$setMinimumValue(0)
ve$setMaximumValue(19)
xml <- ve$encode()
```

# Index

- \*Topic **Binding**
  - ISOBinding, [59](#)
- \*Topic **BoundAssociationRole**
  - ISOBoundAssociationRole, [60](#)
- \*Topic **BoundFeatureAttribute**
  - ISOBoundFeatureAttribute, [61](#)
- \*Topic **CRS**
  - GMLAbstractCRS, [8](#)
- \*Topic **Classification**
  - ISOClassification, [67](#)
- \*Topic **CoupledResource**
  - ISOCoupledResource, [75](#)
- \*Topic **CoverageContentType**
  - ISOCoverageContentType, [77](#)
- \*Topic **DCP**
  - ISODCPList, [89](#)
- \*Topic **Datatype**
  - ISODatatype, [86](#)
- \*Topic **DimensionNameType**
  - ISODimensionNameType, [94](#)
- \*Topic **EvaluationMethodType**
  - ISOEvaluationMethodType, [100](#)
- \*Topic **GML**
  - GMLAbstractCoordinateOperation, [6](#)
  - GMLAbstractCoordinateSystem, [7](#)
  - GMLAbstractCRS, [8](#)
  - GMLAbstractGeneralConversion, [9](#)
  - GMLAbstractGeneralDerivedCRS, [10](#)
  - GMLAbstractGeneralOperationParameter, [11](#)
  - GMLAbstractGML, [12](#)
  - GMLAbstractObject, [13](#)
  - GMLAbstractSingleCRS, [14](#)
  - GMLAbstractSingleOperation, [14](#)
  - GMLAbstractTimeObject, [16](#)
  - GMLAbstractTimePrimitive, [17](#)
  - GMLAffineCS, [18](#)
  - GMLBaseUnit, [18](#)
  - GMLCartesianCS, [19](#)
  - GMLConventionalUnit, [20](#)
  - GMLConversion, [21](#)
  - GMLCoordinateSystemAxis, [22](#)
  - GMLCylindricalCS, [23](#)
  - GMLDefinition, [24](#)
  - GMLDerivedCRS, [25](#)
  - GMLDerivedUnit, [26](#)
  - GMLElement, [27](#)
  - GMLEllipsoidalCS, [28](#)
  - GMLGeodeticCRS, [28](#)
  - GMLLinearCS, [29](#)
  - GMLObliqueCartesianCS, [30](#)
  - GMLOperationMethod, [31](#)
  - GMLOperationParameter, [32](#)
  - GMLOperationParameterGroup, [33](#)
  - GMLPolarCS, [34](#)
  - GMLProjectedCRS, [34](#)
  - GMLSphericalCS, [35](#)
  - GMLTemporalCS, [36](#)
  - GMLTimeCS, [37](#)
  - GMLUnitDefinition, [39](#)
  - GMLUserDefinedCS, [40](#)
  - GMLVerticalCS, [40](#)
- \*Topic **ISO**
  - GMLAbstractCoordinateOperation, [6](#)
  - GMLAbstractCoordinateSystem, [7](#)
  - GMLAbstractCRS, [8](#)
  - GMLAbstractGeneralConversion, [9](#)
  - GMLAbstractGeneralDerivedCRS, [10](#)
  - GMLAbstractGeneralOperationParameter, [11](#)
  - GMLAbstractGML, [12](#)
  - GMLAbstractObject, [13](#)
  - GMLAbstractSingleCRS, [14](#)
  - GMLAbstractSingleOperation, [14](#)
  - GMLAbstractTimeGeometricPrimitive, [15](#)
  - GMLAbstractTimeObject, [16](#)
  - GMLAbstractTimePrimitive, [17](#)

- GMLAffineCS, 18
- GMLBaseUnit, 18
- GMLCartesianCS, 19
- GMLConventionalUnit, 20
- GMLConversion, 21
- GMLCoordinateSystemAxis, 22
- GMLCylindricalCS, 23
- GMLDefinition, 24
- GMLDerivedCRS, 25
- GMLDerivedUnit, 26
- GMLElement, 27
- GMLEllipsoidalCS, 28
- GMLGeodeticCRS, 28
- GMLLinearCS, 29
- GMLObliqueCartesianCS, 30
- GMLOperationMethod, 31
- GMLOperationParameter, 32
- GMLOperationParameterGroup, 33
- GMLPolarCS, 34
- GMLProjectedCRS, 34
- GMLSphericalCS, 35
- GMLTemporalCS, 36
- GMLTimeCS, 37
