

Package ‘geometa’

August 4, 2020

Type Package

Title Tools for Reading and Writing ISO/OGC Geographic Metadata

Version 0.6-3

Date 2020-08-04

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 (>= 3.3.0)

Imports methods, R6, XML, httr, jsonlite

Suggests sf, ncd4, EML, emld, units, udunits2, testthat, roxygen2

License MIT + file LICENSE

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

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

LazyLoad yes

RoxygenNote 7.1.0

NeedsCompilation no

Author Emmanuel Blondel [aut, cre] (<<https://orcid.org/0000-0002-5870-5762>>)

Repository CRAN

Date/Publication 2020-08-04 12:40:06 UTC

R topics documented:

cacheISOClasses	9
convert_metadata	9
geometa	10
geometaLogger	11
geometa_coverage	11

getGeometaOption	12
getISOClasses	13
getISOCodelist	13
getISOCodelists	14
getISOInternalCodelists	14
getISOMetadataNamespace	15
getISOMetadataNamespaces	15
getISOMetadataSchemas	16
getMappingFormats	16
getMappings	17
GMLAbstractCoordinateOperation	17
GMLAbstractCoordinateSystem	18
GMLAbstractCoverage	19
GMLAbstractCRS	20
GMLAbstractCurve	20
GMLAbstractDiscreteCoverage	21
GMLAbstractFeature	22
GMLAbstractGeneralConversion	23
GMLAbstractGeneralDerivedCRS	23
GMLAbstractGeneralOperationParameter	24
GMLAbstractGeneralParameterValue	25
GMLAbstractGeometricAggregate	26
GMLAbstractGeometricPrimitive	26
GMLAbstractGeometry	27
GMLAbstractGML	28
GMLAbstractImplicitGeometry	29
GMLAbstractObject	29
GMLAbstractReferenceableGrid	30
GMLAbstractRing	31
GMLAbstractSingleCRS	32
GMLAbstractSingleOperation	32
GMLAbstractSurface	33
GMLAbstractTimeGeometricPrimitive	34
GMLAbstractTimeObject	34
GMLAbstractTimePrimitive	35
GMLAffineCS	36
GMLBaseUnit	37
GMLCartesianCS	38
GMLCompoundCRS	38
GMLConventionalUnit	39
GMLConversion	40
GMLCoordinateSystemAxis	41
GMLCOVAbstractCoverage	42
GMLCOVExtension	43
GMLCylindricalCS	43
GMLDefinition	44
GMLDerivedCRS	45
GMLDerivedUnit	46

GMLElement	47
GMLEllipsoidalCS	47
GMLEnvelope	48
GMLEnvelopeWithTimePeriod	49
GMLGeneralGridAxis	50
GMLGeodeticCRS	51
GMLGrid	52
GMLGridCoverage	53
GMLGridFunction	53
GMLLinearCS	54
GMLLinearRing	55
GMLLineString	56
GMLMultiCurve	57
GMLMultiCurveCoverage	58
GMLMultiPoint	58
GMLMultiPointCoverage	59
GMLMultiSolidCoverage	60
GMLMultiSurface	61
GMLMultiSurfaceCoverage	62
GMLObliqueCartesianCS	63
GMLOperationMethod	63
GMLOperationParameter	64
GMLOperationParameterGroup	65
GMLParameterValue	66
GMLParameterValueGroup	67
GMLPoint	68
GMLPolarCS	69
GMLPolygon	70
GMLProjectedCRS	71
GMLRectifiedGrid	72
GMLRectifiedGridCoverage	73
GMLReferenceableGridByArray	74
GMLReferenceableGridByTransformation	74
GMLReferenceableGridByVectors	75
GMLSphericalCS	76
GMLTemporalCRS	77
GMLTemporalCS	77
GMLTimeCS	78
GMLTimeInstant	79
GMLTimePeriod	79
GMLUnitDefinition	80
GMLUserDefinedCS	81
GMLVerticalCRS	82
GMLVerticalCS	83
INSPIREMetadataValidator	84
ISOAbsoluteExternalPositionalAccuracy	85
ISOAbstractAggregate	86
ISOAbstractCarrierOfCharacteristics	87

ISOAbstractCatalogue	88
ISOAbstractCompleteness	89
ISOAbstractGenericName	90
ISOAbstractLogicalConsistency	90
ISOAbstractObject	91
ISOAbstractPositionalAccuracy	93
ISOAbstractPropertyType	94
ISOAbstractReferenceSystem	95
ISOAbstractResult	96
ISOAbstractTemporalAccuracy	96
ISOAbstractThematicAccuracy	97
ISOAccuracyOfATimeMeasurement	98
ISOAddress	99
ISOAggregateInformation	100
ISOAnchor	102
ISOAngle	103
ISOApplicationSchemaInformation	103
ISOAssociation	104
ISOAssociationRole	105
ISOAssociationType	106
ISOAttributes	107
ISOBand	107
ISOBaseBoolean	109
ISOBaseCharacterString	110
ISOBaseDate	111
ISOBaseDateTime	111
ISOBaseDecimal	112
ISOBaseInteger	113
ISOBaseReal	114
ISOBinary	114
ISOBinding	115
ISOBoundAssociationRole	116
ISOBoundFeatureAttribute	117
ISOBoundingPolygon	117
ISOBrowseGraphic	118
ISOCarrierOfCharacteristics	119
ISOCellGeometry	120
ISOCharacterSet	121
ISOCitation	121
ISOCitationSeries	124
ISOClassification	125
ISOCodelist	125
ISOCodelistValue	126
ISOCompletenessCommission	127
ISOCompletenessOmission	128
ISOConceptualConsistency	129
ISOConformanceResult	131
ISOConstraint	132

ISOConstraints	133
ISOContact	134
ISOContentInformation	135
ISOCountry	136
ISOCoupledResource	136
ISOCouplingType	137
ISOCoverageContentType	138
ISOCoverageDescription	139
ISODataFile	140
ISODataIdentification	141
ISODataQuality	146
ISODataQualityAbstractElement	148
ISODataset	149
ISODatatype	150
ISODate	151
ISODateType	152
ISODCPList	153
ISODefinitionReference	153
ISODefinitionSource	154
ISODigitalTransferOptions	155
ISODimension	156
ISODimensionNameType	157
ISODistance	158
ISODistribution	158
ISODistributionUnits	160
ISODistributor	160
ISODomainConsistency	162
ISOElementSequence	163
ISOEvaluationMethodType	164
ISOExtendedElementInformation	165
ISOExtent	167
ISOFeatureAssociation	168
ISOFeatureAttribute	169
ISOFeatureCatalogue	170
ISOFeatureCatalogueDescription	173
ISOFeatureOperation	174
ISOFeatureType	175
ISOFileName	177
ISOFormat	178
ISOFormatConsistency	180
ISOFreeText	181
ISOGeographicBoundingBox	182
ISOGeographicDescription	183
ISOGeographicExtent	184
ISOGeometricObjects	184
ISOGeometricObjectType	185
ISOGeorectified	186
ISOGeoreferenceable	187

ISOGriddedDataPositionalAccuracy	189
ISOGridSpatialRepresentation	190
ISOHierarchyLevel	192
ISOIdentification	192
ISOIdentifier	194
ISOImageDescription	195
ISOImageryAbstractGeolocationInformation	198
ISOImageryAcquisitionInformation	198
ISOImageryAlgorithm	200
ISOImageryBand	201
ISOImageryBandDefinition	203
ISOImageryContext	204
ISOImageryCoverageDescription	205
ISOImageryCoverageResult	206
ISOImageryEnvironmentalRecord	207
ISOImageryEvent	208
ISOImageryGCP	210
ISOImageryGCPCollection	211
ISOImageryGeometryType	212
ISOImageryGeorectified	213
ISOImageryGeoreferenceable	214
ISOImageryImageDescription	215
ISOImageryInstrument	216
ISOImageryMetadata	218
ISOImageryNominalResolution	225
ISOImageryObjective	227
ISOImageryObjectiveType	229
ISOImageryOperation	230
ISOImageryOperationType	231
ISOImageryPlan	232
ISOImageryPlatform	234
ISOImageryPlatformPass	236
ISOImageryPolarisationOrientation	237
ISOImageryPriority	238
ISOImageryProcessing	239
ISOImageryProcessStep	241
ISOImageryProcessStepReport	243
ISOImageryRangeElementDescription	244
ISOImageryRequestedDate	245
ISOImageryRequirement	246
ISOImagerySensorType	249
ISOImagerySequence	250
ISOImagerySource	251
ISOImageryTransferFunctionType	252
ISOImageryTrigger	253
ISOImageryUsability	254
ISOImagingCondition	254
ISOInheritanceRelation	255

ISOInitiative	256
ISOInitiativeType	257
ISOKeywords	258
ISOKeywordType	259
ISOLanguage	260
ISOLegalConstraints	261
ISOLength	263
ISOLineage	263
ISOListedValue	265
ISOLocale	266
ISOLocaleContainer	267
ISOLocalisedCharacterString	268
ISOLocalName	269
ISOMaintenanceFrequency	269
ISOMaintenanceInformation	270
ISOMeasure	271
ISOMedium	272
ISOMediumFormat	273
ISOMediumName	274
ISOMemberName	274
ISOMetadata	275
ISOMetadataExtensionInformation	283
ISOMetadataNamespace	285
ISOMetaIdentifier	286
ISOMimeFileType	286
ISOMultiplicity	287
ISOMultiplicityRange	288
ISONonQuantitativeAttributeAccuracy	289
ISOObligation	290
ISOOnLineFunction	291
ISOOnlineResource	292
ISOOperationMetadata	293
ISOOtherAggregate	294
ISOParameter	295
ISOParameterDirection	296
ISOPixelOrientation	297
ISOPlatform	298
ISOPortrayalCatalogueReference	299
ISOPresentationForm	300
ISOProcessStep	301
ISOProductionSeries	302
ISOPropertyType	303
ISOQuantitativeAttributeAccuracy	304
ISOQuantitativeResult	305
ISORangeDimension	306
ISORecord	307
ISORecordType	308
ISOReferenceIdentifier	309

ISOReferenceSystem	310
ISORepresentativeFraction	311
ISOResolution	312
ISOResponsibleParty	313
ISORestriction	314
ISORole	315
ISORoleType	316
ISOScale	317
ISOScope	317
ISOScopeDescription	318
ISOScopedName	319
ISOSecurityConstraints	320
ISOSensor	321
ISOSeries	322
ISOServiceIdentification	323
ISOSource	325
ISOSpatialRepresentation	327
ISOSpatialRepresentationType	328
ISOSpatialTemporalExtent	329
ISOSRVServiceIdentification	330
ISOStandardOrderProcess	335
ISOStatus	337
ISOStereoMate	337
ISOTelephone	338
ISOTemporalConsistency	339
ISOTemporalExtent	340
ISOTemporalValidity	341
ISOThematicClassificationCorrectness	342
ISOTimePeriod	343
ISOTopicCategory	344
ISOTopologicalConsistency	345
ISOTopologyLevel	346
ISOTypeName	347
ISOUnlimitedInteger	348
ISOURL	348
ISOUsage	349
ISOVectorSpatialRepresentation	350
ISOVerticalExtent	351
pivot_format	352
readISO19139	353
registerISOCodelist	354
registerISOMetadataNamespace	354
registerISOMetadataSchema	355
registerMappingFormat	356
registerMappings	356
setGeometaOption	357
setISOCodelists	357
setISOMetadataNamespaces	358

setISOMetadataSchemas	358
setMappingFormats	358

Index	359
--------------	------------

cacheISOClasses	<i>cacheISOClasses</i>
-----------------	------------------------

Description

[cacheISOClasses](#) allows to cache the list of **geometa** classes or extended. This is especially required to fasten the decoding of metadata elements from an XML file. It is called internally by **geometa** the first function [getISOClasses](#) is called and each time the function [readISO19139](#) function is called to integrate eventually new classes added by user to extend **geometa** model (case of ISO profiles).

Usage

```
cacheISOClasses()
```

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
cacheISOClasses()
```

convert_metadata	<i>convert_metadata</i>
------------------	-------------------------

Description

`convert_metadata` is a tentative generic metadata converter to convert from one source object, represented in a source metadata object model in R (eg eml) to a target metadata object, represented in another target metadata object model (eg **geometa ISOMetadata**). This function relies on a list of mapping rules defined to operate from the source metadata object to the target metadata object. This list of mapping rules is provided in a tabular format. A version is embedded in **geometa** and can be returned with [getMappings](#).

Usage

```
convert_metadata(obj, from, to, mappings, verbose)
```

Arguments

obj	a metadata object given in one of the mapping formats known by geometa . The object should be a valid id as listed by getMappingFormats , supported as source format (from is TRUE).
from	a valid mapping format id (see getMappingFormats) that indicates the metadata model / format used for the argument obj
to	a valid mapping format id (see getMappingFormats) to convert to
mappings	a <code>data.frame</code> giving the reference mapping rules to convert metadata object. This <code>data.frame</code> is by default the output of getMappings .
verbose	print debugging messages. Default is FALSE

Value

an metadata object in the model specified as to argument

Note

This function is mainly used internally in as generic methods to convert from one metadata format to another. It is exported for extension to user custom metadata formats or for debugging purpose. This converter is still experimental.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

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:	0.6-3
Date:	2020-08-04
License:	MIT
LazyLoad:	yes

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

geometaLogger *geometaLogger*

Description

geometaLogger

Format

R6Class object.

Value

Object of R6Class for modelling a simple logger

Abstract Methods

INFO(text) Logger to report information. Used internally

WARN(text) Logger to report warnings. Used internally

ERROR(text) Logger to report errors. Used internally

Note

Logger class used internally by geometa

geometa_coverage *geometa_coverage*

Description

geometa_coverage is a function to report coverage of ISO/OGC standard classes in package **geometa**. The function will inspect all classes of the ISO/OGC standards and will scan if **geometa** supports it.

Usage

```
geometa_coverage()
```

Value

an object of class data.frame

Note

This function is used as Quality Assurance indicator to assess the percentage of completeness of ISO/OGC standards in **geometa**.

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
cov <- geometa_coverage()
```

<code>getGeometaOption</code>	<i>getGeometaOption</i>
-------------------------------	-------------------------

Description

`getGeometaOption` allows to get an option from **geometa**

Usage

```
getGeometaOption(option)
```

Arguments

`option` the name of the option

Value

the option

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getGeometaOption("schemaBaseUrl")
```

getISOClasses	<i>getISOClasses</i>
---------------	----------------------

Description

get the list of cached ISO classes

Usage

```
getISOClasses()
```

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getISOClasses()
```

getISOCodelist	<i>getISOCodelist</i>
----------------	-----------------------

Description

getISOCodelist allows to get a registered ISO codelist by id registered in **geometa**

Usage

```
getISOCodelist(id)
```

Arguments

id identifier of the codelist

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getISOCodelist(id = "LanguageCode")
```

`getISOCodelists` *getISOCodelists*

Description

`getISOCodelists` allows to get the list of ISO codelists registered in **geometa**, their description and XML definition. The object returned is of class "data.frame"

Usage

```
getISOCodelists()
```

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getISOCodelists()
```

`getISOInternalCodelists`
getISOInternalCodelists

Description

`getISOInternalCodelists` allows to get the list of ISO codelists registered in **geometa**

Usage

```
getISOInternalCodelists()
```

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getISOInternalCodelists()
```

`getISOMetadataNamespace`
getISOMetadataNamespace

Description

`getISOMetadataNamespace` gets a namespace given its id

Usage

`getISOMetadataNamespace(id)`

Arguments

id namespace prefix

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

`getISOMetadataNamespace("GMD")`

`getISOMetadataNamespaces`
getISOMetadataNamespaces

Description

`getISOMetadataNamespaces` gets the list of namespaces registered

Usage

`getISOMetadataNamespaces()`

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

`getISOMetadataNamespaces()`

`getISOMetadataSchemas` *getISOMetadataSchemas*

Description

`getISOMetadataSchemas` gets the schemas registered in **geometa**

Usage

```
getISOMetadataSchemas()
```

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
getISOMetadataSchemas()
```

`getMappingFormats` *getMappingFormats*

Description

`getMappingFormats` gets the mapping formats registered in **geometa**

Usage

```
getMappingFormats(pretty)
```

Arguments

`pretty` by default TRUE to return the list of formats as `data.frame`. Set to FALSE to return a list of `pivot_format` objects

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

`getMappings`*getMappings*

Description

List the mappings rules to convert from/to other metadata formats (currently EML/emld objects and NetCDF-CF/ncdf4 objects)

Usage`getMappings()`**Value**

a `data.frame` containing the metadata mapping rules

`GMLAbstractCoordinateOperation`*GMLAbstractCoordinateOperation*

Description

`GMLAbstractCoordinateOperation`

Format

[R6Class](#) object.

Value

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

Fields

`operationVersion` [[GMLElement](#)]

`coordinateOperationAccuracy` [[GMLElement](#)]

`sourceCRS` [[GMLElement](#)]

`targetCRS` [[GMLElement](#)]

Methods

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS
 setDomainOfValidity(domainOfValidity) Sets the domain of validity
 addScope(scope) Adds a scope
 delScope(scope) Deletes a scope
 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
 OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractCoordinateSystem

GMLAbstractCoordinateSystem

Description

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractCoverage *GMLAbstractCoverage*

Description

GMLAbstractCoverage

Format

[R6Class](#) object.

Value

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

Methods

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

`setDomainSet(domainSet)` Set a domain set. The object should be a GML Geometric or Time object

`setRangeSet(rangeSet)` Set a range set. Currently not implemented in **geometa**

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

GMLAbstractCRS

GMLAbstractCRS

Description

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t

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

GMLAbstractCurve

GMLAbstractCurve

Description

GMLAbstractCurve

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, sfg)` This method is used to instantiate a GML abstract curve

Note

Experimental

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

GMLAbstractDiscreteCoverage

GMLAbstractDiscreteCoverage

Description

GMLAbstractDiscreteCoverage

Format

[R6Class](#) object.

Value

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

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class `GMLGridFunction` or `GMLCoverageMappingRule`

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

GMLAbstractFeature	<i>GMLAbstractFeature</i>
--------------------	---------------------------

Description

GMLAbstractFeature

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, element, attrs, defaults)` This method is used to instantiate a GML abstract feature
`setBoundedBy(envelope)` Set the boundedBy as object of class GMLEnvelope

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

GMLAbstractGeneralConversion
GMLAbstractGeneralConversion

Description

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractGeneralDerivedCRS
GMLAbstractGeneralDerivedCRS

Description

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractGeneralOperationParameter

GMLAbstractGeneralOperationParameter

Description

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

GMLAbstractGeneralParameterValue

GMLAbstractGeneralParameterValue

Description

GMLAbstractGeneralParameterValue

Format

[R6Class](#) object.

Value

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

Methods

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

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

GMLAbstractGeometricAggregate

GMLAbstractGeometricAggregate

Description

GMLAbstractGeometricAggregate

Format[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an GML abstract Geometric Aggregate**Methods**`new(xml, element, attrs, defaults)` This method is used to instantiate a GML abstract Geometric aggregate**Note**Class used internally by `geometa`**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
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractGeometricPrimitive

GMLAbstractGeometricPrimitive

Description

GMLAbstractGeometricPrimitive

Format[R6Class](#) object.

Value

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

Methods

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

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

GMLAbstractGeometry *GMLAbstractGeometry*

Description

GMLAbstractGeometry

Format

[R6Class](#) object.

Value

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

Methods

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

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

GMLAbstractGML

GMLAbstractGML

Description

GMLAbstractGML

Format

[R6Class](#) object.

Value

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

Fields

descriptionReference [[GMLElement](#)]

identifier [[GMLElement](#)]

name [[GMLElement](#)]

Methods

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

setDescription(description) Set the description

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLAbstractImplicitGeometry
GMLAbstractImplicitGeometry

Description

GMLAbstractImplicitGeometry

Format

[R6Class](#) object.

Value

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

Methods

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

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

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

GMLAbstractObject *GMLAbstractObject*

Description

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

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

GMLAbstractReferenceableGrid

GMLAbstractReferenceableGrid

Description

GMLAbstractReferenceableGrid

Format

[R6Class](#) object.

Value

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

Methods

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

Note

Class used internally by geometa

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

- OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>
OGC GML 3.3 Schema. <http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd>

GMLAbstractRing	<i>GMLAbstractRing</i>
-----------------	------------------------

Description

GMLAbstractRing

Format

[R6Class](#) object.

Value

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

Methods

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

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

GMLAbstractSingleCRS *GMLAbstractSingleCRS*

Description

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

GMLAbstractSingleOperation
GMLAbstractSingleOperation

Description

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t

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

GMLAbstractSurface *GMLAbstractSurface*

Description

GMLAbstractSurface

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, sfg)` This method is used to instantiate a GML abstract surface

Note

Experimental

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

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

GMLAbstractTimeGeometricPrimitive

GMLAbstractTimeGeometricPrimitive

Description

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 ISOAbstractTimeGeometricPrimitive

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

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

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

GMLAbstractTimePrimitive

GMLAbstractTimePrimitive

Description

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

GMLAffineCS

GMLAffineCS

Description

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 <emmanuel.blondel1@gmail.com>

References

- ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
- OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLBaseUnit

GMLBaseUnit

Description

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
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

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t

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

GMLCompoundCRS

GMLCompoundCRS

Description

GMLCompoundCRS

Format

[R6Class](#) object.

Value

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

Fields

componentReferenceSystem

Methods

new(xml, defaults, id) This method is used to instantiate a GML Abstract CRS
addComponentReferenceSystem(referenceSystem) Adds a reference system
delComponentReferenceSystem(referenceSystem) Deletes a reference 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
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLConventionalUnit *GMLConventionalUnit*

Description

GMLConventionalUnit

Format

[R6Class](#) object.

Value

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

Fields

conversionToPreferredUnit [[GMLElement](#)]
roughConversionToPreferredUnit [[GMLElement](#)]
derivationUnitTerm [[GMLElement](#)]

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. 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$setIdentifier("identifiant", "codespace")
gml$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

Format

[R6Class](#) object.

Value

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

Fields

method [[GMLElement](#)]

parameterValue [list of [GMLAbstractGeneralParameterValue](#)]

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLCoordinateSystemAxis

GMLCoordinateSystemAxis

Description

GMLCoordinateSystemAxis

Format

[R6Class](#) object.

Value

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

Fields

axisAbbrev [[GMLElement](#)]
axisDirection [[GMLElement](#)]
minimumValue [[GMLElement](#)]
maximumValue [[GMLElement](#)]
rangeMeaning [[GMLElement](#)]

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLCOVAbstractCoverage

GMLCOVAbstractCoverage

Description

GMLCOVAbstractCoverage

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling a GMLCOV Abstract Coverage

Methods

`new(xml, element, attrs, defaults, wrap)` This method is used to create an GMLCOV Abstract Coverage

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

GML 3.2.1 Application Schema for Coverages <http://www.opengis.net/gmlcov/1.0>

GMLCOVExtension

GMLCOVExtension

Description

GMLCOVExtension

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling a GMLCOV Extension

Methods

`new(xml, element, attrs, defaults, wrap)` This method is used to create a GMLCOV extension

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

GML 3.2.1 Application Schema for Coverages <http://www.opengis.net/gmlcov/1.0>

GMLCylindricalCS

GMLCylindricalCS

Description

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

GMLDefinition

GMLDefinition

Description

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
`addRemark(remark)` Adds a remark
`delRemark(remark)` Deletes a remark

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
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

Format

[R6Class](#) object.

Value

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

Fields

baseCRS [[GMLElement](#)]
derivedCRSType [[GMLElement](#)]
coordinateSystem [[GMLElement](#)]

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLDerivedUnit

GMLDerivedUnit

Description

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. 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

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

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

GMLEnvelope

GMLEnvelope

Description

GMLEnvelope

Format

[R6Class](#) object.

Value

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

Fields

`lowerCorner` [[matrix](#)]

`upperCorner` [[matrix](#)]

Methods

`new(xml, bbox, srsName, srsDimension, axisLabels, uomLabels)` This method is used to instantiate a GML envelope. The argument 'bbox' should be a matrix of dim 2,2 giving the x/y min/max values of a bounding box, as returned by `bbox` function in package **sp**

Note

Experimental

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

GMLEnvelopeWithTimePeriod

GMLEnvelopeWithTimePeriod

Description

GMLEnvelopeWithTimePeriod

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an GML envelope with time period

Fields

beginPosition [[GMLElement](#)]

endPosition [[GMLElement](#)]

Methods

`new(xml, bbox, beginPosition, endPosition, srsName, srsDimension, axisLabels, uomLabels)`

This method is used to instantiate a GML envelope with time period. The `bbox` parameter should be an object of class `matrix` with 2 columns giving min/max values of each dimension (handled by row). At least one row is required (1D), and can be extended with as many dimensions required. The parameters `beginPosition` and `endPosition` allow to specify the temporal extent of the envelope.

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

GMLGeneralGridAxis *GMLGeneralGridAxis*

Description

GMLGeneralGridAxis

Format

[R6Class](#) object.

Value

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

Fields

offsetVector [[GMLElement](#)]
coefficients [[GMLElement](#)]
gridAxesSpanned [[GMLElement](#)]
sequenceRule [[GMLElement](#)]

Methods

new(xml) This method is used to instantiate a GML GeneralGridAxis.

Note

Experimental

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>
OGC GML 3.3 Schema. <http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd>

GMLGeodeticCRS	<i>GMLGeodeticCRS</i>
----------------	-----------------------

Description

GMLGeodeticCRS

Format

[R6Class](#) object.

Value

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

Fields

ellipsoidalCS [[GMLEllipsoidalCS](#)]
cartesianCS [[GMLCartesianCS](#)]
sphericalCS [[GMLSphericalCS](#)]
geodeticDatum [[GMLGeodeticDatum](#)]

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLGrid

GMLGrid

Description

GMLGrid

Format

[R6Class](#) object.

Value

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

Methods

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

`setGridEnvelope(m)` Set the grid envelope limits as object.

`setAxisLabels(xlabel, ylabel)` Set the Axis labels

`addAxisName(axisName)` Adds an axis name

`delAxisName(axisName)` Deletes an axis 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

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

GMLGridCoverage

GMLGridCoverage

Description

GMLGridCoverage

Format

R6Class object.

Value

Object of R6Class for modelling an GML grid coverage

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class GMLGridFunction or GMLCoverageMappingRule

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

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

GMLGridFunction

GMLGridFunction

Description

GMLGridFunction

Format

R6Class object.

Value

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

Methods

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

`setSequenceRule(sequenceRule)` Set the sequence rule, as object of class character

`setStartPoint(x,y)` Set the start point

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

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

GMLLinearCS

GMLLinearCS

Description

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.blondel1@gmail.com>

References

- ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLLinearRing

GMLLinearRing

Description

GMLLinearRing

Format

[R6Class](#) object.

Value

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

Fields

posList

Methods

`new(xml, m)` This method is used to instantiate a GML LinearRing

Note

Experimental

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

GMLLineString	<i>GMLLineString</i>
---------------	----------------------

Description

GMLLineString

Format

[R6Class](#) object.

Value

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

Fields

posList

Methods

`new(xml, sfg)` This method is used to instantiate a GML linestring

Note

Experimental

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

GMLMultiCurve

GMLMultiCurve

Description

GMLMultiCurve

Format

[R6Class](#) object.

Value

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

Fields

curveMember

Methods

`new(xml, sfg)` This method is used to instantiate a GML multicurve

`addCurveMember(curve)` Add a curve member

`delCurveMember(curve)` Deletes a curve member

Note

Experimental

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

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

GMLMultiCurveCoverage *GMLMultiCurveCoverage*

Description

GMLMultiCurveCoverage

Format

[R6Class](#) object.

Value

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

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class `GMLGridFunction` or `GMLCoverageMappingRule`

Note

Class used internally by `geometa`

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

GMLMultiPoint

GMLMultiPoint

Description

GMLMultiPoint

Format

[R6Class](#) object.

Value

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

Fields

pointMember

Methods

new(xml, sfg) This method is used to instantiate a GML multipoint

addPointMember(point) Add a point member

delPointMember(point) Deletes a point member

Note

Experimental

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

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

GMLMultiPointCoverage *GMLMultiPointCoverage*

Description

GMLMultiPointCoverage

Format

[R6Class](#) object.

Value

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

Methods

new(xml, element, attrs, defaults) This method is used to instantiate a GMLmultipoint coverage

setCoverageFunction(coverageFunction) Set a coverage function, object of class GMLGridFunction or GMLCoverageMappingRule

Note

Class used internally by geometa

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

GMLMultiSolidCoverage *GMLMultiSolidCoverage*

Description

GMLMultiSolidCoverage

Format

[R6Class](#) object.

Value

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

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class `GMLGridFunction` or `GMLCoverageMappingRule`

Note

Class used internally by geometa

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

GMLMultiSurface	<i>GMLMultiSurface</i>
-----------------	------------------------

Description

GMLMultiSurface

Format

[R6Class](#) object.

Value

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

Fields

surfaceMember

Methods

`new(xml, sfg)` This method is used to instantiate a GML multisurface

`addSurfaceMember(surface)` Add a surface member

`delSurfaceMember(surface)` Deletes a surface member

Note

Experimental

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

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

GMLMultiSurfaceCoverage

GMLMultiSurfaceCoverage

Description

GMLMultiSurfaceCoverage

Format

[R6Class](#) object.

Value

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

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class `GMLGridFunction` or `GMLCoverageMappingRule`

Note

Class used internally by `geometa`

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

GMLObliqueCartesianCS *GMLObliqueCartesianCS*

Description

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

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

GMLOperationMethod *GMLOperationMethod*

Description

GMLOperationMethod

Format

[R6Class](#) object.

Value

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

Fields

formulaCitation [ISOCitation]
formula [GMLElement]
sourceDimensions [GMLElement]
targetDimensions [GMLElement]
parameter [list of [GMLOperationParameter or GMLOperationParameterGroup]]

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
setFormula(formula) Sets a formula, object of class character
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
OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLOperationParameter *GMLOperationParameter*

Description

GMLOperationParameter

Format

[R6Class](#) object.

Value

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

Inherited methods

from 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

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

GMLOperationParameterGroup

GMLOperationParameterGroup

Description

GMLOperationParameterGroup

Format

[R6Class](#) object.

Value

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

Fields

maximumOccurs [[GMLElement](#)]

parameter [[GMLOperationParameter](#)|[GMLOperationParameterGroup](#)]

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.blondell@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t
 OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLParameterValue	<i>GMLParameterValue</i>
-------------------	--------------------------

Description

GMLParameterValue

Format

[R6Class](#) object.

Value

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

Fields

value [[GMLElement](#)]
 operationParameter [[GMLOperationParameter](#)]

Methods

new(xml, defaults) This method is used to instantiate a GML ParameterValue
 setValue(value, uom) Sets the value (object of class "numeric"), with unit of measure (uom)
 setStringValue(value) Sets a string value
 setIntegerValue(value) Sets an integer value
 setBooleanValue(value) Sets a boolean value
 setValueFile(value) Sets a file value
 setOperationParameter(operationParameter) Sets the operation parameter, object of class
 GMLOperationParameter

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

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

Examples

```
gml <- GMLParameterValue$new()
gml$setValue(1.1, "test")
op <- GMLOperationParameter$new()
op$setDescriptionReference("someref")
op$setIdentifier("identifier", "codespace")
op$addName("name1", "codespace")
op$addName("name2", "codespace")
op$setMinimumOccurs(2L)
gml$setOperationParameter(op)
xml <- gml$encode()
```

GMLParameterValueGroup

GMLParameterValueGroup

Description

GMLParameterValueGroup

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an GML parameter value group

Fields

parameterValue [[GMLParameterValue](#)]

group [[GMLOperationParameterGroup](#)]

Methods

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

`addParameterValue(parameterValue)` Adds a parameter value, object of class 'GMLParameter-
terValue'

`delParameterValue(parameterValue)` Deletes a parameter value, object of class 'GMLParam-
eterValue'

`setOperationParameterGroup(operationParameterGroup)` Sets the operation parameter group,
object of class GMLOperationParameterGroup

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

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

Examples

```
gml <- GMLParameterValueGroup$new()
```

GMLPoint

GMLPoint

Description

GMLPoint

Format

[R6Class](#) object.

Value

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

Fields

pos

Methods

`new(xml, sfg = NULL, m = NULL)` This method is used to instantiate a GML point either using an
object of class `sfg` (from `sf`), or using a matrix object

Note

Experimental

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

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

GMLPolarCS

GMLPolarCS

Description

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

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

GMLPolygon

GMLPoint

Description

GMLPoint

Format

[R6Class](#) object.

Value

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

Fields

pos

Methods

`new(xml, sfg)` This method is used to instantiate a GML point

Note

Experimental

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

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

GMLProjectedCRS	<i>GMLProjectedCRS</i>
-----------------	------------------------

Description

GMLProjectedCRS

Format

[R6Class](#) object.

Value

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

Fields

baseGeodeticCRS [[GMLGeodeticCRS](#)]

cartesianCS [[GMLCartesianCS](#)]

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

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

GMLRectifiedGrid	<i>GMLRectifiedGrid</i>
------------------	-------------------------

Description

GMLRectifiedGrid

Format

[R6Class](#) object.

Value

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

Inherited Methods

`setGridEnvelope(xmin, xmax, ymin, ymax)` Set the grid envelope limits with xmin,xmax,ymin and ymax.

`setAxisLabels(xlabel,ylabel)` Set the Axis labels

`addAxisName(axisName)` Adds an axis name

`delAxisName(axisName)` Deletes an axis name

Methods

`new(xml, element)` This method is used to instantiate a GML rectified grid

`setOrigin(x,y)` Set the origin of the rectified grid

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

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

GMLRectifiedGridCoverage

GMLRectifiedGridCoverage

Description

GMLRectifiedGridCoverage

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an GML rectified grid coverage

Methods

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

`setCoverageFunction(coverageFunction)` Set a coverage function, object of class `GMLGridFunction` or `GMLCoverageMappingRule`

Note

Class used internally by `geometa`

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

GMLReferenceableGridByArray
GMLReferenceableGridByArray

Description

GMLReferenceableGridByArray

Format

[R6Class](#) object.

Value

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

Methods

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

Note

Class used internally by geometa

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>
OGC GML 3.3 Schema. <http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd>

GMLReferenceableGridByTransformation
GMLReferenceableGridByTransformation

Description

GMLReferenceableGridByTransformation

Format

[R6Class](#) object.

Value

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

Methods

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

Note

Class used internally by geometa

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

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

OGC GML 3.3 Schema. <http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd>

GMLReferenceableGridByVectors

GMLReferenceableGridByVectors

Description

GMLReferenceableGridByVectors

Format

[R6Class](#) object.

Value

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

Methods

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

`setOrigin(coords)` Set the grid origin coordinates

`addGeneralGridAxis(axis)` Add a general grid axis definition, object of class GMLGeneralGridAxis.
Returns TRUE if added, FALSE otherwise.

`delGeneralGridAxis(axis)` Deletes a general grid axis definition, object of class GMLGeneralGridAxis.
Returns TRUE if deleted, FALSE otherwise.

Note

Class used internally by geometa

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

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

OGC GML 3.3 Schema. <http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd>

GMLSphericalCS

GMLSphericalCS

Description

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t

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

GMLTemporalCRS

GMLTemporalCRS

Description

GMLTemporalCRS

Format

[R6Class](#) object.

Value

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

Fields

timeCS [[GMLTimeCS](#)]

temporalDatum [[GMLTemporalDatum](#)]

Methods

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

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

GMLTemporalCS

GMLTemporalCS

Description

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

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

GMLTimeCS

GMLTimeCS

Description

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

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

GMLTimeInstant	<i>GMLTimeInstant</i>
----------------	-----------------------

Description

GMLTimeInstant

Format

R6Class object.

Value

Object of R6Class for modelling an GMLTimeInstant

Fields

timePosition [numeric|character|Date|POSIXt]

Methods

new(xml, timePosition) This method is used to instantiate an GMLTimeInstant
setTimePosition(timePosition) Sets the position (date or date and time of the resource contents), as object of class "POSIXct"/"POSIXt" or "Date"

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

Examples

```
time <- ISOdate(2000, 1, 12, 12, 59, 45)
md <- GMLTimeInstant$new(timePosition = time)
xml <- md$encode()
```

GMLTimePeriod	<i>GMLTimePeriod</i>
---------------	----------------------

Description

GMLTimePeriod

Format

R6Class object.

Value

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

Fields

beginPosition [[POSIXt](#)]

endPosition [[POSIXt](#)]

duration [[character](#)]

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

Format

[R6Class](#) object.

Value

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

Fields

quantityTypeReference [[character](#)]

catalogSymbol [[character](#)]

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

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

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.blondel1@gmail.com>

References

ISO 19136:2007 Geographic Information – Geographic Markup Language. http://www.iso.org/iso/iso_catalogue/catalogue_t

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

GMLVerticalCRS

GMLVerticalCRS

Description

GMLVerticalCRS

Format

[R6Class](#) object.

Value

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

Fields

`verticalCS` [[GMLVerticalCS](#)]

`verticalDatum` [`GMLVerticalDatum`]

Methods

`new(xml, defaults, id)` This method is used to instantiate a GML temporal 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
- OGC Geography Markup Language. <http://www.opengeospatial.org/standards/gml>

GMLVerticalCS

GMLVerticalCS

Description

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

INSPIREMetadataValidator

INSPIREMetadataValidator

Description

INSPIREMetadataValidator

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for setting an INSPIREMetadataValidator

Methods

`new()` This method is used to instantiate an INSPIRE Metadata validator

`uploadFile(path)` Upload a XML metadata file to INSPIRE web-service. Method called internally through `getValidationReport`.

`getValidationReport(obj, file, raw)` Get validation report for a metadata specified either as R object of class `ISOMetadata` (from **geometa** package) or `XMLInternalNode` (from **XML** package), or as XML file, providing the path of the XML file to be sent to the INSPIRE metadata validator web-service. By default, a summary report is returned. To append the raw response of INSPIRE validation web-service to the summary report, set `raw = TRUE`.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

INSPIRE Reference Validator Web Service (<http://inspire.ec.europa.eu/validator/swagger-ui.html>)

Examples

```
inspireValidator <- INSPIREMetadataValidator$new()
inspireReport <- inspireValidator$getValidationReport(obj = ISOMetadata$new())
```