- GMLTimePeriod, 37
- GMLUnitDefinition, 39
- GMLUserDefinedCS, 40
- GMLVerticalCS, 40
- ISOAbsoluteExternalPositionalAccuracy, 41
- ISOAbstractCatalogue, 42
- ISOAbstractGenericName, 44
- ISOAbstractObject, 45
- ISOAbstractPositionalAccuracy, 46
- ISOAddress, 47
- ISOAnchor, 48
- ISOAngle, 49
- ISOAssociationRole, 50
- ISOBand, 51
- ISOBaseBoolean, 52
- ISOBaseCharacterString, 53
- ISOBaseDate, 54
- ISOBaseDateTime, 55
- ISOBaseDecimal, 56
- ISOBaseInteger, 57
- ISOBaseReal, 58
- ISOBinding, 59
- ISOBoundAssociationRole, 60
- ISOBoundFeatureAttribute, 61
- ISOBrowseGraphic, 62
- ISOCarrierOfCharacteristics, 63
- ISOCellGeometry, 64
- ISOCharacterSet, 65
- ISOCitation, 66
- ISOClassification, 67
- ISOCodelist, 68
- ISOCodeListValue, 69
- ISOConformanceResult, 70
- ISOConstraint, 71
- ISOConstraints, 72
- ISOContact, 73
- ISOContentInformation, 74
- ISOCoupledResource, 75
- ISOCouplingType, 76
- ISOCoverageContentType, 77
- ISOCoverageDescription, 78
- ISODataIdentification, 79
- ISODataQuality, 83
- ISODataQualityAbstractElement, 85
- ISODatatype, 86
- ISODate, 87
- ISODateType, 88
- ISODCPList, 89
- ISODefinitionReference, 90
- ISODefinitionSource, 91
- ISODigitalTransferOptions, 92
- ISODimension, 93
- ISODimensionNameType, 94
- ISODistance, 95
- ISODistribution, 96
- ISODistributor, 97
- ISODomainConsistency, 98
- ISOElementSequence, 99
- ISOEvaluationMethodType, 100
- ISOExtendedElementInformation, 101
- ISOExtent, 103
- ISOFeatureAssociation, 105
- ISOFeatureAttribute, 106
- ISOFeatureCatalogue, 107
- ISOFeatureCatalogueDescription, 110
- ISOFeatureOperation, 111
- ISOFeatureType, 112
- ISOFileName, 114
- ISOFormat, 115
- ISOGeographicBoundingBox, 116
- ISOGeographicDescription, 117

- ISOGeographicExtent, 118
- ISOGeometricObjects, 119
- ISOGeometricObjectType, 120
- ISOGeoreferenceable, 121
- ISOGriddedDataPositionalAccuracy, 122
- ISOGridSpatialRepresentation, 124
- ISOHierarchyLevel, 125
- ISOIdentification, 126
- ISOIdentifier, 128
- ISOInheritanceRelation, 129
- ISOKeywords, 130
- ISOKeywordType, 131
- ISOLanguage, 132
- ISOLegalConstraints, 133
- ISOLength, 134
- ISOLineage, 135
- ISOListedValue, 136
- ISOLocalName, 138
- ISOMaintenanceFrequency, 138
- ISOMaintenanceInformation, 139
- ISOMeasure, 140
- ISOMemberName, 141
- ISOMetadata, 142
- ISOMetadataExtensionInformation, 148
- ISOMetadataNamespace, 150
- ISOMetaIdentifier, 151
- ISOMimeType, 152
- ISOMultiplicity, 153
- ISOMultiplicityRange, 154
- ISOObligation, 155
- ISOOnlineResource, 156
- ISOOperationMetadata, 157
- ISOParameter, 158
- ISOParameterDirection, 159
- ISOPresentationForm, 160
- ISOProcessStep, 161
- ISOPropertyType, 162
- ISORangeDimension, 163
- ISORecord, 164
- ISORecordType, 165
- ISOReferenceIdentifier, 166
- ISOReferenceSystem, 167
- ISORepresentativeFraction, 168
- ISOResponsibleParty, 169
- ISORestriction, 170
- ISORole, 171
- ISORoleType, 172
- ISOScale, 173
- ISOScope, 174
- ISOScopedName, 175
- ISOSecurityConstraints, 175
- ISOServiceIdentification, 177