ISOAbsoluteExternalPositionalAccuracy
ISOAbsoluteExternalPositionalAccuracy

Description

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.blondell@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()

```

ISOAbstractAggregate *ISOAbstractAggregate*

Description

ISOAbstractAggregate

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an ISOAbstractAggregate
`addComposedOf(composedOf)` Adds a dataset, object of class ISODataset
`delComposedOf(composedOf)` Deletes a dataset, object of class ISODataset
`addSeriesMetadata(metadata)` Adds a series metadata, object of class ISOMetadata
`delSeriesMetadata(metadata)` Deletes a series metadata, object of class ISOMetadata
`addSubset(subset)` Adds a subset, object that inherits from ISOAbstractAggregate
`delSubset(subset)` Deletes a subset, object that inherits from ISOAbstractAggregate
`addSuperset(superset)` Adds a superset, object that inherits from ISOAbstractAggregate
`delSuperset(superset)` Deletes a superset, object that inherits from ISOAbstractAggregate

Note

abstract class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractCarrierOfCharacteristics
ISOAbstractCarrierOfCharacteristics

Description

ISOAbstractCarrierOfCharacteristics

Format

[R6Class](#) object.

Value

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

Fields

featureType [[ISOFeatureType](#)]
constrainedBy [[ISOConstraint](#)]

Methods

new(xml, defaults) This method is used to instantiate an [ISOAbstractCarrierOfCharacteristics](#)
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](#)

Note

abstract class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOAbstractCatalogue *ISOAbstractCatalogue*

Description

ISOAbstractCatalogue

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)]
scope [[character](#)]
fieldOfApplication [[character](#)]
versionNumber [[character](#)]
versionDate [[character](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOAbstractCatalogue](#)

`setName(name, locales)` Sets the name. Locale names can be specified as list with the locales argument.

`addScope(scope, locales)` Adds scope (object of class character). Locale names can be specified as list with the locales argument.

`delScope(scope, locales)` Deletes scope. Locale names can be specified as list with the locales argument. Local names should match those of the scope to be deleted, otherwise nothing will be deleted.

`addFieldOfApplication(fieldOfApplication, locales)` Adds a field of application (object of class character). Locale names can be specified as list with the locales argument.

`delFieldOfApplication(fieldOfApplication, locales)` Deletes fieldOfApplication. Locale names can be specified as list with the locales argument. Local names should match those of the field of application to be deleted, otherwise nothing will be deleted.

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

`setVersionDate(versionDate)` Sets version date

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19139:2007 Metadata - XML schema implementation

ISOAbstractCompleteness

ISOAbstractCompleteness

Description

ISOAbstractCompleteness

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOAbstractCompleteness](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractGenericName

ISOAbstractGenericName

Description

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

ISOAbstractLogicalConsistency

ISOAbstractLogicalConsistency

Description

ISOAbstractLogicalConsistency

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOAbstractLogicalConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

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

Description

ISOAbstractObject

Format

[R6Class](#) object.

Value

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

Static Methods

`getISOStandardByPrefix(prefix)` Inherit the ISO (and/or OGC) standard reference for a given standard prefix (e.g. GMD). The object returned is a `data.frame` containing the specification reference and title.

`getISOStandard(clazz)` Inherit the ISO (and/or OGC) standard reference for a given **geometa** class. The object returned is a `data.frame` containing the specification reference and title.

`getISOClasses(extended, pretty)` Get the list of classes supported by **geometa**. By default, `extended` is set to `FALSE` (restrained to **geometa** environment). If `TRUE`, this allows to list eventual classes loaded in your global environment and that extend **geometa** classes. The argument `pretty` gives a the list of classes and associated ISO/OGC standard information as `data.frame`.

`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`
- `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, inspire, resetSerialID, setSerialID, encoding)` 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`. The argument `setSerialID` is used by **geometa** to generate automatically serial IDs associated to XML elements, in particular for GML, default value is `TRUE` (recommended value). The argument `resetSerialID` is used by **geometa** for resetting mandatory IDs associated to XML elements, such as GML objects, default value is `TRUE` (recommended value). Setting `inspire` to `TRUE` (default `FALSE`), the metadata will be checked with the INSPIRE metadata validator (online web-service provided by INSPIRE).
- `validate(xml, strict, inspire)` Validates the encoded XML against ISO 19139 XML schemas. If `strict` is `TRUE`, a error will be raised. Default is `FALSE`. Setting `inspire` to `TRUE` (default `FALSE`), the metadata will be checked with the INSPIRE metadata validator (online web-service provided by INSPIRE).
- `save(file, ...)` Saves the current metadata object XML representation to a file. This utility ensures proper indentation of XML file produced. Additional parameters from `$encode()` method can be specified, such as `inspire` to check the INSPIRE metadata validity.
- `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
- `setIsNull(isNull, reason)` Sets the object as null object for the XML. In case `isNull` is `TRUE`, a reason should be specified among values 'inapplicable', 'missing', 'template', 'unknown', 'withheld'. By default, the reason is set 'missing'.
- `contains(field, metadataElement)` Indicates of the present class object contains an 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
- `addFieldAttrs(field, ...)` Allows to add one more xlink attributes a field (element property)
- `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.
createLocalisedProperty(text, locales) Creates a localised property made of a default text and a list of localised texts.

Note

Abstract ISO Metadata class used internally by geometa

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

ISOAbstractPositionalAccuracy

ISOAbstractPositionalAccuracy

Description

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.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractPropertyType

ISOAbstractPropertyType

Description

ISOAbstractPropertyType

Format

[R6Class](#) object.

Value

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

Fields

memberName [[ISOLocalName](#)]

definition [[character](#)]

cardinality [[ISOMultiplicity](#)]

definitionReference [[ISODefinitionReference](#)]

featureCatalogue [[ISOFeatureCatalogue](#)]

Methods inherited from [ISOAbstractCarrierOfCharacteristics](#)

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 [ISOAbstractPropertyType](#)

setMemberName(memberName) Sets the member name. Object of class [ISOLocalName](#) or "character"

setDefinition(definition, locales) Sets the definition. Locale names can be specified as list with the locales argument.

setCardinality(lower, upper) Sets the cardinality boundaries lower and upper of class numeric

setDefinitionReference(definitionReference) Sets the definition Reference, object of class [ISODefinitionReference](#)

setFeatureCatalogue(featureCatalogue) Sets a feature catalogue, object of class [ISOFeatureCatalogue](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOAbstractReferenceSystem

ISOAbstractReferenceSystem

Description

ISOAbstractReferenceSystem

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO abstract RS Reference system

Methods

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

`setName(name)` Set the name, object of class ISOReferenceIdentifier

`addDomainOfValidity(domainOfValidity)` Adds a domain of validity, object of class ISOExtent

`delDomainOfValidity(domainOfValidity)` Deletes a domain of validity, object of class ISOExtent

Note

abstract class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractResult *ISOAbstractResult*

Description

ISOAbstractResult

Format

[R6Class](#) object.

Value

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

Methods

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

Note

abstract class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractTemporalAccuracy
ISOAbstractTemporalAccuracy

Description

ISOAbstractTemporalAccuracy

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOAbstractTemporalAccuracy](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAbstractThematicAccuracy
ISOAbstractThematicAccuracy

Description

ISOAbstractThematicAccuracy

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOAbstractThematicAccuracy](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

 ISOAccuracyOfATimeMeasurement

ISOAccuracyOfATimeMeasurement

Description

ISOAccuracyOfATimeMeasurement

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOAccuracyOfATimeMeasurement](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOAccuracyOfATimeMeasurement$new()
dq$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()

```

ISOAddress

ISOAddress

Description

ISOAddress

Format

[R6Class](#) object.

Value

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

Fields

deliveryPoint [[character](#)]
 city [[character](#)]
 postalCode [[character](#)]
 country [[ISOCountry](#)]
 electronicMailAddress [[character](#)]

Methods

new(xml) This method is used to instantiate an [ISOAddress](#)
 setDeliveryPoint(deliveryPoint, locales) Sets the delivery point. Locale names can be specified as list with the locales argument.
 setCity(city, locales) Sets the city. Locale names can be specified as list with the locales argument.
 setPostalCode(postalCode, locales) Sets the postal code. Locale names can be specified as list with the locales argument.
 setCountry(country, locales) Sets the country. Locale names can be specified as list with the locales argument.
 setEmail(email, locales) Sets the electronic Mail address. Locale names can be specified as list with the locales argument.

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()
```

ISOAggregateInformation

ISOAggregateInformation

Description

ISOAggregateInformation

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling a ISO AggregateInformation

Fields

aggregateDataSetName [[ISOCitation](#)]
aggregateDataSetIdentifier [[ISOMetaIdentifier](#)]
associationType [[ISOAssociationType](#)]
initiativeType [[ISOInitiativeType](#)]

Methods

new(xml) This method is used to instantiate an [ISOAggregateInformation](#)

setAggregateDataSetName(datasetName) Sets aggregate dataset name, as an object of class [ISOCitation](#)

setAggregateDataSetIdentifier(datasetIdentifier) Sets aggregate dataset identifier, as an object of class [ISOMetaIdentifier](#)

setAssociationType(associationType) Sets the association type

setInitiativeType(initiativeType) Sets the initiative type

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
md <- ISOAggregateInformation$new()

#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)

#citation
ct <- ISOCitation$new()
ct$setTitle("sometitle")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
```

```

d$dateType("publication")
ct$addDate(d)
ct$setEdition("1.0")
ct$setEditionDate(ISOdate(2015,1,1))
ct$addIdentifier(ISOmetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp)
md$setAggregateDataSetName(ct)

md$setAssociationType("source")
md$setInitiativeType("investigation")

xml <- md$encode()

```

ISOAnchor

ISOAnchor

Description

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

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

ISOApplicationSchemaInformation

ISOApplicationSchemaInformation

Description

ISOApplicationSchemaInformation

Format

[R6Class](#) object.

Value

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

Fields

name [[ISOCitation](#)]
schemaLanguage [[character](#)]
constraintLanguage [[character](#)]
schemaAscii [[character](#)]
graphicsFile [[ISOBinary](#)]
softwareDevelopmentFile [[ISOBinary](#)]
softwareDevelopmentFileFormat [[character](#)]

Methods

new(xml) This method is used to instantiate an [ISOApplicationSchemaInformation](#)
setName(name) Sets name, object of class [ISOCitation](#)
setSchemaLanguage(schemaLanguage) Sets schema language
setConstraintLanguage(constraintLanguage) Sets constraint language
setSchemaAscii(schemaAscii) Sets schema Ascii
setGraphicsFile(graphicsFile) Sets graphics file
setSoftwareDevelopmentFile(file) Sets software development file
setSoftwareDevelopmentFileFormat(format) Sets software development file format

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAssociation

ISOAssociation

Description

ISOAssociation

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOAssociation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOAssociationRole	<i>ISOAssociationRole</i>
--------------------	---------------------------

Description

ISOAssociationRole

Format

[R6Class](#) object.

Value

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

Fields

type [[ISORoleType](#)]
isOrdered [[logical](#)]
isNavigable [[logical](#)]
relation [[ISOFeatureAssociation](#)]
rolePlayer [[ISOFeatureType](#)]

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.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOAssociationType *ISOAssociationType*

Description

ISOAssociationType

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOAssociationType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#geomOnly
geomOnly <- ISOAssociationType$new(value = "source")
```

ISOAttributes

ISOAttributes

Description

ISOAttributes

Format

[R6Class](#) object.

Value

Spatial object of [R6Class](#) for modelling a list of ISO xml attributes

Methods

`new(...)` This method is used to instantiate a vector of attributes to be used for empty element properties.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

```
attrs <- ISOAttributes$new(href = "http://somelink", title = "sometitle")
```

ISOBand

ISOBand

Description

ISOBand

Format

[R6Class](#) object.

Value

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

Fields

maxValue [numeric]
 minValue [numeric]
 units [GMLUnitDefinition]
 peakResponse [numeric]
 bitsPerValue [integer]
 toneGradation [integer]
 scaleFactor [numeric]
 offset [numeric]

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.blondell@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

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

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

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

ISOBaseDateTime

Description

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

ISOBaseDecimal

Description

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

ISOBaseInteger

Description

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

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

ISOBinary

ISOBinary

Description

ISOBinary

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] binary

Methods

new(xml, value) This method is used to instantiate an [ISOBinary](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

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

Examples

```
bin <- ISOBinary$new(value = "http://someuri")
```

ISOBinding

ISOBinding

Description

ISOBinding

Format

[R6Class](#) object.

Value

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

Fields

description [[character](#)] description

globalProperty [[ISOPropertyType](#)] property type

Methods

- `new(xml, defaults)` This method is used to instantiate an [ISOBinding](#)
- `setDescription(description, locales)` Set description of inheritance relation. Locale names can be specified as list with the locales argument.
- `setPropertyType(propertyType)` Set global property, object of class [ISOPropertyType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOBoundAssociationRole

ISOBoundAssociationRole

Description

ISOBoundAssociationRole

Format

[R6Class](#) object.

Value

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

Fields

`rolePlayer` [[ISOFeatureType](#)] feature type

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.blondell@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOBoundFeatureAttribute
ISOBoundFeatureAttribute

Description

ISOBoundFeatureAttribute

Format

[R6Class](#) object.

Value

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

Fields

valueType [[ISOTypeName](#)] value type

Methods

new(xml, defaults) This method is used to instantiate an [ISOBoundFeatureAttribute](#)

setTypeNames(typeNames) Set typeNames, object of class [ISOTypeName](#) or character

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOBoundingPolygon *ISOBoundingPolygon*

Description

ISOBoundingPolygon

Format

[R6Class](#) object.

Value

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

Fields

polygon [[GMLAbstractGeometry](#)] a (list of) geometrie(s)

Methods

new(xml, sfg) This method is used to instantiate an [ISOBoundingPolygon](#)

Note

Experimental

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOBrowseGraphic *ISOBrowseGraphic*

Description

ISOBrowseGraphic

Format

[R6Class](#) object.

Value

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

Fields

fileName [[character](#)] file name

fileDescription [[character](#)] file description

fileType [[character](#)] file type

Methods

new(xml, fileName, fileDescription, fileType) This method is used to instantiate an [ISOBrowseGraphic](#)

setFileName(fileName, locales) Set file name. Locale names can be specified as list with the locales argument.

setFileDescription(fileDescription, locales) Set file description. Locale names can be specified as list with the locales argument.

setFileType(fileType, locales) Set file type. Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

Methods inherited from [ISOAbstractCarrierOfCharacteristics](#)

[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](#)

Methods

[new\(xml, defaults\)](#) This method is used to instantiate an [ISOCarrierOfCharacteristics](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOCellGeometry *ISOCellGeometry*

Description

ISOCellGeometry

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISOCellGeometry](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

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

Format

[R6Class](#) object.

Value

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

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	<i>ISOCitation</i>
-------------	--------------------

Description

ISOCitation

Format

[R6Class](#) object.

Value

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

Fields

title [[character](#)] title
 alternateTitle [list of [character](#)] alternateTitle
 date [[ISODate](#)] the citation date with date type
 edition [[character](#)] citation edition
 editionDate [[DatePOSIXt](#)] date or date/time of edition
 identifier [list of [ISOMetaIdentifier](#)] identifier
 citedResponsibleParty [list of [ISOResponsibleParty](#)] responsible party
 presentationForm [list of [ISOPresentationForm](#)] presentation form
 series [link{[ISOCitationSeries](#)}] series
 otherCitationDetails [[character](#)] other citation details
 collectiveTitle [[character](#)] collective title
 ISBN [[character](#)] ISBN
 ISSN [[character](#)] ISSN

Methods

new(xml) This method is used to instantiate an [ISOCitation](#)
 setTitle(title, locales) Sets the title. Locale names can be specified as list with the locales argument.
 setAlternateTitle(alternateTitle, locales) Sets an alternate title. Locale names can be specified as list with the locales argument.
 addAlternateTitle(alternateTitle, locales) Adds an alternate title. Locale names can be specified as list with the locales argument.
 delAlternateTitle(alternateTitle, locales) Adds an alternate title. Locale names can be specified as list with the locales argument.
 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](#)
 addIdentifier(identifier) Adds an identifier as object of class [ISOMetaIdentifier](#)
 delIdentifier(identifier) Deletes an identifier as object of class [ISOMetaIdentifier](#)
 setCitedResponsibleParty(rp) Sets the cited responsible party, object of class [ISOResponsibleParty](#)
 setPresentationForm(presentationForm) Sets the presentation form, object of class [ISOPresentationForm](#)

addPresentationForm(presentationForm) Adds a presentation form, object of class `ISOPresentationForm`
 delPresentationForm(presentationForm) Deletes a presentation form, object of class `ISOPresentationForm`
 setSeries(series) Set series, object of class `ISOCitationSeries`
 setOtherCitationDetails(otherCitationDetails, locales) Set other citation details. Locale names can be specified as list with the locales argument.
 setCollectiveTitle(collectiveTitle, locales) Set collective title. Locale names can be specified as list with the locales argument.
 setISBN(isbn) Set the ISBN
 setISSN(issn) Set the ISSN

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
md$addPresentationForm("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$setContactInfo(contact)
md$addCitedResponsibleParty(rp)
xml <- md$encode()
```

ISOCitationSeries *ISOCitationSeries*

Description

ISOCitationSeries

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)] citation series name
issueIdentification [[character](#)] issue identification number
page [[character](#)] page

Methods

`new(xml)` This method is used to instantiate an ISOCitationSeries
`setName(name, locales)` Sets a name (object of class "character"). Locale names can be specified as list with the locales argument.
`setIssueIdentification(issueId, locales)` Sets an issue identification (object of class "character"). Locale names can be specified as list with the locales argument.
`setPage(page, locales)` Sets a page (object of class "character"). Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOClassification	<i>ISOClassification</i>
-------------------	--------------------------

Description

ISOClassification

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOClassification](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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	<i>ISOCodelist</i>
-------------	--------------------

Description

ISOCodelist

Format

[R6Class](#) object.

Value

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

Fields

id [[character](#)] codelist id
 refFile [[character](#)] reference XML file
 codeSpace [[character](#)] codelist codeSpace
 identifier [[character](#)] codelist identifier
 description [[character](#)] codelist description
 entries [[data.frame](#)] the list of codelist entries

Methods

new(refFile, id) 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

ISOCodeListValue

Description

ISOCodeListValue

Format

[R6Class](#) object.

Value

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

Fields

codelistId [[character](#)] codelist identifier
 attrs [[list](#)] list of codelist attributes
 value [[character](#)] codelist value
 valueDescription [[character](#)] codelist value description

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`). The argument `setValueDescription` allows to force having description set as value, default is `FALSE` in which case the name will be preferred, and in case no name is provided, code value will be used.

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

Note

Abstract ISO codelist class used internally by `geometa`

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOCompletenessCommission

ISOCompletenessCommission

Description

ISOCompletenessCommission

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an `ISOCompletenessCommission`

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOCompletenessCommission](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOCompletenessCommission$new()
dq$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()
```

ISOCompletenessOmission

ISOCompletenessOmission

Description

ISOCompletenessOmission

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOCompletenessOmission](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOCompletenessOmission$new()
dq$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()
```

ISOConceptualConsistency

ISOConceptualConsistency

Description

ISOConceptualConsistency

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOConceptualConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOConceptualConsistency$new()
dq$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()
```

ISOConformanceResult *ISOConformanceResult*

Description

ISOConformanceResult

Format

R6Class object.

Value

Object of R6Class for modelling an ISO ConformanceResult

Fields

specification [ISOCitation] specification citation
explanation [character] explanation about conformance
pass [logical] if the conformance is passing or not

Methods

new(xml, value) This method is used to instantiate an ISOConformanceResult
setSpecification(specification) Sets the specification (an ISOCitation object)
setExplanation(explanation, locales) Sets the explanation. Locale names can be specified as list with the locales argument.
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

Format

[R6Class](#) object.

Value

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

Fields

description [[character](#)]

Methods

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

`setDescription(description, locales)` Sets the description. Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

Examples

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

ISOConstraints

ISOConstraints

Description

ISOConstraints

Format

[R6Class](#) object.

Value

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

Fields

useLimitation [[character](#)] use limitation

Methods

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

`addUseLimitation(useLimitation, locales)` Adds a use limitation. Locale names can be specified as `list` with the `locales` argument.

`setUseLimitation(useLimitation, locales)` Sets a use limitation. Locale names can be specified as `list` with the `locales` argument.

`delUseLimitation(useLimitation, locales)` Deletes a use limitation. Locale names can be specified as `list` with the `locales` argument.

Note

Abstract ISO class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOContact

ISOContact

Description

ISOContact

Format

[R6Class](#) object.

Value

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

Fields

phone [[ISOTelephone](#)] phone
address [[ISOAddress](#)] address
onlineResource [[ISOOnlineResource](#)] online resource

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

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.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOCountry

ISOCountry

Description

ISOCountry

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOCountry](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

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

ISOCoupledResource

ISOCoupledResource

Description

ISOCoupledResource

Format

[R6Class](#) object.

Value

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

Fields

operationName [[character](#)] operation name

identifier [[character](#)] identifier

Methods

new(xml) This method is used to instantiate an [ISOCoupledResource](#)

setOperationName(operationName, locales) Set the operation name. Locale names can be specified as list with the locales argument.

setIdentifier(identifier, locales) Set the identifier. Locale names can be specified as list with the locales argument.

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

ISOCouplingType

Description

ISOCouplingType

Format

[R6Class](#) object.

Value

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

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

Format

[R6Class](#) object.

Value

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

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

Format

[R6Class](#) object.

Value

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

Fields

attributeDescription [[ISORecordType](#)] attribute description
contentType [[character](#)] content type
dimension [[ISORangeDimension](#)] range dimension(s)

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()

```

ISODataFile

*ISODataFile***Description**

ISODataFile

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

Methods

`new(xml)` This method is used to instantiate an [ISODataFile](#)

`setFileName(fileName)` Set the file name, object of class [ISOFileName](#)

`setDescription(fileDescription, locales)` Set the file description, object of class 'character'. Locale names can be specified as list with the `locales` argument.

`setFileType(type)` Set the file type, object of class [ISOMimeType](#).

`addFeatureType(featureType)` Add a feature type, object of class [ISOLocalName](#) or `link{ISOScopedName}`, or eventually a 'character' in which case the `featureType` will be coerced to a local name.

`delFeatureType(featureType)` Deletes a feature type, object of class [ISOLocalName](#) or `link{ISOScopedName}`, or eventually a 'character' in which case the `featureType` will be coerced to a local name.

`setFileFormat(fileFormat)` Set the file format, object of class [ISOFormat](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO/TS 19139:2007 Geographic information – XML

Examples

```
md <- ISODataFile$new()
md$setFileName(ISOFileName$new(file = "someuri", name = "filename"))
md$setDescription("description")
md$setFileType(ISOMimeType$new(type = "sOMEMIMETYPE", name = "Mime type name"))
md$addFeatureType("feature_type")
f <- ISOFormat$new()
f$setName("name")
f$setVersion("1.0")
f$setAmendmentNumber("2")
f$setSpecification("specification")
md$setFileFormat(f)
xml <- md$encode()
```

ISODataIdentification *ISODataIdentification*

Description

ISODataIdentification

Format

[R6Class](#) object.

Value

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

Fields

spatialRepresentationType [[ISOSpatialRepresentationType](#)] spatial representation type

spatialResolution [[ISOResolution](#)] spatial resolution(s)

language [[ISOLanguage](#)] locale language(s)

characterSet [[ISOCharacterSet](#)] character set

topicCategory [[ISOTopicCategory](#)] topic categorie(s)

environmentDescription [[character](#)] environment description

extent [[ISOExtent](#)] extent(s)

supplementalInformation [[character](#)] supplemental information

Inherited methods

setCitation(citation) Sets an object of class [ISOCitation](#)

setAbstract(abstract, locales) Sets an abstract (object of class "character"). Locale names can be specified as list with the locales argument.

setPurpose(purpose, locales) Sets a purpose (object of class "character"). Locale names can be specified as list with the locales argument.

addCredit(credit, locales) Adds a credit (object of class "character"). Locale names can be specified as list with the locales argument.

delCredit(credit, locales) Deletes a credit (object of class "character"). Locale names can be specified as list with the locales argument.

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\(\)](#).
 addSpatialResolution(resolution) Adds a spatial resolution, object of class [ISOResolution](#)
 delSpatialResolution(resolution) Deletes a spatial resolution, object of class [ISOResolution](#)
 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, locales) Sets supplemental information (object of class "character"). Locale names can be specified as list with the locales argument.

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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(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()
```

ISODataQuality

ISODataQuality

Description

ISODataQuality

Format

[R6Class](#) object.

Value

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

Fields

scope [[ISOScope](#)] scope
report [[ISODomainConsistency](#)] report
lineage [[ISOLineage](#)] lineage information

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 with a 'dataset' scope
dq <- ISODataQuality$new()
scope <- ISOScope$new()
scope$setLevel("dataset")
dq$setScope(scope)

#add data quality reports...

#add a report the data quality
dc <- ISODomainConsistency$new()
result <- ISOConformanceResult$new()
spec <- ISOCitation$new()
spec$setTitle("Data Quality check")
spec$setAlternateTitle("This is is some data quality check report")
d <- ISODate$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 INSPIRE reports?
#INSPIRE - interoperability of spatial data sets and services
dc_inspire1 <- ISODomainConsistency$new()
cr_inspire1 <- ISOConformanceResult$new()
cr_inspire_spec1 <- ISOCitation$new()
cr_title <- paste(
  "Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards interoperability of spatial data",
  "sets and services"
)
cr_inspire_spec1$setTitle(cr_title)
cr_inspire1$setExplanation("See the referenced specification")
cr_inspire_date1 <- ISODate$new()
cr_inspire_date1$setDate(ISOdate(2010,12,8))
cr_inspire_date1$setDateType("publication")
cr_inspire_spec1$addDate(cr_inspire_date1)
cr_inspire1$setSpecification(cr_inspire_spec1)
cr_inspire1$setPass(TRUE)
dc_inspire1$addResult(cr_inspire1)
dq$addReport(dc_inspire1)
#INSPIRE - metadata
dc_inspire2 <- ISODomainConsistency$new()
cr_inspire2 <- ISOConformanceResult$new()
cr_inspire_spec2 <- ISOCitation$new()
cr_title2 <- paste(
  "COMMISSION REGULATION (EC) No 1205/2008 of 3 December 2008 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards metadata"
)

```

```

)
cr_inspire_spec2$title(cr_title2)
cr_inspire2$setExplanation("See the referenced specification")
cr_inspire_date2 <- ISODate$new()
cr_inspire_date2$date(ISOdate(2008,12,4))
cr_inspire_date2$dateType("publication")
cr_inspire_spec2$addDate(cr_inspire_date2)
cr_inspire2$setSpecification(cr_inspire_spec2)
cr_inspire2$setPass(TRUE)
dc_inspire2$addResult(cr_inspire2)
dq$addReport(dc_inspire2)

#add lineage (more example of lineages in ISOLineage documentation)
lineage <- ISOLineage$new()
lineage$setStatement("statement")
dq$setLineage(lineage)

#xml
xml <- dq$encode()

```

ISODataQualityAbstractElement

ISODataQualityAbstractElement

Description

ISODataQualityAbstractElement

Format

[R6Class](#) object.

Value

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

Fields

nameOfMeasure [[character](#)] measure name
 measureIdentification [[ISOMetaIdentifier](#)] measure identification
 measureDescription [[character](#)] measure description
 evaluationMethodType [[ISOEvaluationMethodType](#)] measure evaluation method type
 evaluationMethodDescription [[character](#)] measure evaluation method description
 evaluationProcedure [[character](#)] evaluation procedure
 dateTime [[POSIXt](#)] dateTime
 result [[ISOConformanceResult](#)] conformance result(s)

Methods

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

`addNameOfMeasure(name, locales)` Add name. Locale names can be specified as `list` with the `locales` argument.

`delNameOfMeasure(name, locales)` Deletes name. Locale names can be specified as `list` with the `locales` argument. Local names should match those of the name to be deleted, otherwise nothing will be deleted.

`setMeasureIdentification(identification)` Sets measure identifier, an object of class `ISOMetaIdentifier`

`setMeasureDescription(description, locales)` Sets measure description. Locale names can be specified as `list` with the `locales` argument.

`setEvaluationMethodType(type)` Sets the type of evaluation method, an object of class `ISOEvaluationMethodType` or any character value among `codeISOEvaluationMethodType$values()`

`setEvaluationMethodDescription(description, locales)` Sets the description of evaluation method. Locale names can be specified as `list` with the `locales` argument.

`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

ISODataSet

ISODataSet

Description

ISODataSet

Format

[R6Class](#) object.

Value

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

Fields

has [[ISOMetadata](#)] associated metadata
 partOf [[ISOAbstractAggregate](#)] part of other aggregate(s)

Methods

new(xml) This method is used to instantiate an [ISODataset](#)
 addHas(metadata) Adds a metadata, object of class [ISOMetadata](#)
 delHas(metadata) Deletes a metadata, object of class [ISOMetadata](#)
 addPartOf(partOf) Adds a partOf element, object that inherits from [ISOAbstractAggregate](#)
 delPartOf(partOf) Deletes a partOf element, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISODatatype

ISODatatype

Description

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	<i>ISODate</i>
---------	----------------

Description

ISODate

Format

[R6Class](#) object.

Value

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

Fields

date [[ISOBaseDate](#)] the date
dateType [[ISODateType](#)] the date type

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$new()
md$setDate(ISOdate(2015, 1, 1, 1))
md$setDateType("publication")
xml <- md$encode()
```

ISODateType

ISODateType

Description

ISODateType

Format

[R6Class](#) object.

Value

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

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

Format

[R6Class](#) object.

Value

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

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

Format

[R6Class](#) object.

Value

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

Fields

sourceIdentifier [[character](#)] source identifier

definitionSource [[ISODefinitionSource](#)] definition source

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

Format

[R6Class](#) object.

Value

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

Fields

source [[ISOCitation](#)] source citation

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.blondell@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISODigitalTransferOptions

ISODigitalTransferOptions

Description

ISODigitalTransferOptions

Format

[R6Class](#) object.

Value

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

Fields

unitsOfDistribution [[character](#)] units of distribution

transferSize [[character](#)] transfer size

onLine [[ISOOnlineResource](#)] online resource(s)

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 [ISOOnlineResource](#)

setOnlineResource(onlineResource) Sets an object of class [ISOOnlineResource](#)

delOnlineResource(onlineResource) Deletes an object of class [ISOOnlineResource](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

Format

[R6Class](#) object.

Value

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

Fields

dimensionName [[ISODimensionNameType](#)] dimension name

dimensionSize [[integer](#)] dimension size

resolution [[ISOMeasure](#)] 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

Format

[R6Class](#) object.

Value

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

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

Format

[R6Class](#) object.

Value

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

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

ISODistribution

Description

ISODistribution

Format

[R6Class](#) object.

Value

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

Fields

distributionFormat [[ISOFormat](#)]
distributor [[ISODistributor](#)]
transferOptions [[ISODigitalTransferOptions](#)]

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, object of class [ISODigitalTransferOptions](#)

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()
```

ISODistributionUnits *ISODistributionUnits*

Description

ISODistributionUnits

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISODistributionUnits](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
unit <- ISODistributionUnits$new(value = "unit")
```

ISODistributor *ISODistributor*

Description

ISODistributor

Format

[R6Class](#) object.

Value

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

Fields

distributorContact [ISOResponsibleParty]
 distributorFormat [ISOFormat]

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$setName("somenam")
contact$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

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISODomainConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISODomainConsistency$new()
dq$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()
```

ISOElementSequence *ISOElementSequence*

Description

ISOElementSequence

Format

[R6Class](#) object.

Value

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

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

Format

[R6Class](#) object.

Value

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

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

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)]
shortName [[character](#)]
domainCode [[integer](#)]
definition [[character](#)]
obligation [[ISOobligation](#)]
condition [[character](#)]
dataType [[ISODateType](#)]
maximumOccurrence [[character](#)]
domainValue [[character](#)]
parentEntity [[character](#)]
rule [[character](#)]
rationale [[character](#)]
source [[ISOResponsibleParty](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOExtendedElementInformation](#)

`setName(name, locales)` Sets the element name, object of class `Character`. Locale names can be specified as list with the `locales` argument.

`setShortName(shortName, locales)` Sets the element shortname, object of class `character`. Locale names can be specified as list with the `locales` argument.

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

`setDefinition(definition, locales)` Sets the element definition, object of class `character`. Locale names can be specified as list with the `locales` argument.

- `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$values()`.
- `setCondition(condition, locales)` Sets the element condition, object of class `character`. Locale names can be specified as `list` with the `locales` argument.
- `setDatatype(dataType)` Sets the element datatype, as object of class `character` or class `ISODatatype`.
If an object of class "character" is specified, it must match the accepted datatype values `ISODatatype$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, locales)` Sets a rule, object of class `character`. Locale names can be specified as `list` with the `locales` argument.
- addRationale(rationale, locales)** Adds a rationale, object of class `character`. Locale names can be specified as `list` with the `locales` argument.
- delRationale(rationale, locales)** Deletes a rationale, object of class `character`. Locale names can be specified as `list` with the `locales` argument. Local names should match those of the keyword to be deleted, otherwise nothing will be deleted.
- addSource(source)** Adds a source, object of class `ISOresponsibleParty`
- delSource(source)** Deletes a source, object of class `ISOresponsibleParty`

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

Format

[R6Class](#) object.

Value

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

Fields

geographicElement [[ISOGeographicExtent](#)] geographic extent(s)
temporalElement [[ISOTemporalExtent](#)] temporal extent(s)
verticalElement [[ISOVerticalExtent](#)] vertical extent(s)

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.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOFeatureAssociation *ISOFeatureAssociation*

Description

ISOFeatureAssociation

Format

[R6Class](#) object.

Value

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

Fields

`roleName` [[ISOAssociationRole](#)] association role(s)

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.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOFeatureAttribute *ISOFeatureAttribute*

Description

ISOFeatureAttribute

Format

[R6Class](#) object.

Value

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

Fields

code [[character](#)]
valueMeasurementUnit [[GMLUnitDefinition](#)]
valueType [[character](#)]
listedValue [[ISOListedValue](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOFeatureAttribute](#)
`setCode(code, locales)` Sets the code. Locale names can be specified as list with the locales argument.
`setValueMeasurementUnit(uom)` Sets the value measurement unit, an object of class [GMLUnitDefinition](#)
`setValueType(typeName, locales)` Sets the value type. Locale names can be specified as list with the locales argument.
`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.blondel1@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

Format

[R6Class](#) object.

Value

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

Fields

producer [ISOResponsibleParty]
functionalLanguage [ISOLanguage]
featureType [ISOFeatureType]
definitionSource [ISODefinitionSource]

Methods inherited from ISOAbstractCatalogue

setName(name, locales) Sets the name. Locale names can be specified as list with the locales argument.

addScope(scope, locales) Adds scope (object of class character). Locale names can be specified as list with the locales argument.

delScope(scope, locales) Deletes scope. Locale names can be specified as list with the locales argument. Local names should match those of the scope to be deleted, otherwise nothing will be deleted.

addFieldOfApplication(fieldOfApplication, locales) Adds a field of application (object of class character). Locale names can be specified as list with the locales argument.

delFieldOfApplication(fieldOfApplication, locales) Deletes fieldOfApplication. Locale names can be specified as list with the locales argument. Local names should match those of the field of application to be deleted, otherwise nothing will be deleted.

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.blondel1@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

Format

[R6Class](#) object.

Value

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

Fields

complianceCode [[logical](#)]
language [[ISOLanguage](#)]
includedWithDataset [[logical](#)]
featureCatalogueCitation [[ISOCitation](#)]

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, uuid)` Adds an object of class `ISOCitation` referencing the link to Feature Catalogue

`delFeatureCatalogueCitation(citation, uuid)` 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

Format

`R6Class` object.

Value

Object of `R6Class` for modelling an ISOFeatureOperation

Fields

signature [`character`]
formalDefinition [`character`]

Methods

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

`setSignature(signature, locales)` Sets the signature. Locale names can be specified as list with the locales argument.

`setFormalDefinition(formalDefinition, locales)` Sets the formal definition. Locale names can be specified as list with the locales argument.

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

Format

[R6Class](#) object.

Value

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

Fields

typeName [[ISOLocalName](#)]
definition [[character](#)]
code [[character](#)]
isAbstract [[logical](#)]

aliases [ISOLocalName]
 inheritsFrom [ISOInheritanceRelation]
 inheritsTo [ISOInheritanceRelation]
 featureCatalogue [ISOFeatureCatalogue]
 constrainedBy [ISOConstraints]
 definitionReference [ISODefinitionReference]
 carrierOfCharacteristics [ISOCarrierOfCharacteristics]

Methods

new(xml) This method is used to instantiate an ISOFeatureType
 setType_name(type_name) Sets the type name. Object of class ISOLocalName or "character"
 setDefinition(definition, locales) Sets the definition. Locale names can be specified as list with the locales argument.
 setCode(code, locales) Sets the code. Locale names can be specified as list with the locales argument.
 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
 addInheritsFrom(rel) Adds a relation (from) as object of class ISOInheritanceRelation
 delInheritsFrom(rel) Deletes a relation (from) as object of class ISOInheritanceRelation
 addInheritsTo(rel) Adds a relation (to) as object of class ISOInheritanceRelation
 delInheritsTo(rel) Deletes a relation (to) as object of class ISOInheritanceRelation
 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$new()
md$setType("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()

```

Description

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

Format

[R6Class](#) object.

Value

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

Fields

name [character] format name
version [character] format version
amendmentNumber [character] format amendment number
specification [character] format specification
fileDecompressionTechnique [character] technique for file decompression
formatDistributor [ISODistributor] format distributor

Methods

new(xml) This method is used to instantiate an ISOFormat
setName(name, locales) Sets the format name. Locale names can be specified as list with the locales argument.
setVersion(version) Sets the format version
setAmendmentNumber(amendmentNumber) Sets an admenment number
setSpecification(specification, locales) Sets the format specification. Locale names can be specified as list with the locales argument.
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")
```

ISOFormatConsistency *ISOFormatConsistency*

Description

ISOFormatConsistency

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOFormatConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOFormatConsistency$new()
dq$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()
```

ISOFreeText

ISOFreeText

Description

ISOFreeText

Format

[R6Class](#) object.

Value

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

Fields

textGroup [[ISOLocalisedCharacterString](#)] localised string(s)

Methods

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

addTextGroup(textGroup) Add a text group, object of class [ISOLocalisedCharacterString](#)

delTextGroup(textGroup) Deletes a text group, object of class [ISOLocalisedCharacterString](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
ft <- ISOFreeText$new()
```

ISOGeographicBoundingBox

ISOGeographicBoundingBox

Description

ISOGeographicBoundingBox

Format

[R6Class](#) object.

Value

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

Fields

westBoundLongitude [[numeric](#)] west longitude

eastBoundLongitude [[numeric](#)] east longitude

southBoundLatitude [[numeric](#)] south latitude

northBoundLatitude [[numeric](#)] north latitude

Methods

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

`setWestBoundLongitude(minx)` Set the west bound longitude.

`setEastBoundLongitude(minx)` Set the west bound longitude.

`setSouthBoundLatitude(miny)` Set the south bound latitude.

`setNorthBoundLatitude(maxy)` Set the north bound latitude.

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

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

ISOGeographicExtent *ISOGeographicExtent*

Description

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

Format

[R6Class](#) object.

Value

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

Fields

geometricObjectType [[ISOGeometricObjectType](#)]
geometricObjectCount [[integer](#)]

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

Format

[R6Class](#) object.

Value

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

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")
```

ISOGeorectified

ISOGeorectified

Description

ISOGeorectified

Format

[R6Class](#) object.

Value

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

Fields

`checkPointAvailability` [[logical](#)]
`checkPointDescription` [[character](#)]
`cornerPoints` [[GMLPoint](#)]
`centerPoint` [[GMLPoint](#)]
`pointInPixel` [[ISOPixelOrientation](#)]
`transformationDimensionDescription` [[character](#)]
`transformationDimensionMapping` [[character](#)]

Inherited methods from [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
 setPixelOrientation(pixelOrientation) Sets the point in pixel orientation, object of class 'character' or [ISOPixelOrientation](#)

Methods

new(xml,value) This method is used to instantiate an [ISOGeorectified](#)
 setCheckPointAvailability(availability) Set checkpoint availability, object of class 'logical' (TRUE/FALSE)
 setCheckPointDescription(description, locales) Set checkpoint description
 addCornerPoint(sfg,m) Adds a corner point, either an object of class 'sfg' (from [sf](#)) or a 'matrix'
 delCornerPoint(sfg,m) Deletes a corner point, either an object of class 'sfg' (from [sf](#)) or a 'matrix'
 setCenterPoint(sfg,m) Sets a center point, either an object of class 'sfg' (from [sf](#)) or a 'matrix'
 setTransformationDimensionDescription(description, locales) Sets the transformation dimension description.
 addTransformationDimensionMapping(mapping) Adds a transformation dimension mapping
 delTransformationDimensionMapping(mapping) Deletes a transformation dimension mapping

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOGeoreferenceable *ISOGeoreferenceable*

Description

ISOGeoreferenceable

Format

[R6Class](#) object.

Value

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

Fields

controlPointAvailability [[logical](#)]
orientationParameterAvailability [[logical](#)]
orientationParameterDescription [[character](#)]
georeferencedParameters [[ISORecord](#)]
parameterCitation [[ISOCitation](#)]

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, locales) Sets the orientation parameter description. Locale names can be specified as list with the locales argument.
setGeoreferencedParameters(record) Sets the georeferenced parameter (object of class [ISORecord](#))
addParameterCitation(citation) Adds a parameter citation, object of class [ISOCitation](#)
delParameterCitation(citation) Deletes a parameter citation, object of class [ISOCitation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@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

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

Format

[R6Class](#) object.

Value

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

Fields

numberOfDimensions [[integer](#)] number of dimensions
axisDimensionProperties [[ISODimension](#)] dimension(s)
cellGeometry [[ISOCellGeometry](#)]
transformationParameterAvailability [[logical](#)]

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.blondell@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

Format

[R6Class](#) object.

Value

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

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

Fields

[citation](#) [[ISOCitation](#)] citation
[abstract](#) [[character](#)] abstract
[purpose](#) [[character](#)] purpose
[credit](#) [[character](#)] credit
[status](#) [[ISOStatus](#)] status
[pointOfContact](#) [[ISOResponsibleParty](#)] point(s) of contact
[resourceMaintenance](#) [[ISOMaintenanceInformation](#)] maintenance information
[graphicOverview](#) [[ISOBrowseGraphic](#)] graphic overview(s)
[resourceFormat](#) [[ISOFormat](#)] format(s)
[descriptiveKeywords](#) [[ISOKeywords](#)] keyword(s)
[resourceConstraints](#) [[ISOConstraints](#)] constraint(s)
[resourceSpecificUsage](#) [[ISOUsage](#)] usage(s)
[aggregationInfo](#) [[ISOAggregateInformation](#)] aggregate information

Methods

[new\(xml, element, namespace, defaults\)](#) This method is used to instantiate an [ISOIdentification](#)
[setCitation\(citation\)](#) Sets an object of class [ISOCitation](#)
[setAbstract\(abstract, locales\)](#) Sets an abstract (object of class "character"). Locale names can be specified as list with the locales argument.
[setPurpose\(purpose, locales\)](#) Sets a purpose (object of class "character"). Locale names can be specified as list with the locales argument.
[addCredit\(credit, locales\)](#) Adds a credit (object of class "character"). Locale names can be specified as list with the locales argument.
[delCredit\(credit, locales\)](#) Deletes a credit (object of class "character"). Locale names can be specified as list with the locales argument.
[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](#)
 addAggregateInformation(aggregateInfo) Adds an object of class [ISOAggregateInformation](#)
 delAggregateInformation(aggregateInfo) Deletes an object of class [ISOAggregateInformation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOIdentifier

ISOIdentifier

Description

ISOIdentifier

Format

[R6Class](#) object.

Value

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

Fields

authority [[ISOCitation](#)] authority
code [[character](#)] 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

ISOImageDescription *ISOImageDescription*

Description

ISOImageDescription

Format

[R6Class](#) object.

Value

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

Fields

illuminationElevationAngle [[numeric](#)]
illuminationAzimuthAngle [[numeric](#)]
imagingCondition [[ISOImagingCondition](#)]
imageQualityCode [[ISOMetaIdentifier](#)]
cloudCoverPercentage [[numeric](#)]
processingLevelCode [[ISOMetaIdentifier](#)]
compressionGenerationQuantity [[integer](#)]

triangulationIndicator [logical]
 radiometricCalibrationDataAvailability [logical]
 cameraCalibrationInformationAvailability [logical]
 filmDistortionInformationAvailability [logical]
 lensDistortionInformationAvailability [logical]

Inherited methods from ISOCoverageDescription

setAttributeDescription Sets the attribute description, as object of class [ISORecordType](#) or character
 setContentTypes(contentTypes) Sets the content Types, 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](#)

Methods

new(xml) This method is used to instantiate an [ISOImageDescription](#)
 setIlluminationElevationAngle(illuminationElevationAngle) Sets the illumination elevation angle
 setIlluminationAzimuthAngle(illuminationAzimuthAngle) Sets the illumination azimuth angle
 setImagingCondition(imagingCondition) Sets the imaging condition, object of class 'character' or [ISOImagingCondition](#)
 setImageQualityCode(code) Sets an image quality code
 setCloudCoverPercentage(cloudCoverPercentage) Sets the cloud cover percentage
 setProcessingLevelCode(code) Sets the processing level code
 setCompressionGenerationQuantity(quantity) Sets compression generation quantity
 setTriangulationIndicator(triangulationIndicator) Sets the triangulation indicator
 setRadiometricCalibrationDataAvailability(radiometricCalibrationDataAvailability) Sets TRUE if radiometric calibration data is available, FALSE otherwise
 setCameraCalibrationInformationAvailability(cameraCalibrationInformationAvailability) Sets TRUE if camera calibration information is available, FALSE otherwise
 setFilmDistortionInformationAvailability(filmDistortionInformationAvailability) Sets TRUE if film distortion information is available, FALSE otherwise
 setLensDistortionInformationAvailability(lensDistortionInformationAvailability) Sets TRUE if lens distortion information is available, FALSE otherwise

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#create image description
md <- ISOImageDescription$new()
md$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)
}

md$setIlluminationElevationAngle(15)
md$setIlluminationAzimuthAngle(10)
md$setImagingCondition("rain")
md$setImageQualityCode("bad")
md$setCloudCoverPercentage(90)
md$setProcessingLevelCode("high")
md$setCompressionGenerationQuantity(1L)
md$setTriangulationIndicator(FALSE)
md$setRadiometricCalibrationDataAvailability(FALSE)
md$setCameraCalibrationInformationAvailability(FALSE)
md$setFilmDistortionInformationAvailability(FALSE)
md$setLensDistortionInformationAvailability(FALSE)

xml <- md$encode()
```

ISOImageryAbstractGeolocationInformation
ISOImageryAbstractGeolocationInformation

Description

ISOImageryAbstractGeolocationInformation

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISOimagery geolocation information

Methods

`new(xml)` This method is used to instantiate an [ISOImageryAbstractGeolocationInformation](#)

Note

abstract class

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

ISOImageryAcquisitionInformation
ISOImageryAcquisitionInformation

Description

ISOImageryAcquisitionInformation

Format

[R6Class](#) object.