- ISOSource, 182
- ISOSpatialRepresentation, 183
- ISOSpatialRepresentationType, 184
- ISOSpatialTemporalExtent, 185
- ISOStandardOrderProcess, 186
- ISOStatus, 187
- ISOTelephone, 188
- ISOTemporalExtent, 189
- ISOTimePeriod, 190
- ISOTopicCategory, 190
- ISOTopologyLevel, 191
- ISOTypeName, 192
- ISOUnlimitedInteger, 193
- ISOURL, 194
- ISOVectorSpatialRepresentation, 195
- ISOVerticalExtent, 196
- \*Topic **InheritanceRelation**
  - ISOInheritanceRelation, 129
- \*Topic **Obligation**
  - ISOObligation, 155
- \*Topic **ObliqueCartesian**
  - GMLObliqueCartesianCS, 30
- \*Topic **OperationMetadata**
  - ISOOperationMetadata, 157
- \*Topic **Operation**
  - GMLAbstractCoordinateOperation, 6
  - GMLAbstractSingleOperation, 14
- \*Topic **Restriction**
  - ISORestriction, 170
- \*Topic **StandardOrderProcess**
  - ISOStandardOrderProcess, 186
- \*Topic **URL**
  - ISOURL, 194
- \*Topic **UserDefined**
  - GMLUserDefinedCS, 40
- \*Topic **absolute**
  - ISOAbsoluteExternalPositionalAccuracy, 41
- \*Topic **abstract**
  - GMLAbstractCoordinateOperation, 6
  - GMLAbstractCoordinateSystem, 7

- GMLAbstractCRS, 8
- GMLAbstractGeneralConversion, 9
- GMLAbstractGeneralDerivedCRS, 10
- GMLAbstractGeneralOperationParameter, 11
- GMLAbstractSingleCRS, 14
- GMLAbstractSingleOperation, 14
- GMLAbstractTimeObject, 16
- GMLAbstractTimePrimitive, 17
- ISOAbstractCatalogue, 42
- ISOAbstractGenericName, 44
- ISOAbstractPositionalAccuracy, 46
- ISODataQualityAbstractElement, 85
- \*Topic **accuracy**
  - ISOAbsoluteExternalPositionalAccuracy, 41
  - ISOAbstractPositionalAccuracy, 46
  - ISOGriddedDataPositionalAccuracy, 122
- \*Topic **address**
  - ISOAddress, 47
  - ISOspatialRepresentation, 183
  - ISOVectorSpatialRepresentation, 195
- \*Topic **affine**
  - GMLAffineCS, 18
- \*Topic **anchor**
  - ISOAnchor, 48
- \*Topic **angle**
  - ISOAngle, 49
- \*Topic **association**
  - ISOAssociationRole, 50
  - ISOFeatureAssociation, 105
- \*Topic **axis**
  - GMLCoordinateSystemAxis, 22
- \*Topic **band**
  - ISOBand, 51
- \*Topic **base**
  - GMLBaseUnit, 18
- \*Topic **boolean**
  - ISOBaseBoolean, 52
- \*Topic **browse**
  - ISOBrowseGraphic, 62
- \*Topic **carrierOfCharacteristics**
  - ISOCarrierOfCharacteristics, 63
- \*Topic **cartesian**
  - GMLCartesianCS, 19
- \*Topic **catalogue**
  - ISOAbstractCatalogue, 42
  - ISOFeatureCatalogue, 107
  - ISOFeatureCatalogueDescription, 110
- \*Topic **category**
  - ISOTopicCategory, 190
- \*Topic **cell**
  - ISOCellGeometry, 64
- \*Topic **characterSet**
  - ISOCharacterSet, 65
- \*Topic **character**
  - ISOBaseCharacterString, 53
- \*Topic **charset**
  - ISOCharacterSet, 65
  - ISOTopologyLevel, 191
- \*Topic **citation**
  - ISOCitation, 66
- \*Topic **codelist**
  - ISOCodelist, 68
- \*Topic **code**
  - ISOCodelistValue, 69
- \*Topic **conformance**
  - ISOConformanceResult, 70
- \*Topic **consistency**
  - ISODomainConsistency, 98
- \*Topic **constraints**
  - ISOConstraints, 72
  - ISOLegalConstraints, 133
  - ISOSecurityConstraints, 175
- \*Topic **constraint**
  - ISOConstraint, 71
- \*Topic **contact**
  - ISOContact, 73
- \*Topic **content**
  - ISOContentInformation, 74
- \*Topic **conventional**
  - GMLConventionalUnit, 20
- \*Topic **conversion**
  - GMLAbstractGeneralConversion, 9
  - GMLConversion, 21
- \*Topic **coordinate**
  - GMLAbstractCoordinateOperation, 6
  - GMLAbstractCoordinateSystem, 7
  - GMLAffineCS, 18
  - GMLCartesianCS, 19
  - GMLCoordinateSystemAxis, 22
  - GMLCylindricalCS, 23
  - GMLEllipsoidalCS, 28



- GMLLinearCS, 29
- GMLObliqueCartesianCS, 30
- GMLPolarCS, 34
- GMLSphericalCS, 35
- GMLTemporalCS, 36
- GMLTimeCS, 37
- GMLUserDefinedCS, 40
- GMLVerticalCS, 40
- \*Topic **coupling**
  - ISOCouplingType, 76
- \*Topic **coverage**
  - ISOCoverageDescription, 78
- \*Topic **crs**
  - GMLAbstractGeneralDerivedCRS, 10
  - GMLAbstractSingleCRS, 14
  - GMLDerivedCRS, 25
  - GMLGeodeticCRS, 28
  - GMLProjectedCRS, 34
- \*Topic **cylindrical**
  - GMLCylindricalCS, 23
- \*Topic **data**
  - ISOAbsoluteExternalPositionalAccuracy, 41
  - ISOAbstractPositionalAccuracy, 46
  - ISODataIdentification, 79
  - ISODataQuality, 83
  - ISODataQualityAbstractElement, 85
  - ISOGriddedDataPositionalAccuracy, 122
- \*Topic **datetime**
  - ISOBaseDateTime, 55
- \*Topic **datatype**
  - ISODataType, 88
- \*Topic **date**
  - ISOBaseDate, 54
  - ISODate, 87
- \*Topic **decimal**
  - ISOBaseDecimal, 56
- \*Topic **definition**
  - GMLBaseUnit, 18
  - GMLConventionalUnit, 20
  - GMLDefinition, 24
  - GMLDerivedUnit, 26
  - GMLUnitDefinition, 39
  - ISODefinitionReference, 90
  - ISODefinitionSource, 91
- \*Topic **derived**
  - GMLDerivedCRS, 25
  - GMLDerivedUnit, 26
- \*Topic **description**
  - ISOCoverageDescription, 78
  - ISOFeatureCatalogueDescription, 110
  - ISOGeographicDescription, 117
- \*Topic **dimension**
  - ISODimension, 93
  - ISORangeDimension, 163
- \*Topic **direction**
  - ISOParameterDirection, 159
- \*Topic **distance**
  - ISODistance, 95
- \*Topic **distribution**
  - ISODigitalTransferOptions, 92
  - ISODistribution, 96
- \*Topic **distributor**
  - ISODistributor, 97
- \*Topic **domain**
  - ISODomainConsistency, 98
- \*Topic **element**
  - GMLElement, 27
  - ISOAbstractObject, 45
  - ISOCodelistValue, 69
  - ISODataQualityAbstractElement, 85
  - ISOExtendedElementInformation, 101
  - ISOMetadata, 142
- \*Topic **ellipsoidal**
  - GMLEllipsoidalCS, 28
- \*Topic **extended**
  - ISOExtendedElementInformation, 101
- \*Topic **extension**
  - ISOMetadataExtensionInformation, 148
- \*Topic **extent**
  - ISOExtent, 103
  - ISOGeographicBoundingBox, 116
  - ISOGeographicExtent, 118
  - ISOScope, 174
  - ISOSpatialTemporalExtent, 185
  - ISOTemporalExtent, 189
  - ISOVerticalExtent, 196
- \*Topic **external**
  - ISOAbsoluteExternalPositionalAccuracy, 41
- \*Topic **feature**
  - ISOConstraint, 71
  - ISOFeatureAssociation, 105

- ISOFeatureAttribute, 106
- ISOFeatureCatalogue, 107