Value

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

Fields

instrument [list of [ISOImageryInstrument](#)]
 operation [list of [ISOImageryOperation](#)]
 platform [list of [ISOImageryPlatform](#)]
 acquisitionPlan [list of [ISOImageryPlan](#)]
 objective [list of [ISOImageryObjective](#)]
 acquisitionRequirement [list of [ISOImageryRequirement](#)]
 environmentConditions [[ISOImageryEnvironmentalRecord](#)]

Methods

new(xml) This method is used to instantiate an [ISOImageryAcquisitionInformation](#)
 addInstrument(instrument) Adds instrument, object of class [ISOImageryInstrument](#)
 delInstrument(instrument) Deletes instrument, object of class [ISOImageryInstrument](#)
 addOperation(operation) Adds instrument, object of class [ISOImageryOperation](#)
 delOperation(operation) Deletes operation, object of class [ISOImageryOperation](#)
 addPlatform(platform) Adds platform, object of class [ISOImageryPlatform](#)
 delPlatform(platform) Deletes platform, object of class [ISOImageryPlatform](#)
 addPlan(plan) Adds plan, object of class [ISOImageryPlan](#)
 delPlan(plan) Deletes plan, object of class [ISOImageryPlan](#)
 addObjective(objective) Adds objective, object of class [ISOImageryObjective](#)
 delObjective(objective) Deletes objective, object of class [ISOImageryObjective](#)
 addRequirement(requirement) Adds requirement, object of class [ISOImageryRequirement](#)
 delRequirement(requirement) Deletes requirement, object of class [ISOImageryRequirement](#)
 setEnvironmentConditions(conditions) Set environment conditions, object of class [ISOImageryEnvironmentalRecord](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – AcquisitionInformation – Part 2: Extensions for imagery and gridded data

Examples

```
md = ISOImageryAcquisitionInformation$new()
xml <- md$encode()
```

ISOImageryAlgorithm *ISOImageryAlgorithm*

Description

ISOImageryAlgorithm

Format

[R6Class](#) object.

Value

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

Fields

citation [[ISOCitation](#)]

description [[character|ISOLocalisedCharacterString](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryAlgorithm](#)

`setCitation(citation)` Set citation, object of class [ISOCitation](#)

`setDescription(description, locales)` Sets a description (object of class "character"). Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryAlgorithm$new()

#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")
rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
```



```

phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp1$setContactInfo(contact1)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)
md$setCitation(ct)
md$setDescription("some description")

xml <- md$encode()

```

ISOImageryBand

ISOImageryBand

Description

ISOImageryBand

Format

[R6Class](#) object.

Value

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

Fields

bandBoundaryDefinition [[ISOImageryBandDefinition](#)]
 nominalSpatialResolution [[numeric](#)]
 transferFunctionType [[ISOImageryTransferFunctionType](#)]
 transmittedPolarisation [[ISOImageryPolarisationOrientation](#)]
 detectedPolarisation [[ISOImageryPolarisationOrientation](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryBand](#)
`setBandBoundaryDefinition(definition)` Set the band boundary definition, object of class [ISOImageryBandDefinition](#) or character among values available by `ISOImageryBandDefinition$values()`
`setNominalSpatialResolution(resolution)` Set the nominal spatial resolution, object of class `numeric`
`setTransmittedPolarisation(polarisation)` Set the transmitted polarisation, object of class [ISOImageryPolarisationOrientation](#) or character among values returned by `ISOImageryPolarisationOrientation$values()`
`setDetectedPolarisation(polarisation)` Set the detected polarisation, object of class [ISOImageryPolarisationOrientation](#) or character among values returned by `ISOImageryPolarisationOrientation$values()`

Methods inherited from [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 <- ISOImageryBand$new()
md$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)

md$setBandBoundaryDefinition("fiftyPercent")
md$setNominalSpatialResolution(14.5)
md$setTransferFunctionType("linear")
md$setTransmittedPolarisation("horizontal")
md$setDetectedPolarisation("horizontal")

xml <- md$encode()
```

ISOImageryBandDefinition

ISOImageryBandDefinition

Description

ISOImageryBandDefinition

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO Imagery Band definition

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryBandDefinition](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
fifty <- ISOImageryBandDefinition$new(value = "fiftyPercent")
```

ISOImageryContext	<i>ISOImageryContext</i>
-------------------	--------------------------

Description

ISOImageryContext

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryContext](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
acquisition <- ISOImageryContext$new(value = "acquisition")
```

ISOImageryCoverageDescription
ISOImageryCoverageDescription

Description

ISOImageryCoverageDescription

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery image description

Fields

rangeElementdescription [[ISOImageryRangeElementDescription](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryCoverageDescription](#)

`addRangeElement(description)` Add range element description, object of class [ISOImageryRangeElementDescription](#)

`delRangeElementDescription(description)` Deletes range element description, object of class [ISOImageryRangeElementDescription](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
#create coverage description
md <- ISOImageryCoverageDescription$new()
md$setAttributeDescription("test")
md$setContenttype("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)
  }

  des <- ISOImageryRangeElementDescription$new()
  des$setName("name")
  des$setDefinition("description")
  des$addRangeElement("record1")
  des$addRangeElement("record2")
  md$addRangeElementDescription(des)
  xml <- md$encode()

```

ISOImageryCoverageResult

ISOImageryCoverageResult

Description

ISOImageryCoverageResult

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery coverage result

Fields

spatialRepresentationType [[ISOSpatialRepresentationType](#)]

resultFile [[ISODataFile](#)]

resultSpatialRepresentation [[ISOSpatialRepresentation](#)]
resultContentDescription [[ISOCoverageDescription](#)]
resultFormat [[ISOFormat](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryCoverageResult](#)
`setSpatialRepresentationType(spatialRepresentationType)` Set the spatial representation type, object of class [ISOSpatialRepresentationType](#), or 'character' among values given by `ISOSpatialRepresentationType$values()` or any free text.
`setResultFile(resultFile)` Set the result data file, object of class [ISODataFile](#).
`setResultSpatialRepresentation(spatialRepresentation)` Set the spatial representation, object of class [ISOSpatialRepresentation](#).
`setResultCoverageDescription(coverageDescription)` Set the coverage description, object of class [ISOCoverageDescription](#)
`setResultFormat(format)` Set the result format, object of class [ISOFormat](#)

Methods inherited from [ISOAbstractResult](#)

See methods description at [ISOAbstractResult](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

ISOImageryEnvironmentalRecord

ISOImageryEnvironmentalRecord

Description

ISOImageryEnvironmentalRecord

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery environmental record

Fields

averageAirTemperature [numeric]
 maxRelativeHumidity [numeric]
 maxAltitude [numeric]
 meteorologicalConditions [ISOBaseCharacterString|ISOLocalisedCharacterString]

Methods

new(xml) This method is used to instantiate an [ISOImageryEnvironmentalRecord](#)
 setAverageAirTemperature(temperature) Set the average air temperature
 setMaxRelativeHumidity(humidity) Set the max relative humidity
 setMaxAltitude(altitude) Set the max altitude
 setMeteorologicalConditions(conditions) Set the meteorological conditions. Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryEnvironmentalRecord$new()
md$setAverageAirTemperature(3)
md$setMaxRelativeHumidity(67)
md$setMaxAltitude(400)
md$setMeteorologicalConditions("some conditions")
xml <- md$encode()
```

ISOImageryEvent

ISOImageryEvent

Description

ISOImageryEvent

Format

[R6Class](#) object.

Value

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

Fields

identifier [[ISOMetaIdentifier](#)]
trigger [[ISOImageryTrigger](#)]
context [[ISOImageryContext](#)]
sequence [[ISOImagerySequence](#)]
time [[POSIXt](#)]
relatedPass [[ISOImageryPlatformPass](#)]
relatedSensor [list of [ISOImageryInstrument](#)]
expectedObjective [list of [ISOImageryObjective](#)]

Methods

new(xml) This method is used to instantiate an [ISOImageryEvent](#)
setIdentifier(identifier) Sets an identifier, object of class character or [ISOMetaIdentifier](#)
setTrigger(trigger) Set the trigger, object of class [ISOImageryTrigger](#), or 'character' among values given by [ISOImageryTrigger\\$values\(\)](#), or free text.
setContext(context) Set the context, object of class [ISOImageryContext](#), or 'character' among values given by [ISOImageryContext\\$values\(\)](#), or free text.
setSequence(sequence) Set the sequence, object of class [ISOImagerySequence](#), or 'character' among values given by [ISOImagerySequence\\$values\(\)](#), or free text.
setTime(time) Set the time, object of class [POSIXt](#).
setPlatformPass(platformPass) Set the platform pass, object of class [ISOImageryPlatformPass](#)
addSensor(sensor) Adds a sensor, object of class [ISOImageryInstrument](#).
delSensor(sensor) Deletes a sensor, object of class [ISOImageryInstrument](#)
addObjective(objective) Adds an objective, object of class [ISOImageryObjective](#)
delObjective(objective) Deletes an objective, object of class [ISOImageryObjective](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryEvent$new()
md$setIdentifier("event_1")
md$setTrigger("manual")
md$setContext("pass")
md$setSequence("instantaneous")
md$setTime(Sys.time())

xml <- md$encode()
```

ISOImageryGCP

ISOImageryGCPCollection

Description

ISOImageryGCPCollection

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery gcp collection

Methods

`new(xml)` This method is used to instantiate an [ISOImageryGCPCollection](#)

`setCollectionIdentification(id)` Set the identifier, object of class integer

`setCollectionName(name, locales)` Sets a name (object of class "character"). Locale names can be specified as list with the locales argument.

`setCoordinateReferenceSystem(crs)` Sets the crs, object of class [ISOReferenceSystem](#)

`addGCP(gcp)` Adds a GCP, object of class [ISOImageryGCP](#)

`delGCP(gcp)` Deletes a GCP, object of class [ISOImageryGCP](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryGCP$new()
require(sf)
pt <- sf::st_point(c(1,1))
md$setGeographicCoordinates(sfg = pt)
xml <- md$encode()
```

ISOImageryGCPCollection

ISOImageryGCPCollection

Description

ISOImageryGCPCollection

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery gcp collection

Methods

`new(xml)` This method is used to instantiate an [ISOImageryGCPCollection](#)

`setCollectionIdentification(id)` Set the identifier, object of class integer

`setCollectionName(name, locales)` Sets a name (object of class "character"). Locale names can be specified as list with the locales argument.

`setCoordinateReferenceSystem(crs)` Sets the crs, object of class [ISOReferenceSystem](#)

`addGCP(gcp)` Adds a GCP, object of class [ISOImageryGCP](#)

`delGCP(gcp)` Deletes a GCP, object of class [ISOImageryGCP](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryGCPCollection$new()
md$setCollectionIdentification(1L)
md$setCollectionName("name")
rs <- ISOReferenceSystem$new()
rsId <- ISOReferenceIdentifier$new(code = "4326", codeSpace = "EPSG")
rs$setReferenceSystemIdentifier(rsId)
md$setCoordinateReferenceSystem(rs)
xml <- md$encode()
```

ISOImageryGeometryType

ISOImageryGeometryType

Description

ISOImageryGeometryType

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO Imagery geometry type

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryGeometryType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
point <- ISOImageryGeometryType$new(value = "point")
```

ISOImageryGeorectified

ISOImageryGeorectified

Description

ISOImageryGeorectified

Format

[R6Class](#) object.

Value

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

Fields

checkPoint [list of [ISOImageryGCP](#)]

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryGeorectified](#)

`addCheckPoint(sfg, m)` Adds a check point, either an object of class 'sfg' (from **sf**) or a 'matrix'

`delCheckPoint(sfg, m)` Deletes a check point, either an object of class 'sfg' (from **sf**) or a 'matrix'

Methods inherited from [ISOGridSpatialRepresentation](#)

See [ISOGridSpatialRepresentation](#)

Methods inherited from [ISOGeorectified](#)

See [ISOGeorectified](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata – Part 2: Extensions for imagery and gridded data

ISOImageryGeoreferenceable

ISOImageryGeoreferenceable

Description

ISOImageryGeoreferenceable

Format

[R6Class](#) object.

Value

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

Fields

geolocationInformation [list of [ISOImageryAbstractGeolocationInformation](#)]

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryGeoreferenceable](#)

`addGeolocationInformation(geolocationInformation)` Adds a geolocation information, an object of class [ISOImageryAbstractGeolocationInformation](#)

`delGeolocationInformation(geolocationInformation)` Deletes geolocation information, an object of class [ISOImageryAbstractGeolocationInformation](#)

Methods inherited from [ISOGridSpatialRepresentation](#)

See [ISOGridSpatialRepresentation](#)

Methods inherited from [ISOGeoreferenceable](#)

See [ISOGeoreferenceable](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata – Part 2: Extensions for imagery and gridded data

ISOImageryImageDescription
ISOImageryImageDescription

Description

ISOImageryImageDescription

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery image description

Fields

rangeElementdescription [[ISOImageryRangeElementDescription](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryImageDescription](#)

`addRangeElement(description)` Add range element description, object of class [ISOImageryRangeElementDescription](#)

`delRangeElementDescription(description)` Deletes range element description, object of class [ISOImageryRangeElementDescription](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
#create image description
md <- ISOImageryImageDescription$new()
md$setAttributeDescription("test")
md$setContenttype("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)
}

md$setIlluminationElevationAngle(15)
md$setIlluminationAzimuthAngle(10)
md$setImagingCondition("rain")
md$setImageQualityCode("bad")
md$setCloudCoverPercentage(90)
md$setProcessingLevelCode("high")
md$setCompressionGenerationQuantity(1L)
md$setTriangulationIndicator(FALSE)
md$setRadiometricCalibrationDataAvailability(FALSE)
md$setCameraCalibrationInformationAvailability(FALSE)
md$setFilmDistortionInformationAvailability(FALSE)
md$setLensDistortionInformationAvailability(FALSE)

des <- ISOImageryRangeElementDescription$new()
des$setName("name")
des$setDefinition("description")
des$addRangeElement("record1")
des$addRangeElement("record2")
md$addRangeElementDescription(des)
xml <- md$encode()

```

ISOImageryInstrument *ISOImageryPlatform*

Description

ISOImageryPlatform

Format

R6Class object.

Value

Object of R6Class for modelling an ISO imagery platform

Fields

citation [list of ISOCitation]

identifier [ISOMetaIdentifier]

type [character|ISOLocalisedCharacterString]

description [character|ISOLocalisedCharacterString]

mountedOn [list of ISOImageryPlatform]

Methods

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

addCitation(citation) Adds citation, object of class ISOCitation

delCitation(citation) Deletes a citation, object of class ISOCitation

setIdentifier(identifier) Sets an identifier, object of class character or ISOMetaIdentifier

setType(type, locales) Sets a type (object of class "character"). Locale names can be specified as list with the locales argument.

setDescription(description, locales) Sets a description (object of class "character"). Locale names can be specified as list with the locales argument.

addPlatform(platform) Add a platform, object of class ISOImageryPlatform

delPlatform(platform) Deletes a platform, object of class ISOImageryPlatform

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryInstrument$new()
md$setIdentifier("identifier")
md$setType("type")
md$setDescription("description")
xml <- md$encode()
```

ISOImageryMetadata *ISOImageryMetadata*

Description

ISOImageryMetadata

Format

[R6Class](#) object.

Value

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

Fields

acquisitionInformation [list of [ISOImageryAcquisitionInformation](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryMetadata](#)

`addAcquisitionInfo(acquisitionInfo)` Add acquisition information, object of class [ISOImageryAcquisitionInformation](#)

`delAcquisitionInfo(acquisitionInfo)` Deletes acquisition information, object of class [ISOImageryAcquisitionInformation](#)

Methods inherited from of 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

addLocale(locale) Adds a locale, object of class ISOLocale
 delLocale(locale) Deletes a locale, object of class ISOLocale
 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.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata – Part 2: Extensions for imagery and gridded data

Examples

```

#example 1 - WRITE: Create an ISO metadata and encode it as XML
#####
md = ISOImageryMetadata$new()
md$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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(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)

  #add an INSPIRE spatial data theme?
  inspire_kwd <- ISOKeywords$new()
  anc1 <- ISOAnchor$new(
    name = "Environmental monitoring facilities",
    href = "http://inspire.ec.europa.eu/theme/ef"
  )

```

```

inspire_kwd$addKeyword(anc1)
inspire_kwd$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle(
  ISOAnchor$new(
    name = "GEMET - INSPIRE themes, version 1.0",
    href="http://www.eionet.europa.eu/gemet/inspire_themes"
  )
)
inspire_date <- ISODate$new()
inspire_date$setDate(as.Date("2008-06-01"))
inspire_date$setDateType("publication")
th$addDate(inspire_date)
inspire_kwd$setThesaurusName(th)
ident$addKeywords(inspire_kwd)

#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)

#create dataQuality object with a 'dataset' scope
dq <- ISODataQuality$new()
scope <- ISOScope$new()
scope$setLevel("dataset")
dq$setScope(scope)

#add data quality reports...

#add a report the data quality
dc <- ISODomainConsistency$new()
result <- ISOConformanceResult$new()
spec <- ISOCitation$new()
spec$setTitle("Data Quality check")
spec$setAlternateTitle("This is is some data quality check report")
d <- ISODate$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 INSPIRE reports?
#INSPIRE - interoperability of spatial data sets and services
dc_inspire1 <- ISODomainConsistency$new()
cr_inspire1 <- ISOConformanceResult$new()
cr_inspire_spec1 <- ISOCitation$new()
cr_title1 <- paste(
  "Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards interoperability of spatial data",
  "sets and services"
)
cr_inspire_spec1$setTitle(cr_title1)
cr_inspire1$setExplanation("See the referenced specification")
cr_inspire_date1 <- ISODate$new()
cr_inspire_date1$setDate(ISOdate(2010,12,8))
cr_inspire_date1$setDateType("publication")
cr_inspire_spec1$addDate(cr_inspire_date1)
cr_inspire1$setSpecification(cr_inspire_spec1)
cr_inspire1$setPass(TRUE)
dc_inspire1$addResult(cr_inspire1)
dq$addReport(dc_inspire1)
#INSPIRE - metadata
dc_inspire2 <- ISODomainConsistency$new()
cr_inspire2 <- ISOConformanceResult$new()
cr_inspire_spec2 <- ISOCitation$new()
cr_title2 <- paste(
  "COMMISSION REGULATION (EC) No 1205/2008 of 3 December 2008 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards metadata"
)
cr_inspire_spec2$setTitle(cr_title2)
cr_inspire2$setExplanation("See the referenced specification")
cr_inspire_date2 <- ISODate$new()
cr_inspire_date2$setDate(ISOdate(2008,12,4))
cr_inspire_date2$setDateType("publication")
cr_inspire_spec2$addDate(cr_inspire_date2)
cr_inspire2$setSpecification(cr_inspire_spec2)
cr_inspire2$setPass(TRUE)
dc_inspire2$addResult(cr_inspire2)
dq$addReport(dc_inspire2)

#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$new()
cit$setTitle("sometitle")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
cit$addDate(d)
cit$setEdition("1.0")
cit$setEditionDate(as.Date(ISOdate(2015, 1, 1, 1)))
contact = ISOContact$new()
fcLink <- ISOOnlineResource$new()
fcLink$setLinkage("http://somedlink/featurecatalogue")
contact$setOnlineResource(fcLink)
rp = ISOResponsibleParty$new()
rp$setRole("publisher")
rp$setContactInfo(contact)
cit$addCitedResponsibleParty(rp)
fcd$addFeatureCatalogueCitation(cit)
md$addContentInfo(fcd)

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

#example 2 - READ: Create an ISO imagery metadata reading from XML
#####

require(XML)
xmlfile <- system.file("extdata/examples", "metadata.xml", package = "geometa")
xml <- xmlParse(xmlfile)
md <- ISOImageryMetadata$new(xml = xml)

```

ISOImageryNominalResolution

ISOImageryNominalResolution

Description

ISOImageryNominalResolution

Format

R6Class object.