- ISOFeatureCatalogueDescription, 110
- ISOFeatureOperation, 111
- ISOFeatureType, 112
- \*Topic **file**
  - ISOFileName, 114
  - ISOMimeType, 152
  - ISOResponsibleParty, 169
  - ISOTelephone, 188
- \*Topic **format**
  - ISOFormat, 115
- \*Topic **form**
  - ISOPresentationForm, 160
- \*Topic **fraction**
  - ISORepresentativeFraction, 168
- \*Topic **general**
  - GMLAbstractGeneralConversion, 9
  - GMLAbstractGeneralOperationParameter, 11
- \*Topic **generic**
  - ISOAbstractGenericName, 44
- \*Topic **geodetic**
  - GMLGeodeticCRS, 28
- \*Topic **geographic**
  - ISOGeographicDescription, 117
- \*Topic **geometric**
  - ISOGeometricObjectType, 120
- \*Topic **geometry**
  - ISOCellGeometry, 64
  - ISOGeometricObjects, 119
- \*Topic **georeferenceable**
  - ISOGeoreferenceable, 121
- \*Topic **graphic**
  - ISOBrowseGraphic, 62
- \*Topic **gridded**
  - ISOGriddedDataPositionalAccuracy, 122
- \*Topic **grid**
  - ISOGeoreferenceable, 121
  - ISOGridSpatialRepresentation, 124
- \*Topic **group**
  - GMLOperationParameterGroup, 33
- \*Topic **hierarchyLevel**
  - ISOHierarchyLevel, 125
  - ISOMaintenanceFrequency, 138
- \*Topic **identification**
  - ISODataIdentification, 79
  - ISOIdentification, 126
  - ISOServiceIdentification, 177
- \*Topic **identifier**
  - ISOIdentifier, 128
  - ISOMetaIdentifier, 151
  - ISOReferenceIdentifier, 166
  - ISOResponsibleParty, 169
  - ISOTelephone, 188
- \*Topic **information**
  - ISOContentInformation, 74
  - ISOExtendedElementInformation, 101
  - ISOMaintenanceInformation, 139
  - ISOMetadataExtensionInformation, 148
- \*Topic **integer**
  - ISOBaseInteger, 57
  - ISOUnlimitedInteger, 193
- \*Topic **keywords**
  - ISOKeywords, 130
- \*Topic **keywordtype**
  - ISOKeywordType, 131
- \*Topic **language**
  - ISOLanguage, 132
- \*Topic **legal**
  - ISOLegalConstraints, 133
- \*Topic **length**
  - ISODistance, 95
  - ISOLength, 134
- \*Topic **level**
  - ISOTopologyLevel, 191
- \*Topic **lineage**
  - ISOLineage, 135
- \*Topic **linear**
  - GMLLinearCS, 29
- \*Topic **listed**
  - ISOListedValue, 136
- \*Topic **local**
  - ISOLocalName, 138
- \*Topic **maintenance**
  - ISOMaintenanceInformation, 139
- \*Topic **measure**
  - ISOAngle, 49
  - ISODistance, 95
  - ISOLength, 134
  - ISOMeasure, 140
  - ISOScale, 173
- \*Topic **metadata**

- ISOAbstractObject, 45
- ISOMetadata, 142
- ISOMetadataExtensionInformation, 148
- ISOMetadataNamespace, 150
- \*Topic **meta**
  - ISOMetaIdentifier, 151
- \*Topic **method**
  - GMLOperationMethod, 31
- \*Topic **mime**
  - ISOMimeFileType, 152
- \*Topic **multiplicity**
  - ISOMultiplicity, 153
  - ISOMultiplicityRange, 154
- \*Topic **namespace**
  - ISOMetadataNamespace, 150
- \*Topic **name**
  - ISOAbstractGenericName, 44
  - ISOFileName, 114
  - ISOLocalName, 138
  - ISOScopedName, 175
- \*Topic **objects**
  - ISOGeometricObjects, 119
- \*Topic **object**
  - GMLAbstractTimeObject, 16
  - ISOGeometricObjectType, 120
- \*Topic **online**