Value

Object of R6Class for modelling an ISO imagery nominal resolution

Fields

scanningResolution [ISODistance]

groundResolution [ISODistance]

Methods

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

setScanningResolution(resolution) Set the scanning resolution, object of class ISODistance

setGroundResolution(resolution) Set the ground resolution, object of class ISODistance

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
#encoding
dq <- ISOImageryNominalResolution$new()
d <- ISODistance$new(value = 1, uom = "m", useUomURI = TRUE)
dq$setScanningResolution(d)
dq$setGroundResolution(d)

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

ISOImageryObjective *ISOImageryObjective*

Description

ISOImageryObjective

Format

[R6Class](#) object.

Value

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

Fields

identifier [[ISOMetaIdentifier](#)]
 priority [[character](#)|[ISOLocalisedCharacterString](#)]
 type [list of [ISOImageryObjectiveType](#)]
 function [list of [character](#)|[ISOLocalisedCharacterString](#)]
 extent [list of [ISOExtent](#)]
 sensingInstrument [list of [ISOImageryInstrument](#)]
 pass [list of [ISOImageryPlatformPass](#)]
 objectiveOccurance [list of [ISOImageryEvent](#)]

Methods

new(xml) This method is used to instantiate an [ISOImageryObjective](#)
 setIdentifier(identifier) Sets an identifier, object of class [character](#) or [ISOMetaIdentifier](#)
 setPriority(priority, locales) Sets a priority (object of class "character"). Locale names can be specified as list with the locales argument.
 addType(type) Add type, object of class [ISOImageryObjectiveType](#) or 'character' among values given by [ISOImageryObjectiveType\\$values\(\)](#) or any free text.
 delType(type) Deletes type, object of class [ISOImageryObjectiveType](#) or 'character' among values given by [ISOImageryObjectiveType\\$values\(\)](#) or any free text.
 addFunction(fun, locales) Adds a function (object of class "character"). Locale names can be specified as list with the locales argument.
 delFunction(fun, locales) Deletes a function (object of class "character"). Locale names can be specified as list with the locales argument.
 addExtent(extent) Adds an extent, object of class [ISOExtent](#)
 delExtent(extent) Deletes an extent, object of class [ISOExtent](#)

addSensingInstrument(instrument) Adds an instrument, object of class [ISOImageryInstrument](#)
 delSensingInstrument(instrument) Deletes an instrument, object of class [ISOImageryInstrument](#)
 addPlatformPass(pass) Adds an platform pass, object of class [ISOImageryPlatformPass](#)
 delPlatformPass(pass) Deletes an platform pass, object of class [ISOImageryPlatformPass](#)
 addObjectiveOccurance(event) Adds an objective occurrence, object of class [ISOImageryEvent](#)
 delObjectiveOccurance(event) Deletes an objective occurrence, object of class [ISOImageryEvent](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```

#encoding
md <- ISOImageryObjective$new()
md$setIdentifier("identifier")
md$setPriority("urgent")
md$addType("survey")
md$addFunction("my_function")
evt <- ISOImageryEvent$new()
evt$setIdentifier("event_1")
evt$setTrigger("manual")
evt$setContext("pass")
evt$setSequence("instantaneous")
evt$setTime(Sys.time())
md$addObjectiveOccurance(evt)
extent <- ISOExtent$new()
bbox <- ISOGeographicBoundingBox$new(minx = -180, miny = -90, maxx = 180, maxy = 90)
extent$setGeographicElement(bbox)
time <- ISOTemporalExtent$new()
start <- ISOdate(2000, 1, 12, 12, 59, 45)
end <- ISOdate(2010, 8, 22, 13, 12, 43)
tp <- GMLTimePeriod$new(beginPosition = start, endPosition = end)
time$setTimePeriod(tp)
extent$setTemporalElement(time)
vert <- ISOVerticalExtent$new()
vert$setMinimumValue(0)
vert$setMaximumValue(19)
extent$setVerticalElement(vert)
md$addExtent(extent)
md$sensingInstrument = NA
md$pass = NA
xml <- md$encode()

```

ISOImageryObjectiveType
ISOImageryObjectiveType

Description

ISOImageryObjectiveType

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryObjectiveType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
survey <- ISOImageryObjectiveType$new(value = "survey")
```

ISOImageryOperation *ISOImageryOperation*

Description

ISOImageryOperation

Format

[R6Class](#) object.

Value

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

Fields

description [[character|ISOLocalisedCharacterString](#)]
 citation [[ISOCitation](#)]
 identifier [[ISOMetaIdentifier](#)]
 status [[ISOStatus](#)]
 type [[ISOImageryOperationType](#)]
 parentOperation [[ISOImageryOperation](#)]
 childOperation [list of [ISOImageryOperation](#)]
 platform [list of [ISOImageryPlatform](#)]
 objective [list of [ISOImageryObjective](#)]
 plan [list of [ISOImageryPlan](#)]
 significantEvent [list of [ISOImageryEvent](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryOperation](#)

`setDescription(description, locales)` Sets a description (object of class "character"). Locale names can be specified as list with the locales argument.

`setCitation(citation)` Sets a citation, object of class [ISOCitation](#)

`setIdentifier(identifier)` Sets an identifier, object of class character or [ISOMetaIdentifier](#)

`setStatus(status)` Sets status, object of class [ISOStatus](#) or 'character', among values given by `ISOStatus$values()` or any free text code

`setType(type)` Sets the operation type, object of class [ISOImageryOperationType](#), or 'character' among values given by `ISOImageryOperationType$values()`, or any free text code.

`setParentOperation(operation)` Sets the parent operation, object of class [ISOImageryOperation](#).

`addChildOperation(operation)` Adds a child operation, object of class [ISOImageryOperation](#)

delChildOperation(operation) Deletes a child operation, object of class [ISOImageryOperation](#)
addPlatform(platform) Adds a platform, object of class [ISOImageryPlatform](#)
delPlatform(platform) Deletes a platform, object of class [ISOImageryPlatform](#)
addObjective(objective) Adds an objective object of class [ISOImageryObjective](#)
delObjective(objective) Deletes an objective, object of class [ISOImageryObjective](#)
setPlan(plan) Sets a plan, object of class [ISOImageryPlan](#)
addSignificantEvent(event) Adds a significant event, object of class [ISOImageryEvent](#)
delSignificantEvent(event) Deletes a significant event, object of class [ISOImageryEvent](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

ISOImageryOperationType
ISOImageryOperationType

Description

ISOImageryOperationType

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO Imagery Operation type

Methods

new(xml, value) This method is used to instantiate an [ISOImageryOperationType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
real <- ISOImageryOperationType$new(value = "real")
```

ISOImageryPlan	<i>ISOImageryPlan</i>
----------------	-----------------------

Description

ISOImageryPlan

Format

[R6Class](#) object.

Value

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

Fields

type [[ISOImageryGeometryType](#)]
 citation [[ISOCitation](#)]
 operation [list of [ISOImageryOperation](#)]
 satisfiedPlan [list of [ISOImageryPlan](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryPlan](#)

`setType(type)` Set the imagery geometry type, object of class [ISOImageryGeometryType](#) or 'character' among values given by [ISOImageryGeometryType\\$values\(\)](#).

`setStatus(status)` Set the imagery plan status, object of class [ISOStatus](#) or 'character' among values given by [ISOStatus\\$values\(\)](#).

`setCitation(citation)` Sets the citation, object of class [ISOCitation](#)

`addOperation(operation)` Adds a operation, object of class [ISOImageryOperation](#)

`delOperation(operation)` Deletes a operation, object of class [ISOImageryOperation](#)

`addSatisfiedRequirement(plan)` Add a satisfied plan, object of class [ISOImageryPlan](#)

`delSatisfiedRequirement(plan)` Deletes a satisfied plan, object of class [ISOImageryPlan](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryPlan$new()
md$setType("point")
md$setStatus("completed")

#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")
rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp1$setContactInfo(contact1)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)
md$setCitation(ct)
xml <- md$encode()
```

ISOImageryPlatform *ISOImageryPlatform*

Description

ISOImageryPlatform

Format

[R6Class](#) object.

Value

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

Fields

citation [list of [ISOCitation](#)]

identifier [[ISOMetaIdentifier](#)]

description [[character](#)|[ISOLocalisedCharacterString](#)]

sponsor [list of [ISOResponsibleParty](#)]

instrument [list of [ISOImageryInstrument](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryPlatform](#)

`addCitation(citation)` Adds citation, object of class [ISOCitation](#)

`delCitation(citation)` Deletes a citation, object of class [ISOCitation](#)

`setIdentifier(identifier)` Sets an identifier, object of class [character](#) or [ISOMetaIdentifier](#)

`setDescription(description, locales)` Sets a description (object of class "character"). Locale names can be specified as list with the locales argument.

`addSponsor(sponsor)` Adds a sponsor, object of class [ISOResponsibleParty](#)

`delSponsor(sponsor)` Deletes a sponsor, object of class [ISOResponsibleParty](#)

`addInstrument(instrument)` Add a instrument, object of class [ISOImageryInstrument](#)

`delInstrument(instrument)` Deletes a instrument, object of class [ISOImageryInstrument](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryPlatform$new()

#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")
rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp1$setContactInfo(contact1)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)
md$addCitation(ct)

md$setIdentifier("identifier")
md$setDescription("some description")

xml <- md$encode()
```

ISOImageryPlatformPass

ISOImageryPlatformPass

Description

ISOImageryPlatformPass

Format

[R6Class](#) object.

Value

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

Fields

identifier [[ISOMetaIdentifier](#)]

extent [?]

relatedEvent [list of [ISOImageryEvent](#)]

Methods

new(xml) This method is used to instantiate an [ISOImageryPlatformPass](#)

setIdentifier(identifier) Sets an identifier, object of class character or [ISOMetaIdentifier](#)

setExtent(extent) Set the extent

addEvent(event) Add a event, object of class [ISOImageryEvent](#)

delEvent(event) Deletes a event, object of class [ISOImageryEvent](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryPlatformPass$new()
md$setIdentifier("identifier")

require(sf)
outer = matrix(c(0,0,10,0,10,10,0,10,0,0),ncol=2, byrow=TRUE)
hole1 = matrix(c(1,1,1,2,2,2,2,1,1,1),ncol=2, byrow=TRUE)
hole2 = matrix(c(5,5,5,6,6,6,6,5,5,5),ncol=2, byrow=TRUE)
pts = list(outer, hole1, hole2)
pl = st_polygon(pts)
md$setExtent(pl)

xml <- md$encode()
```

ISOImageryPolarisationOrientation

ISOImageryPolarisationOrientation

Description

ISOImageryPolarisationOrientation

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO Imagery Polarisation orientation

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryPolarisationOrientation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
h <- ISOImageryPolarisationOrientation$new(value = "horizontal")
```

ISOImageryPriority *ISOImageryPriority*

Description

ISOImageryPriority

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryPriority](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
highImp <- ISOImageryPriority$new(value = "highImportance")
```

ISOImageryProcessing *ISOImageryProcessing*

Description

ISOImageryProcessing

Format

[R6Class](#) object.

Value

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

Fields

identifier [[ISOMetaIdentifier](#)]
softwareReference [list of [ISOCitation](#)]
procedureDescription [[character](#)|[ISOLocalisedCharacterString](#)]
documentation [list of [ISOCitation](#)]
runTimeParameters [[character](#)]
algorithm [list of [ISOImageryAlgorithm](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryProcessing](#)
`setIdentifier(identifier)` Sets an identifier, object of class [character](#) or [ISOMetaIdentifier](#)
`addSoftwareReference(softwareReference)` Adds software reference, object of class [ISOCitation](#)
`delSoftwareReference(softwareReference)` Deletes a software reference, object of class [ISOCitation](#)
`setProcedureDescription(description, locales)` Sets a description (object of class "[character](#)"). Locale names can be specified as `list` with the `locales` argument.
`addDocumentation(documentation)` Adds documentation, object of class [ISOCitation](#)
`delDocumentation(documentation)` Deletes documentation, object of class [ISOCitation](#)
`setRunTimeParameters(params)` Set the runtime parameters, object of class [character](#)
`addAlgorithm(algorithm)` Add a algorithm, object of class [ISOImageryAlgorithm](#)
`delAlgorithm(algorithm)` Deletes a algorithm, object of class [ISOImageryAlgorithm](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryProcessing$new()

#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")
rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp1$setContactInfo(contact1)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)

md$setIdentifier("identifier")
md$setProcedureDescription("some description")
md$addSoftwareReference(ct)
md$addDocumentation(ct)
md$setRunTimeParameters("params")

xml <- md$encode()
```

ISOImageryProcessStep *ISOImageryProcessStep*

Description

ISOImageryProcessStep

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery process step

Fields

processingInformation [[ISOImageryProcessing](#)]

output [list of [ISOImagerySource](#)]

report [list of [ISOImageryProcessStepReport](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryProcessStep](#)

`setProcessingInformation(processingInfo)` Set the processing information, object of class [ISOImageryProcessing](#)

`addOutput(output)` Add output, object of class [ISOSource](#)

`delOutput(output)` Deletes output, object of class [ISOSource](#)

`addReport(report)` Add report, object of class [ISOImageryProcessStepReport](#)

`delReport(report)` Deletes report, object of class [ISOImageryProcessStepReport](#)

Methods inherited from [ISOProcessStep](#)

See methods description at [ISOProcessStep](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```

ps <- ISOImageryProcessStep$new()
ps$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)

#specific methods to ISO 19115-2
process <- ISOImageryProcessing$new()

#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")
rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp1$setContactInfo(contact1)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)

process$setIdentifier("identifier")
process$setProcedureDescription("some description")
process$addSoftwareReference(ct)

```

```
process$addDocumentation(ct)
process$setRunTimeParameters("params")
ps$setProcessingInformation(process)

#output
trg <- ISOImagerySource$new()
trg$setProcessedLevel("level")
res <- ISOImageryNominalResolution$new()
d <- ISODistance$new(value = 1, uom = "m", useUomURI = TRUE)
res$setScanningResolution(d)
trg$setResolution(res)
ps$addOutput(trg)

#report
rep <- ISOImageryProcessStepReport$new()
rep$setName("report")
rep$setDescription("description")
rep$setFileType("filetype")
ps$addReport(rep)

xml <- ps$encode()
```

ISOImageryProcessStepReport
ISOImageryProcessStepReport

Description

ISOImageryProcessStepReport

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)|[ISOLocalisedCharacterString](#)]

description [[character](#)|[ISOLocalisedCharacterString](#)]

fileType [[character](#)|[ISOLocalisedCharacterString](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryProcessStepReport](#)

`setName(name, locales)` Sets a name (object of class "character"). Locale names can be specified as list with the locales argument.

`setDescription(description, locales)` Sets a description (object of class "character"). Locale names can be specified as list with the locales argument.

`setFileType(fileType, locales)` Sets a fileType (object of class "character"). Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImageryProcessStepReport$new()
md$setName("my_report")
md$setDescription("description")
md$setFileType("md")
xml <- md$encode()
```

ISOImageryRangeElementDescription

ISOImageryRangeElementDescription

Description

ISOImageryRangeElementDescription

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)]
description [[character](#)]
rangeElement [[ISORecord](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImageryRangeElementDescription](#)

`setName(name, locales)` Sets the name. Locale names can be specified as list with the locales argument.

`setDefinition(description, locales)` Sets the description. Locale names can be specified as list with the locales argument.

`addRangeElement(record)` Add range element, object of class [ISORecord](#)

`delRangeElement(record)` Deletes range element, object of class [ISORecord](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
#create object
md <- ISOImageryRangeElementDescription$new()
md$setName("name")
md$setDefinition("description")
md$addRangeElement("record1")
md$addRangeElement("record2")
xml <- md$encode()
```

ISOImageryRequestedDate

ISOImageryRequestedDate

Description

ISOImageryRequestedDate

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery requested date

Fields

requestedDateOfCollection [[POSIXt](#)]
latestAcceptableDate [[POSIXt](#)]

Methods

new(xml) This method is used to instantiate an [ISOImageryRequestedDate](#)
setRequestedDateOfCollection(date) Set the requested date of collection, object of class [POSIXt](#)
setLatestAcceptableDate(date) Set the latest acceptable date, object of class [POSIXt](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
#create band range dimension
md <- ISOImageryRequestedDate$new()
md$setRequestedDateOfCollection(Sys.time())
md$setLatestAcceptableDate(Sys.time())
xml <- md$encode()
```

ISOImageryRequirement *ISOImageryRequirement*

Description

ISOImageryRequirement

Format

[R6Class](#) object.

Value

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

Fields

citation [ISOCitation]
 identifier [ISOMetaIdentifier]
 requestor [list of ISOResponsibleParty]
 recipient [list of ISOResponsibleParty]
 priority [ISOImageryPriority]
 requestedDate [ISOImageryRequestedDate]
 expiryDate [POSIXt]
 satisfiedPlan [list of ISOImageryPlan]

Methods

new(xml) This method is used to instantiate an *ISOImageryRequirement*
 setCitation(citation) Sets the citation, object of class *ISOCitation*
 setIdentifier(identifier) Sets an identifier, object of class character or *ISOMetaIdentifier*
 addRequestor(requestor) Adds a requestor, object of class *ISOResponsibleParty*
 delRequestor(requestor) Deletes a requestor, object of class *ISOResponsibleParty*
 addRecipient(recipient) Adds a recipient, object of class *ISOResponsibleParty*
 delRecipient(recipient) Deletes a recipient, object of class *ISOResponsibleParty*
 setPriority(priority) Set the priority, object of class *ISOImageryPriority*, or an object of class 'character' among values given by *ISOImageryPriority*\$values().
 setRequestedDate(date) Set requested date, object of class *ISOImageryRequestedDate*
 setExpiryDate(date) Set expiry date, object of class *POSIXt*
 addSatisfiedPlan(plan) Add a satisfied plan, object of class *ISOImageryPlan*
 delSatisfiedPlan(plan) Deletes a satisfied plan, object of class *ISOImageryPlan*

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```

md <- ISOImageryRequirement$new()
md$setIdentifier("identifier")
#add citation
rp1 <- ISOResponsibleParty$new()
rp1$setIndividualName("someone1")
rp1$setOrganisationName("somewhere1")
rp1$setPositionName("someposition1")

```

```

rp1$setRole("pointOfContact")
contact1 <- ISOContact$new()
phone1 <- ISOTelephone$new()
phone1$setVoice("myphonenumber1")
phone1$setFacsimile("myfacsimile1")
contact1$setPhone(phone1)
address1 <- ISOAddress$new()
address1$setDeliveryPoint("theaddress1")
address1$setCity("thecity1")
address1$setPostalCode("111")
address1$setCountry("France")
address1$setEmail("someone1@theorg.org")
contact1$setAddress(address1)
res <- ISOOnlineResource$new()
res$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somename")
contact1$setOnlineResource(res)
rp2 <- ISOResponsibleParty$new()
rp2$setIndividualName("someone2")
rp2$setOrganisationName("somewhere2")
rp2$setPositionName("someposition2")
rp2$setRole("pointOfContact")
contact2 <- ISOContact$new()
phone2 <- ISOTelephone$new()
phone2$setVoice("myphonenumber2")
phone2$setFacsimile("myfacsimile2")
contact1$setPhone(phone2)
address2 <- ISOAddress$new()
address2$setDeliveryPoint("theaddress2")
address2$setCity("thecity2")
address2$setPostalCode("111")
address2$setCountry("France")
address2$setEmail("someone2@theorg.org")
contact2$setAddress(address2)
contact2$setOnlineResource(res)
rp2$setContactInfo(contact2)

#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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp1)
md$setCitation(ct)
md$addRequestor(rp1)
md$addRecipient(rp2)
md$setPriority("highImportance")

```



```
rd <- ISOImageryRequestedDate$new()
rd$setRequestedDateOfCollection(Sys.time())
rd$setLatestAcceptableDate(Sys.time())
md$setRequestedDate(rd)
md$setExpiryDate(Sys.time())
xml <- md$encode()
```

ISOImagerySensorType *ISOImagerySensorType*

Description

ISOImagerySensorType

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling an ISO imagery sensor type

Methods

`new(xml, value)` This method is used to instantiate an [ISOImagerySensorType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImagerySensorType$new()
```

ISOImagerySequence *ISOImagerySequence*

Description

ISOImagerySequence

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOImagerySequence](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
inst <- ISOImagerySequence$new(value = "instantaneous")
```

ISOImagerySource *ISOImagerySource*

Description

ISOImagerySource

Format

[R6Class](#) object.

Value

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

Fields

processedLevel [[ISOMetaIdentifier](#)]
resolution [[ISOImageryNominalResolution](#)]

Methods

`new(xml)` This method is used to instantiate an [ISOImagerySource](#)
`setProcessedLevel(processedLevel)` Sets processed level, object of class character or [ISOMetaIdentifier](#)
`setResolution(resolution)` Set the resolution, object of class [ISOImageryNominalResolution](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

```
md <- ISOImagerySource$new()
md$setProcessedLevel("identifier")
res <- ISOImageryNominalResolution$new()
d <- ISODistance$new(value = 1, uom = "m", useUomURI = TRUE)
res$setScanningResolution(d)
md$setResolution(res)

xml <- md$encode()
```

ISOImageryTransferFunctionType

ISOImageryTransferFunctionType

Description

ISOImageryTransferFunctionType

Format

R6Class object.

Value

Object of R6Class for modelling an ISO imagery transfer function type

Methodsnew(xml, value) This method is used to instantiate an [ISOImageryTransferFunctionType](#)**Author(s)**

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
log <- ISOImageryTransferFunctionType$new(value = "logarithmic")
```

ISOImageryTrigger	<i>ISOImageryTrigger</i>
-------------------	--------------------------

Description

ISOImageryTrigger

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value)` This method is used to instantiate an [ISOImageryTrigger](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

Examples

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

#some def
auto <- ISOImageryTrigger$new(value = "automatic")
```

ISOImageryUsability *ISOImageryUsability*

Description

ISOImageryUsability

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOImageryUsability](#)

Methods inherited from [ISODataQualityAbstractElement](#)

See methods description at [ISODataQualityAbstractElement](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115-2:2009 - Geographic information – Metadata Part 2: Extensions for imagery and gridded data

ISOImagingCondition *ISOImagingCondition*

Description

ISOImagingCondition

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISOImagingCondition](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#ImagingCondition
ImagingCondition <- ISOImagingCondition$new(value = "rain")
```

ISOInheritanceRelation

ISOInheritanceRelation

Description

ISOInheritanceRelation

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)] name
description [[character](#)] description
uniqueInstance [[logical](#)]
subtype [[ISOFeatureType](#)] subtype
supertype [[ISOFeatureType](#)] supertype

Methods

new(xml, defaults) This method is used to instantiate an [ISOInheritanceRelation](#)
 setName(name, locales) Set name of inheritance relation. Locale names can be specified as list with the locales argument.
 setDescription(description, locales) Set description of inheritance relation. Locale names can be specified as list with the locales argument.
 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

ISOInitiative

ISOInitiative

Description

ISOInitiative

Format

[R6Class](#) object.

Value

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

Methods

new(xml) This method is used to instantiate an [ISOInitiative](#)

Inherited methods from ISOAbstractAggregate

addComposedOf(composedOf) Adds a dataset, object of class ISODataset
 delComposedOf(composedOf) Deletes a dataset, object of class ISODataset
 addSeriesMetadata(metadata) Adds a series metadata, object of class ISOMetadata
 delSeriesMetadata(metadata) Deletes a series metadata, object of class ISOMetadata
 addSubset(subset) Adds a subset, object that inherits from ISOAbstractAggregate
 delSubset(subset) Deletes a subset, object that inherits from ISOAbstractAggregate
 addSuperset(superset) Adds a superset, object that inherits from ISOAbstractAggregate
 delSuperset(superset) Deletes a superset, object that inherits from ISOAbstractAggregate

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOInitiativeType *ISOInitiativeType*

Description

ISOInitiativeType

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOInitiativeType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#geomOnly
geomOnly <- ISOInitiativeType$new(value = "campaign")
```

ISOKeywords

ISOKeywords

Description

ISOKeywords

Format

R6Class object.

Value

Object of R6Class for modelling a ISO set of keywords

Fieldskeyword [[character](#)] keyword(s)type [[ISOKeywordType](#)] keyword typethesaurusName [[ISOCitation](#)] thesaurus citation**Methods**

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

addKeyword(keyword, locales) Adds a keyword. Locale names can be specified as list with the locales argument.

delKeyword(keyword, locales) Deletes a keyword. Locale names can be specified as list with the locales argument. Local names should match those of the keyword to be deleted, otherwise nothing will be deleted.

setKeywordType(keywordType) Sets the keyword type

setThesaurusName(thesaurusName) Sets the thesaurus name

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

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()

#Example for INSPIRE (GEMET Spatial Data Theme)
inspire_kwd <- ISOKeywords$new()
anc1 <- ISOAnchor$new(
  name = "Environmental monitoring facilities",
  href = "http://inspire.ec.europa.eu/theme/ef"
)
inspire_kwd$addKeyword(anc1)
inspire_kwd$setKeywordType("theme")
th <- ISOCitation$new()
th$setTitle(
  ISOAnchor$new(
    name = "GEMET - INSPIRE themes, version 1.0",
    href="http://www.eionet.europa.eu/gemet/inspire_themes"
  )
)
inspire_date <- ISODate$new()
inspire_date$setDate(as.Date("2008-06-01"))
inspire_date$setDateType("publication")
th$addDate(inspire_date)
inspire_kwd$setThesaurusName(th)

```

Description

ISOKeywordType

Format

R6Class object.

Value

Object of R6Class for modelling an ISO KeywordType

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

R6Class object.

Value

Object of R6Class for modelling an ISO Language

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

Fields

`accessConstraints` [[ISORestriction](#)] access constraint(s)

`useConstraints` [[ISORestriction](#)] use constraint(s)

`otherConstraints` [[character](#)] other constraint(s)

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.blondel1@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

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 <emmanuel.blondel1@gmail.com>

References

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

ISOLineage

ISOLineage

Description

ISOLineage

Format

[R6Class](#) object.

Value

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

Fields

statement [[character](#)] statement
 processStep [[ISOProcessStep](#)] process step(s)
 source [[ISOSource](#)] source(s)

Methods

new(xml) This method is used to instantiate an [ISOLineage](#)
 setStatement(statement, locales) Sets the statement. Locale names can be specified as list with the locales argument.
 addProcessStep(processStep) Adds a process step (object of class [ISOProcessStep](#))
 delProcessStep(processStep) Deletes a process step, object of class [ISOProcessStep](#)
 addSource(source) Adds a source (object of class [ISOSource](#))
 delSource(source) Deletes a source (object of class [ISOSource](#))

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
lineage <- ISOLineage$new()
lineage$setStatement("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

Format

[R6Class](#) object.

Value

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

Fields

label [[character](#)] label

code [[character](#)] code

definition [[character](#)] definition

definitionReference [[ISODefinitionReference](#)] definition reference

Methods

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

`setLabel(label, locales)` Sets the label. Locale names can be specified as list with the locales argument.

`setCode(code, locales)` Sets the code. Locale names can be specified as list with the locales argument.

`setDefinition(definition, locales)` Sets the definition. Locale names can be specified as list with the locales argument.

`setDefinitionReference(definitionReference)` Sets the definition reference, object of class [ISODefinitionReference](#)

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()
```

ISOLocale

ISOLocale

Description

ISOLocale

Format

[R6Class](#) object.

Value

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

Fields

languageCode [[ISOLanguageIcharacter](#)] language code

country [[ISOCountryIcharacter](#)] country code

characterEncoding [[ISOCharacterSetIcharacter](#)] character encoding

Methods

new(xml, code, country, encoding) This method is used to instantiate an ISOLocale

setId(id) Set the locale Id, to bind to ISOFreeText elements to identify a locale text.

setLanguage(language) Set the language, object of class "character" (language code), or object of class [ISOLanguage](#).

setCountry(country) Set the country, object of class "character" (country code), or object of class [ISOCountry](#).

setCharacterSet(charset) Set the character encoding, object of class "character" (encoding code), or object of class [ISOCharacterSet](#).

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
loc <- ISOLocale$new()
loc$setId("eng")
loc$setLanguage("eng")
loc$setCountry("UK")
loc$setCharacterSet("utf8")
```

ISOLocaleContainer	<i>ISOLocaleContainer</i>
--------------------	---------------------------

Description

ISOLocaleContainer

Format

[R6Class](#) object.

Value

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

Fields

description [[character](#)] description
 locale [[ISOLocale](#)] locale
 date [[ISODate](#)] date(s)
 responsibleParty [[ISOResponsibleParty](#)] responsible party
 localisedString [[character](#)] localised string(s)

Methods

`new(xml)` This method is used to instantiate an [ISOLocaleContainer](#)
`setDescription(description, locales)` Sets the process step description. Locale names can be specified as list with the `locales` argument.
`setLocale(locale)` Sets the locale, object of class [ISOLocale](#)
`addDate(date)` Add a date, object of class [ISODate](#)
`delDate(date)` Deletes a date, object of class [ISODate](#)
`addResponsibleParty(responsibleParty)` Add a responsible party, object of class [ISOResponsibleParty](#)
`delResponsibleParty(responsibleParty)` Deletes a responsible party, object of class [ISOResponsibleParty](#)
`addLocalisedString(string)` Adds a localised string
`delLocalisedString(string)` Deletes a localised string

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOLocalisedCharacterString
ISOLocalisedCharacterString

Description

ISOLocalisedCharacterString

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, locale, value)` This method is used to instantiate an [ISOLocalisedCharacterString](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

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

Examples

```
str <- ISOLocalisedCharacterString$new(locale = "FR", value = "ma description")
str$encode()
```

ISOLocalName	<i>ISOLocalName</i>
--------------	---------------------

Description

ISOLocalName

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] local name

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

ISOMaintenanceFrequency	<i>ISOMaintenanceFrequency</i>
-------------------------	--------------------------------

Description

ISOMaintenanceFrequency

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOMaintenanceFrequency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

Fields

`maintenanceAndUpdateFrequency` [[ISOMaintenanceFrequency](#)|[character](#)] frequency

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

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] measure

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

 ISOMedium

ISOMedium

Description

ISOMedium

Format

[R6Class](#) object.

Value

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

Fields

name [[ISOMediumName|character](#)] name
 density [[numeric](#)] density
 densityUnits [character](#) density unit
 volumes [[integer](#)] volumes
 mediumFormat [[ISOMediumFormat|character](#)] format
 mediumNode [character](#) note

Methods

`new(xml)` This method is used to instantiate an [ISOMedium](#)
`setName(name)` Set the medium name, object of class 'character' or [ISOMediumName](#)
`addDensity(density)` Adds a density
`delDensity(density)` Deletes density
`setDensityUnits(densityUnits)` Set density unit
`setVolumes(volumes)` Set volumes
`addMediumFormat(mediumFormat)` Adds a medium format, object of class 'character' or [ISOMediumFormat](#)
`delMediumFormat(mediumFormat)` Deletes a medium format, object of class 'character' or [ISOMediumFormat](#)
`setMediumNote(mediumNote, locales)` Set a medium note. Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
md <- ISOMedium$new()
md$setName("satellite")
md$addDensity(1.0)
md$setDensityUnits("string")
md$setVolumes(1L)
md$addMediumFormat("tar")
md$setMediumNote("some note")
xml <- md$encode()
```

ISOMediumFormat

ISOMediumFormat

Description

ISOMediumFormat

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOMediumFormat](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#MediumFormat
MediumFormat <- ISOMediumFormat$new(value = "tar")
```

ISOMediumName	<i>ISOMediumName</i>
---------------	----------------------

Description

ISOMediumName

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISOMediumName](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#MediumName
MediumName <- ISOMediumName$new(value = "satellite")
```

ISOMemberName	<i>ISOMemberName</i>
---------------	----------------------

Description

ISOMemberName

Format

[R6Class](#) object.

Value

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

Fields

aName [[character](#)] member name

attributeType [[ISOTypeName|character](#)] attribute type

Methods

new(xml, aName, attributeType) This method is used to instantiate an [ISOMemberName](#)

setName(aName) Set the aName, object of class character. Locale names can be specified as list with the locales argument.

setAttributeType(attributeType, locales) Set the attribute type, object of class ISOTypeName or character Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

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

ISOMetadata

ISOMetadata

Description

ISOMetadata

Format

[R6Class](#) object.

Value

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

Fields

fileIdentifier [[character](#)] metadata file identifier
 language [[ISOLanguage|character](#)] metadata language coded
 characterSet [[ISOCharacterSet|character](#)] character set
 parentIdentifier [[character](#)] parent metadata identifier
 hierarchyLevel [[ISOHierarchyLevel|character](#)] hierarchy level
 hierarchyLevelName [[character](#)] hierarchy level name
 contact [[ISOResponsibleParty](#)] contact(s)
 dateStamp [[Date|POSIXt](#)] datestamp
 metadataStandardName [[character](#)] metadata standard name
 metadataStandardVersion [[character](#)] metadata standard version
 dataSetURI [[character](#)] dataset URI
 spatialRepresentationInfo [[ISOSpatialRepresentation](#)] the spatial representation
 referenceSystemInfo [[ISOReferenceSystem](#)] the reference system
 metadataExtensionInfo [[ISOMetadataExtensionInformation](#)] metadata extension
 identificationInfo [[ISOIdentification](#)] identification information
 contentInfo [[ISOFeatureCatalogueDescription|ISOCoverageDescription](#)] content description
 distributionInfo [[ISODistribution](#)] distribution information
 dataQualityInfo [[ISODataQuality](#)] data quality
 metadataMaintenance [[ISOMaintenanceInformation](#)] maintenance info

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
 addLocale(locale) Adds a locale, object of class ISOLocale
 delLocale(locale) Deletes a locale, object of class ISOLocale
 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("someresourcenam")
  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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(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)

#add an INSPIRE spatial data theme?
inspire_kwd <- ISOKeywords$new()
anc1 <- ISOAnchor$new(
  name = "Environmental monitoring facilities",

```



```

    href = "http://inspire.ec.europa.eu/theme/ef"
  )
  inspire_kwd$addKeyword(anc1)
  inspire_kwd$setKeywordType("theme")
  th <- ISOCitation$new()
  th$setTitle(
    ISOAnchor$new(
      name = "GEMET - INSPIRE themes, version 1.0",
      href="http://www.eionet.europa.eu/gemet/inspire_themes"
    )
  )
  inspire_date <- ISODate$new()
  inspire_date$setDate(as.Date("2008-06-01"))
  inspire_date$setDateType("publication")
  th$addDate(inspire_date)
  inspire_kwd$setThesaurusName(th)
  ident$addKeywords(inspire_kwd)

#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)

#create dataQuality object with a 'dataset' scope
dq <- ISODataQuality$new()
scope <- ISOScope$new()
scope$setLevel("dataset")
dq$setScope(scope)

#add data quality reports...

#add a report the data quality
dc <- ISODomainConsistency$new()
result <- ISOConformanceResult$new()
spec <- ISOCitation$new()
spec$setTitle("Data Quality check")

```

```

spec$setAlternateTitle("This is is some data quality check report")
d <- ISODate$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 INSPIRE reports?
#INSPIRE - interoperability of spatial data sets and services
dc_inspire1 <- ISODomainConsistency$new()
cr_inspire1 <- ISOConformanceResult$new()
cr_inspire_spec1 <- ISOCitation$new()
cr_title1 <- paste(
  "Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards interoperability of spatial data",
  "sets and services"
)
cr_inspire_spec1$setTitle(cr_title1)
cr_inspire1$setExplanation("See the referenced specification")
cr_inspire_date1 <- ISODate$new()
cr_inspire_date1$setDate(ISOdate(2010,12,8))
cr_inspire_date1$setDateType("publication")
cr_inspire_spec1$addDate(cr_inspire_date1)
cr_inspire1$setSpecification(cr_inspire_spec1)
cr_inspire1$setPass(TRUE)
dc_inspire1$addResult(cr_inspire1)
dq$addReport(dc_inspire1)
#INSPIRE - metadata
dc_inspire2 <- ISODomainConsistency$new()
cr_inspire2 <- ISOConformanceResult$new()
cr_inspire_spec2 <- ISOCitation$new()
cr_title2 <- paste(
  "COMMISSION REGULATION (EC) No 1205/2008 of 3 December 2008 implementing Directive 2007/2/EC",
  "of the European Parliament and of the Council as regards metadata"
)
cr_inspire_spec2$setTitle(cr_title2)
cr_inspire2$setExplanation("See the referenced specification")
cr_inspire_date2 <- ISODate$new()
cr_inspire_date2$setDate(ISOdate(2008,12,4))
cr_inspire_date2$setDateType("publication")
cr_inspire_spec2$addDate(cr_inspire_date2)
cr_inspire2$setSpecification(cr_inspire_spec2)
cr_inspire2$setPass(TRUE)
dc_inspire2$addResult(cr_inspire2)
dq$addReport(dc_inspire2)

#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$new()
cit$setTitle("sometitle")
d <- ISODate$new()
d$setDate(ISOdate(2015, 1, 1, 1))
d$setDateType("publication")
cit$addDate(d)
cit$setEdition("1.0")
cit$setEditionDate(as.Date(ISOdate(2015, 1, 1, 1)))
contact = ISOContact$new()
fcLink <- ISOOnlineResource$new()
fcLink$setLinkage("http://somelink/featurecatalogue")
contact$setOnlineResource(fcLink)
rp = ISOResponsibleParty$new()
rp$setRole("publisher")
rp$setContactInfo(contact)
cit$addCitedResponsibleParty(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

Format

[R6Class](#) object.

Value

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

Fields

extensionOnLineResource [[ISOOnlineResource](#)] online resource

extendedElementInformation [[ISOExtendedElementInformation](#)] extended element(s)

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$setVoice("myphonenumber")
phone$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$setContactInfo(contact)
elem$addSource(rp)

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

xml <- md$encode()
```

ISOMetadataNamespace *ISOMetadataNamespace*

Description

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

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()
```

ISOMimeFileType *ISOMimeFileType*

Description

ISOMimeFileType

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

ISOMultiplicity

Description

ISOMultiplicity

Format

[R6Class](#) object.

Value

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

Fields

range [[ISOMultiplicityRange](#)] multiplicity range

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

Format

[R6Class](#) object.

Value

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

Fields

lower [[integer](#)] lower value

upper [[integer](#)] upper 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()
```

ISONonQuantitativeAttributeAccuracy
ISONonQuantitativeAttributeAccuracy

Description

ISONonQuantitativeAttributeAccuracy

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISONonQuantitativeAttributeAccuracy](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISONonQuantitativeAttributeAccuracy$new()
dq$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()
```

ISOobligation

ISOobligation

Description

ISOobligation

Format

[R6Class](#) object.

Value

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

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")
```

ISOOnLineFunction *ISOOnLineFunction*

Description

ISOOnLineFunction

Format

[R6Class](#) object.

Value

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

Fields

value

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#example
download <- ISOOnLineFunction$new(value = "download")
```

ISOOnlineResource *ISOOnlineResource*

Description

ISOOnlineResource

Format

[R6Class](#) object.

Value

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

Fields

linkage [[ISOURL](#)] linkage
protocol [[character](#)] protocol
name [[character](#)] name
description [[character](#)] description
function [[ISOOnlineFunction](#)] online function

Methods

`new(xml)` This method is used to instantiate an [ISOOnlineResource](#)
`setLinkage(linkage)` Sets the linkage (URL), an object of class [character](#) or [ISOURL](#)
`setProtocol(protocol, locales)` Sets the protocol. Locale names can be specified as list with the locales argument.
`setName(name, locales)` Sets the name. Locale names can be specified as list with the locales argument.
`setDescription(description, locales)` Sets the description. Locale names can be specified as list with the locales argument.
`setOnlineFunction(onlineFunction)` Sets the online function, object of class [ISOOnlineFunction](#) or [character](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

ISOOperationMetadata *ISOOperationMetadata*

Description

ISOOperationMetadata

Format

[R6Class](#) object.

Value

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

Fields

operationName [[character](#)] operation name
DCP [[ISODCPList](#)] DCP
operationDescription [[character](#)] operation description
invocationName [[character](#)] invocation name
parameters [[ISOParameter](#)] parameter(s)
connectPoint [[ISOOnlineResource](#)] online resources
dependsOn [[ISOOperationMetadata](#)] dependent operation metadata

Methods

new(xml) This method is used to instantiate an [ISOOperationMetadata](#)
setOperationName(operationName, locales) Set the operation name. Locale names can be specified as list with the locales argument.
addDCP(dcp) Add a DCP
delDCP(dcp) Deletes a DCP
setOperationDescription(operationDescription, locales) Set the operation description. Locale names can be specified as list with the locales argument.