  - ISOOnlineResource, 156
- \*Topic **operation**
  - GMLAbstractGeneralOperationParameter, 11
  - GMLOperationMethod, 31
  - GMLOperationParameter, 32
  - GMLOperationParameterGroup, 33
  - ISOFeatureAttribute, 106
  - ISOFeatureOperation, 111
- \*Topic **parameter**
  - GMLAbstractGeneralOperationParameter, 11
  - GMLOperationParameter, 32
  - GMLOperationParameterGroup, 33
  - ISOParameter, 158
  - ISOParameterDirection, 159
- \*Topic **period**
  - GMLTimePeriod, 37
  - ISOTimePeriod, 190
- \*Topic **polar**
  - GMLPolarCS, 34
- \*Topic **positional**
  - ISOAbsoluteExternalPositionalAccuracy, 41
  - ISOAbstractPositionalAccuracy, 46
  - ISOGriddedDataPositionalAccuracy, 122
- \*Topic **presentation**
  - ISOPresentationForm, 160
- \*Topic **primitive**
  - GMLAbstractTimeGeometricPrimitive, 15
  - GMLAbstractTimePrimitive, 17
- \*Topic **process**
  - ISOProcessStep, 161
- \*Topic **projected**
  - GMLProjectedCRS, 34
- \*Topic **property**
  - ISOPropertyType, 162
- \*Topic **quality**
  - ISOAbsoluteExternalPositionalAccuracy, 41
  - ISOAbstractPositionalAccuracy, 46
  - ISODataQuality, 83
  - ISODataQualityAbstractElement, 85
  - ISOGriddedDataPositionalAccuracy, 122
- \*Topic **range**
  - ISOMultiplicityRange, 154
  - ISORangeDimension, 163
- \*Topic **real**
  - ISOBaseReal, 58
- \*Topic **record**
  - ISOElementSequence, 99
  - ISOMemberName, 141
  - ISORecord, 164
  - ISORecordType, 165
- \*Topic **reference**
  - ISODefinitionReference, 90
  - ISOReferenceIdentifier, 166
  - ISOReferenceSystem, 167
- \*Topic **representation**
  - ISOGeoreferenceable, 121
  - ISOGridSpatialRepresentation, 124
  - ISOSpatialRepresentationType, 184
- \*Topic **representative**
  - ISORepresentativeFraction, 168
- \*Topic **resource**
  - ISOOnlineResource, 156

- \*Topic **result**
  - ISOConformanceResult, 70
- \*Topic **roleType**
  - ISORoleType, 172
- \*Topic **role**
  - ISOAssociationRole, 50
  - ISORole, 171
- \*Topic **scale**
  - ISOScale, 173
- \*Topic **scoped**
  - ISOScopedName, 175
- \*Topic **security**
  - ISOSecurityConstraints, 175
- \*Topic **service**
  - ISOServiceIdentification, 177
- \*Topic **single**
  - GMLAbstractGeneralDerivedCRS, 10
  - GMLAbstractSingleCRS, 14
  - GMLAbstractSingleOperation, 14
- \*Topic **source**
  - ISODefinitionSource, 91
  - ISOSource, 182
- \*Topic **spatialtemporal**
  - ISOSpatialTemporalExtent, 185
- \*Topic **spatial**
  - ISOGeoreferenceable, 121
  - ISOGridSpatialRepresentation, 124
  - ISOSpatialRepresentationType, 184
- \*Topic **spherical**
  - GMLSphericalCS, 35
- \*Topic **status**
  - ISOStatus, 187
- \*Topic **step**
  - ISOProcessStep, 161
- \*Topic **string**
  - ISOBaseCharacterString, 53
- \*Topic **system**
  - GMLAbstractCoordinateSystem, 7
  - GMLAffineCS, 18
  - GMLCartesianCS, 19
  - GMLCoordinateSystemAxis, 22
  - GMLCylindricalCS, 23
  - GMLEllipsoidalCS, 28
  - GMLLinearCS, 29
  - GMLObliqueCartesianCS, 30
  - GMLPolarCS, 34
  - GMLSphericalCS, 35
  - GMLTemporalCS, 36
  - GMLTimeCS, 37
  - GMLUserDefinedCS, 40
  - GMLVerticalCS, 40
  - ISOReferenceSystem, 167
- \*Topic **temporal**
  - GMLAbstractTimeGeometricPrimitive, 15
  - GMLTemporalCS, 36
  - ISOTemporalExtent, 189
- \*Topic **time**
  - GMLAbstractTimeObject, 16
  - GMLAbstractTimePrimitive, 17
  - GMLTimeCS, 37
  - GMLTimePeriod, 37
  - ISOTimePeriod, 190
- \*Topic **topic**
  - ISOTopicCategory, 190
- \*Topic **topology**
  - ISOTopologyLevel, 191
- \*Topic **typename**
  - ISOTypeName, 192
- \*Topic **type**
  - ISOCouplingType, 76
  - ISOFeatureType, 112
  - ISOGeometricObjectType, 120
  - ISOMimeFileType, 152
  - ISOPropertyType, 162
  - ISOSpatialRepresentationType, 184
- \*Topic **unit**
  - GMLBaseUnit, 18
  - GMLConventionalUnit, 20
  - GMLDerivedUnit, 26
  - GMLUnitDefinition, 39
- \*Topic **unlimited**
  - ISOUnlimitedInteger, 193
- \*Topic **value**
  - ISOListedValue, 136
- \*Topic **vertical**
  - GMLVerticalCS, 40
  - ISOVerticalExtent, 196
- fetchISOCodelists, 5
- geometa, 6
- geometa-package (geometa), 6
- GMLAbstractCoordinateOperation, 6
- GMLAbstractCoordinateSystem, 7
- GMLAbstractCRS, 8
- GMLAbstractGeneralConversion, 9

- GMLAbstractGeneralDerivedCRS, [10](#)
- GMLAbstractGeneralOperationParameter, [11](#)
- GMLAbstractGML, [12](#)
- GMLAbstractObject, [13](#)
- GMLAbstractSingleCRS, [14](#)
- GMLAbstractSingleOperation, [14](#)
- GMLAbstractTimeGeometricPrimitive, [15](#)
- GMLAbstractTimeObject, [16](#)
- GMLAbstractTimePrimitive, [17](#)
- GMLAffineCS, [18](#)
- GMLBaseUnit, [18](#)
- GMLCartesianCS, [19](#)
- GMLConventionalUnit, [20](#)
- GMLConversion, [21](#)
- GMLCoordinateSystemAxis, [22](#)
- GMLCylindricalCS, [23](#)
- GMLDefinition, [24](#)
- GMLDerivedCRS, [25](#)
- GMLDerivedUnit, [26](#)
- GMLElement, [27](#)
- GMLEllipsoidalCS, [28](#)
- GMLGeodeticCRS, [28](#)
- GMLLinearCS, [29](#)
- GMLObliqueCartesianCS, [30](#)
- GMLOperationMethod, [31](#)
- GMLOperationParameter, [32](#)
- GMLOperationParameterGroup, [33](#)
- GMLPolarCS, [34](#)
- GMLProjectedCRS, [34](#)
- GMLSphericalCS, [35](#)
- GMLTemporalCS, [36](#)
- GMLTimeCS, [37](#)
- GMLTimePeriod, [37](#)
- GMLUnitDefinition, [39](#)
- GMLUserDefinedCS, [40](#)
- GMLVerticalCS, [40](#)
  
- ISOAbsoluteExternalPositionalAccuracy, [41](#)
- ISOAbstractCatalogue, [42](#)
- ISOAbstractGenericName, [44](#)
- ISOAbstractObject, [45](#)
- ISOAbstractPositionalAccuracy, [46](#)
- ISOAddress, [47](#)
- ISOAnchor, [48](#)
- ISOAngle, [49](#)
- ISOAssociationRole, [50](#)
- ISOBand, [51](#)
- ISOBaseBoolean, [52](#)
- ISOBaseCharacterString, [53](#)
- ISOBaseDate, [54](#)
- ISOBaseDateTime, [55](#)
- ISOBaseDecimal, [56](#)
- ISOBaseInteger, [57](#)
- ISOBaseReal, [58](#)
- ISOBinding, [59](#)
- ISOBoundAssociationRole, [60](#)
- ISOBoundFeatureAttribute, [61](#)
- ISOBrowseGraphic, [62](#)
- ISOCarrierOfCharacteristics, [63](#)
- ISOCellGeometry, [64](#)
- ISOCharacterSet, [65](#)
- ISOCitation, [66](#)
- ISOClassification, [67](#)
- ISOCodelist, [68](#)
- ISOCodelistValue, [69](#)
- ISOConformanceResult, [70](#)
- ISOConstraint, [71](#)
- ISOConstraints, [72](#)
- ISOContact, [73](#)
- ISOContentInformation, [74](#)
- ISOCoupledResource, [75](#)
- ISOCouplingType, [76](#)
- ISOCoverageContentType, [77](#)
- ISOCoverageDescription, [78](#)
- ISODataIdentification, [79](#)
- ISODataQuality, [83](#)
- ISODataQualityAbstractElement, [85](#)
- ISODatatype, [86](#)
- ISODate, [87](#)
- ISODataType, [88](#)
- ISODCPList, [89](#)
- ISODefinitionReference, [90](#)
- ISODefinitionSource, [91](#)
- ISODigitalTransferOptions, [92](#)
- ISODimension, [93](#)
- ISODimensionNameType, [94](#)
- ISODistance, [95](#)
- ISODistribution, [96](#)
- ISODistributor, [97](#)
- ISODomainConsistency, [98](#)
- ISOElementSequence, [99](#)
- ISOEvaluationMethodType, [100](#)
- ISOExtendedElementInformation, [101](#)
- ISOExtent, [103](#)
- ISOFeatureAssociation, [105](#)

- ISOFeatureAttribute, 106
- ISOFeatureCatalogue, 107
- ISOFeatureCatalogueDescription, 110
- ISOFeatureOperation, 111
- ISOFeatureType, 112
- ISOFileName, 114
- ISOFormat, 115
- ISOGeographicBoundingBox, 116
- ISOGeographicDescription, 117
- ISOGeographicExtent, 118
- ISOGeometricObjects, 119
- ISOGeometricObjectType, 120
- ISOGeoreferenceable, 121
- ISOGriddedDataPositionalAccuracy, 122
- ISOGridSpatialRepresentation, 124
- ISOHierarchyLevel, 125
- ISOIdentification, 126
- ISOIdentifier, 128
- ISOInheritanceRelation, 129
- ISOKeywords, 130
- ISOKeywordType, 131
- ISOLanguage, 132
- ISOLegalConstraints, 133
- ISOLength, 134
- ISOLineage, 135
- ISOListedValue, 136
- ISOLocalName, 138
- ISOMaintenanceFrequency, 138
- ISOMaintenanceInformation, 139
- ISOMeasure, 140
- ISOMemberName, 141
- ISOMetadata, 142
- ISOMetadataExtensionInformation, 148
- ISOMetadataNamespace, 150
- ISOMetaIdentifier, 151
- ISOMimeFileType, 152
- ISOMultiplicity, 153
- ISOMultiplicityRange, 154
- ISOobligation, 155
- ISOonlineResource, 156
- ISOoperationMetadata, 157
- ISOParameter, 158
- ISOParameterDirection, 159
- ISOPresentationForm, 160
- ISOProcessStep, 161
- ISOPropertyType, 162
- ISORangeDimension, 163
- ISORecord, 164
- ISORecordType, 165
- ISOReferenceIdentifier, 166
- ISOReferenceSystem, 167
- ISORepresentativeFraction, 168
- ISOResponsibleParty, 169
- ISORestriction, 170
- ISORole, 171
- ISORoleType, 172
- ISOScale, 173
- ISOScope, 174
- ISOScopedName, 175
- ISOSecurityConstraints, 175
- ISOServiceIdentification, 177
- ISOSource, 182
- ISOspatialRepresentation, 183
- ISOspatialRepresentationType, 184
- ISOspatialTemporalExtent, 185
- ISOStandardOrderProcess, 186
- ISOStatus, 187
- ISOTelephone, 188
- ISOtemporalExtent, 189
- ISOTimePeriod, 190
- ISOTopicCategory, 190
- ISOTopologyLevel, 191
- ISOtypeName, 192
- ISOUnlimitedInteger, 193
- ISOURL, 194
- ISOVectorSpatialRepresentation, 195
- ISOVerticalExtent, 196
  
- R6Class, 6–41, 43–45, 47–66, 68–79, 84–97, 99–101, 104–107, 110–112, 114–121, 123–126, 128–135, 137–142, 149–177, 182, 184–196