`setInvocationName(invocationName, locales)` Set the invocation name. Locale names can be specified as list with the locales argument.

`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, object of class [ISOOperationMetadata](#)

`delDependentOperationMetadata(operationMetadata)` Deletes dependent operation metadata, object of class [ISOOperationMetadata](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19119:2005 - Geographic information – Services

Examples

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

ISOOtherAggregate *ISOOtherAggregate*

Description

ISOOtherAggregate

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOOtherAggregate](#)

Inherited methods from [ISOAbstractAggregate](#)

[addComposedOf\(composedOf\)](#) Adds a dataset, object of class [ISODataset](#)
[delComposedOf\(composedOf\)](#) Deletes a dataset, object of class [ISODataset](#)
[addSeriesMetadata\(metadata\)](#) Adds a series metadata, object of class [ISOMetadata](#)
[delSeriesMetadata\(metadata\)](#) Deletes a series metadata, object of class [ISOMetadata](#)
[addSubset\(subset\)](#) Adds a subset, object that inherits from [ISOAbstractAggregate](#)
[delSubset\(subset\)](#) Deletes a subset, object that inherits from [ISOAbstractAggregate](#)
[addSuperset\(superset\)](#) Adds a superset, object that inherits from [ISOAbstractAggregate](#)
[delSuperset\(superset\)](#) Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOPParameter

ISOPParameter

Description

ISOPParameter

Format

[R6Class](#) object.

Value

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

Fields

name [[character](#)] name
direction [[ISOPParameterDirection](#)] direction
description [[character](#)] description
optionality [[logical](#)] parameter optionality
repeatability [[logical](#)] parameter repeatability
valueType [[ISOTypeNameCharacter](#)] value type

Methods

- `new(xml)` This method is used to instantiate an [ISOPParameter](#)
- `setName(name, attributeType, locales)` Sets the parameter name (character) and attribute-Type ([ISOTypeName](#) or character). Locale names can be specified as list with the locales argument.
- `setDirection(direction)` Sets the direction, an object of class [ISOPParameterDirection](#) or any character value among `ISOPParameterDirection$values()`
- `setDescription(description, locales)` Sets the parameter description. Locale names can be specified as list with the locales argument.
- `setOptionality(optional)` Set whether the parameter is optional (TRUE), FALSE otherwise
- `setRepeatability(repeatable)` Set whether the parameter is repeatable (TRUE), FALSE otherwise
- `setValueType(valueType, locales)` Sets the type of parameter value, object of class [ISOTypeName](#) or character Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19119:2005 - Geographic information – Services

Examples

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

ISOPParameterDirection *ISOPParameterDirection*

Description

ISOPParameterDirection

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISOParameterDirection](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19119:2005 - Geographic information – Services

Examples

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

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

ISOPixelOrientation *ISOPixelOrientation*

Description

ISOPixelOrientation

Format

[R6Class](#) object.

Value

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

Methods

`new(xml,value, description)` This method is used to instantiate an [ISOPixelOrientation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

#PixelOrientation
PixelOrientation <- ISOPixelOrientation$new(value = "center")
```

ISOPlatform

ISOPlatform

Description

ISOPlatform

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOPlatform](#)

Inherited methods from [ISOAbstractAggregate](#)

`addComposedOf(composedOf)` Adds a dataset, object of class [ISODataset](#)
`delComposedOf(composedOf)` Deletes a dataset, object of class [ISODataset](#)
`addSeriesMetadata(metadata)` Adds a series metadata, object of class [ISOMetadata](#)
`delSeriesMetadata(metadata)` Deletes a series metadata, object of class [ISOMetadata](#)
`addSubset(subset)` Adds a subset, object that inherits from [ISOAbstractAggregate](#)
`delSubset(subset)` Deletes a subset, object that inherits from [ISOAbstractAggregate](#)
`addSuperset(superset)` Adds a superset, object that inherits from [ISOAbstractAggregate](#)
`delSuperset(superset)` Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOPortrayalCatalogueReference
ISOPortrayalCatalogueReference

Description

ISOPortrayalCatalogueReference

Format

[R6Class](#) object.

Value

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

Fields

portrayalCatalogueCitation [[ISOCitation](#)] the catalogue citation

Methods

new(xml) This method is used to instantiate an [ISOPortrayalCatalogueReference](#)

addCitation(citation) Add citation, object of class [ISOCitation](#)

delCitation(citation) Deletes citation, object of class [ISOCitation](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
md <- ISOPortrayalCatalogueReference$new()
#citation
rp <- ISOResponsibleParty$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://somelink")
res$setName("somename")
contact$setOnlineResource(res)
rp$setContactInfo(contact)
ct <- ISOCitation$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$setEditionDate(as.Date(ISOdate(2015, 1, 1, 1)))
ct$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(rp)
md$addCitation(ct)

xml <- md$encode()

```

ISOPresentationForm *ISOPresentationForm*

Description

ISOPresentationForm

Format

[R6Class](#) object.

Value

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

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	<i>ISOProcessStep</i>
----------------	-----------------------

Description

ISOProcessStep

Format

[R6Class](#) object.

Value

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

Fields

description [[character](#)] process step description
 rationale [[character](#)] rationale
 dateTime [[POSIXt](#)] dateTime of the process ste
 processor [[ISOResponsibleParty](#)] party responsible of the process step

Methods

new(xml) This method is used to instantiate an [ISOProcessStep](#)
 setDescription(description, locales) Sets the process step description. Locale names can be specified as list with the locales argument.
 setRationale(rationale, locales) Sets the process step rationale. Locale names can be specified as list with the locales argument.
 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()
```

ISOProductionSeries *ISOProductionSeries*

Description

ISOProductionSeries

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOProductionSeries](#)

Inherited methods from [ISOAbstractAggregate](#)

[addComposedOf\(composedOf\)](#) Adds a dataset, object of class [ISODataset](#)
[delComposedOf\(composedOf\)](#) Deletes a dataset, object of class [ISODataset](#)
[addSeriesMetadata\(metadata\)](#) Adds a series metadata, object of class [ISOMetadata](#)
[delSeriesMetadata\(metadata\)](#) Deletes a series metadata, object of class [ISOMetadata](#)
[addSubset\(subset\)](#) Adds a subset, object that inherits from [ISOAbstractAggregate](#)
[delSubset\(subset\)](#) Deletes a subset, object that inherits from [ISOAbstractAggregate](#)
[addSuperset\(superset\)](#) Adds a superset, object that inherits from [ISOAbstractAggregate](#)
[delSuperset\(superset\)](#) Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOPropertyType	<i>ISOPropertyType</i>
-----------------	------------------------

Description

ISOPropertyType

Format

[R6Class](#) object.

Value

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

Methods

[new\(xml, defaults\)](#) This method is used to instantiate an [ISOPropertyType](#)

Methods Inherited from [ISOAbstractCarrierOfCharacteristics](#)

[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 from ISOAbstractPropertyType

`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, locales)` Sets the definition. Locale names can be specified as list with the locales argument.
`setCardinality(lower, upper)` Sets the cardinality boundaries lower and upper of class numeric
`setDefinitionReference(definitionReference)` Sets the definition Reference, object of class [ISODefinitionReference](#)
`setFeatureCatalogue(featureCatalogue)` Sets a feature catalogue, object of class [ISOFeatureCatalogue](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19110:2005 Methodology for Feature cataloguing

ISOQuantitativeAttributeAccuracy
ISOQuantitativeAttributeAccuracy

Description

ISOQuantitativeAttributeAccuracy

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOQuantitativeAttributeAccuracy](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOQuantitativeAttributeAccuracy$new()
dq$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()
```

ISOQuantitativeResult *ISOQuantitativeResult*

Description

ISOQuantitativeResult

Format

[R6Class](#) object.

Value

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

Fields

valueType [ISORecordType] record type
valueUnit [GMLUnitDefinition] unit
errorStatistic [character] error statistic
value [ISORecord] record(s)

Methods

new(xml) This method is used to instantiate an [ISOQuantitativeResult](#)
setValueType(valueType) Sets value type
setValueUnit(valueUnit) Sets value unit
setErrorStatistic(errorStatistic) Sets error statistic
addValue(value) Add value
delValue(value) Deletes value

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

ISORangeDimension *ISORangeDimension*

Description

ISORangeDimension

Format

[R6Class](#) object.

Value

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

Fields

sequenceIdentifier [[ISOMemberName](#)] sequence identifier
 descriptor [[character](#)] description

Methods

new(xml) This method is used to instantiate an ISORangeDimension
 setSequenceIdentifier(memberName) Sets the sequence identifier, object of class ISOMemberName
 setDescription(descriptor ,locales) Sets the descriptor, object of class character. Locale names can be specified as list with the locales argument.

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

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] record

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

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] record type

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

Format

[R6Class](#) object.

Value

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

Fields

codeSpace [[character](#)] codespace

version [[character](#)] version

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

Format

[R6Class](#) object.

Value

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

Fields

referenceSystemIdentifier [[ISOReferenceIdentifier](#)] the reference system identifier

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

Format

[R6Class](#) object.

Value

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

Fields

denominator [[integer](#)] denominator

Methods

`new(xml, denominator)` This method is used to instantiate an [ISORepresentativeFraction](#)
`setDenominator(denominator)` Sets the denominator, object of class integer

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

ISOResolution

ISOResolution

Description

ISOResolution

Format

R6Class object.

Value

Object of R6Class for modelling an ISO Resolution

Fields

equivalentScale [ISORepresentativeFraction] equivalent scale

distance [ISODistance] distance

Methods

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

setEquivalentScale(equivalentScale) Set equivalent scale. By setting an equivalent scale, the value of field 'distance' will be set to NULL.

setDistance(distance) Set distance. By setting a distance, the value of field 'equivalentScale' will be set to NULL.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
md <- ISOResolution$new()
md$setDistance(ISODistance$new(value = 1, uom = "m", useUomURI = TRUE))
xml <- md$encode()
```

ISOResponsibleParty *ISOResponsibleParty*

Description

ISOResponsibleParty

Format

[R6Class](#) object.

Value

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

Fields

individualName [[character](#)] Individual name
organisationName [[character](#)] Organization name
positionName [[character](#)] Position name
contactInfo [[ISOContact](#)] contact information
role [[ISORole](#)] role

Methods

new(xml, value) This method is used to instantiate an ISOResponsibleParty
setIndividualName(individualName, locales) Set the individual name. Locale names can be specified as list with the locales argument.
setOrganisationName(organisationName, locales) Set the organisation name. Locale names can be specified as list with the locales argument.
setPositionName(positionName, locales) Set the position name. Locale names can be specified as list with the locales argument.
setContactInfo(contactInfo) Set the contact info, should be an object of class [ISOContact](#)
setRole(role) Set the role, either an object of class "character" (among values available in [ISORole](#).\$values()) or an object of class [ISORole](#).

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

Format

R6Class object.

Value
Object of [R6Class](#) for modelling an ISO Restriction
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 <- ISORestriction$values(labels = TRUE)

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

ISORole

ISORole

Description

ISORole

Format

[R6Class](#) object.

Value

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

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

Methods

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

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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	<i>ISOScale</i>
----------	-----------------

Description

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	<i>ISOScope</i>
----------	-----------------

Description

ISOScope

Format

[R6Class](#) object.

Value

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

Fields

level [[ISOHierarchyLevel](#)] the scope/hierarchy level

Methods

new(xml) This method is used to instantiate an [ISOScope](#)

setLevel(level) Sets the scope level, object of class 'character' or [ISOHierarchyLevel](#)

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()
```

ISOScopeDescription *ISOScopeDescription*

Description

ISOScopeDescription

Format

[R6Class](#) object.

Value

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

Methods

new(xml) This method is used to instantiate an [ISOScopeDescription](#)

addAttribute(attribute) Adds an attribute

delAttribute(attribute) Deletes an attribute

addAttributeInstance(attributeInstance) Adds an attribute instance

delAttributeInstance(attributeInstance) Deletes an attribute instance

addFeature(feature) Adds an feature

`delFeature(feature)` Deletes an feature
`addFeatureInstance(featureInstance)` Adds an feature instance
`delFeatureInstance(featureInstance)` Deletes an feature instance
`setDataset(dataset)` Set the dataset
`setOther(other)` Set other description

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

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

Description

ISOScopedName

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] scope name

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

ISOSecurityConstraints

Description

ISOSecurityConstraints

Format

[R6Class](#) object.

Value

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

Fields

classification [[ISOClassification](#)] security classification

userNote [[character](#)] user note

classificationSystem [[character](#)] classification system

handlingDescription [[character](#)] description

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, locales)` Sets a user note as object of class "character". Locale names can be specified as list with the locales argument.

`setClassificationSystem(classificationSystem, locales)` Sets a classification system as object of class "character". Locale names can be specified as list with the locales argument.

`setHandlingDescription(handlingDescription, locales)` Sets a handling description as object of class "character". Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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()
```

ISOSensor

*ISOSensor***Description**

ISOSensor

Format[R6Class](#) object.**Value**Object of [R6Class](#) for modelling an ISOSensor**Methods**`new(xml)` This method is used to instantiate an [ISOSensor](#)**Inherited methods from [ISOAbstractAggregate](#)**

`addComposedOf(composedOf)` Adds a dataset, object of class [ISODataset](#)
`delComposedOf(composedOf)` Deletes a dataset, object of class [ISODataset](#)
`addSeriesMetadata(metadata)` Adds a series metadata, object of class [ISOMetadata](#)
`delSeriesMetadata(metadata)` Deletes a series metadata, object of class [ISOMetadata](#)
`addSubset(subset)` Adds a subset, object that inherits from [ISOAbstractAggregate](#)
`delSubset(subset)` Deletes a subset, object that inherits from [ISOAbstractAggregate](#)
`addSuperset(superset)` Adds a superset, object that inherits from [ISOAbstractAggregate](#)
`delSuperset(superset)` Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOSeries

ISOSeries

Description

ISOSeries

Format

[R6Class](#) object.

Value

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

Methods

`new(xml)` This method is used to instantiate an [ISOSeries](#)

Inherited methods from [ISOAbstractAggregate](#)

`addComposedOf(composedOf)` Adds a dataset, object of class [ISODataset](#)

`delComposedOf(composedOf)` Deletes a dataset, object of class [ISODataset](#)

`addSeriesMetadata(metadata)` Adds a series metadata, object of class [ISOMetadata](#)

`delSeriesMetadata(metadata)` Deletes a series metadata, object of class [ISOMetadata](#)

`addSubset(subset)` Adds a subset, object that inherits from [ISOAbstractAggregate](#)

`delSubset(subset)` Deletes a subset, object that inherits from [ISOAbstractAggregate](#)

`addSuperset(superset)` Adds a superset, object that inherits from [ISOAbstractAggregate](#)

`delSuperset(superset)` Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOServiceIdentification

ISOServiceIdentification

Description

ISOServiceIdentification

Format

[R6Class](#) object.

Value

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

Methods inherited from [ISOIdentification](#)

`setCitation(citation)` Sets an object of class [ISOCitation](#)

`setAbstract(abstract, locales)` Sets an abstract (object of class "character"). Locale names can be specified as list with the locales argument.

`setPurpose(purpose, locales)` Sets a purpose (object of class "character"). Locale names can be specified as list with the locales argument.

`addCredit(credit, locales)` Adds a credit (object of class "character"). Locale names can be specified as list with the locales argument.

`delCredit(credit, locales)` Deletes a credit (object of class "character"). Locale names can be specified as list with the locales argument.

`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](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

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("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$setLinkage("http://www.somewhereovertheweb.org")
res$setName("somenam")
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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(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)

xml <- md$encode()

```

ISOSource

ISOSource

Description

ISOSource

Format

[R6Class](#) object.

Value

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

Fields

description [[character](#)] source description
 scaleDenominator [[ISORepresentativeFraction](#)] scale denominator
 sourceReferenceSystem [[ISOReferenceSystem](#)] source reference system
 sourceCitation [[ISOCitation](#)] the source citation
 sourceExtent [[ISOExtent](#)] the source extent(s)
 sourceStep [[ISOProcessStep](#)] the source process step(s)

Methods

new(xml) This method is used to instantiate an [ISOSource](#)
 setDescription(description, locales) Sets the source description. Locale names can be specified as list with the locales argument.
 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 [ISOReferenceSystem](#))
 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 (object of class [ISOExtent](#))
 addProcessStep(processStep) Adds a source process step (object of class [ISOProcessStep](#))
 delProcessStep(processStep) Deletes a source processStep(object of class [ISOProcessStep](#))

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

ISOSpatialRepresentation

ISOSpatialRepresentation

Description

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

Format

[R6Class](#) object.

Value

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

Methods

`new(xml, value, description)` This method is used to instantiate an [ISOSpatialRepresentationType](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

R6Class object.

Value

Object of R6Class for modelling an ISO SpatialTemporalExtent

Fields

spatialExtent [[ISOGeographicExtent](#)] the geographic extent(s)

Methods inherited from [ISOTemporalExtent](#)

setTimeInstant(timeInstant) Sets a time instant, object of class [GMLTimeInstant](#)

setTimePeriod(timePeriod) Sets a time period, object of class [GMLTimePeriod](#)

Methods

new(xml) This method is used to instantiate an [ISOSpatialTemporalExtent](#)

addSpatialExtent(spatialExtent) Adds an object of class [ISOGeographicExtent](#)

delSpatialExtent(spatialExtent) Deletes an object of class [ISOGeographicExtent](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@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$addSpatialExtent(spatialExtent)

xml <- md$encode()
```

ISOSRVServiceIdentification
ISOSRVServiceIdentification

Description

ISOSRVServiceIdentification

Format

[R6Class](#) object.

Value

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

Fields

serviceType [[character](#)] the service type
 serviceTypeVersion [[character](#)] the service type version(s)
 accessProperties [[ISOStandardOrderProcess](#)] the access properties
 restrictions [[ISOConstraints](#)] restrictions applied to service
 keywords [[ISOKeywords](#)] service keywords
 extent [[ISOExtent](#)] service extent(s)
 coupledResource [[ISOCoupledResource](#)] the coupled resource
 couplingType [[ISOCouplingType](#)] the coupling type
 containsOperations [[ISOOperationMetadata](#)] the operation metadata
 operatesOn [[ISODataIdentification](#)] data identification on which service operates

Methods inherited from [ISOServiceIdentification](#)

setCitation(citation) Sets an object of class [ISOCitation](#)
 setAbstract(abstract, locales) Sets an abstract (object of class "character"). Locale names can be specified as list with the locales argument.
 setPurpose(purpose, locales) Sets a purpose (object of class "character"). Locale names can be specified as list with the locales argument.
 addCredit(credit, locales) Adds a credit (object of class "character"). Locale names can be specified as list with the locales argument.

`delCredit(credit, locales)` Deletes a credit (object of class "character"). Locale names can be specified as list with the locales argument.

`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 [ISOSRVServiceIdentification](#)

`setServiceType(serviceType)` Set the type of service

`addServiceTypeVersion(version)` Adds a service type version

`delServiceTypeVersion(version)` Deletes a service type version

`setAccessProperties(accessProperties)` Sets the access properties, as object of class [ISOStandardOrderProcess](#)

`setRestrictions(restrictions)` Sets the restrictions associated to the service, as object of class [ISOConstraints](#)

`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](#)

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

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

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

addCoupledResource(resource) Adds a coupled resource, object of class [ISOCoupledResource](#)
 delCoupledResource(resource) Deletes a coupled resource, object of class [ISOCoupledResource](#)
 setCouplingType(couplingType) Set the coupling type, object of class character values among
 following: "loose" "mixed" "tight"
 addOperationMetadata(operationMetadata) Adds operation metadata, object of class [ISOOperationMetadata](#)
 delOperationMetadata(operationMetadata) Deletes operation metadata, object of class [ISOOperationMetadata](#)
 addOperatesOn(dataIdentification) Adds an operates on relationship, object of class [ISODataIdentification](#)
 delOperatesOn(dataIdentification) Deletes an operates on relationship, object of class [ISODataIdentification](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19119:2005 - Geographic information – Services

Examples

```

#encoding
md <- ISOSRVServiceIdentification$new()
md$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("somenam")
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$addIdentifier(ISOMetaIdentifier$new(code = "identifier"))
ct$addPresentationForm("mapDigital")
ct$addCitedResponsibleParty(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$addParameter(param)
}
outParam <-ISOParameter$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$setInvocationName(invocationName)
for(i in 1:3){
  param <- ISOParameter$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")
  wpsOp$addParameter(param)
}
outParam <-ISOParameter$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()

```

Description

ISOStandardOrderProcess

Format

[R6Class](#) object.

Value

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

Fields

fees [[character](#)]
plannedAvailableDateTime [[POSIXt](#)] the datetime
orderingInstructions [[character](#)] ordering instructions
turnaround [[character](#)] turnaround

Methods

`new(xml)` This method is used to instantiate an [ISOStandardOrderProcess](#)
`setFees(fees, locales)` Sets fees, object of class character. Locale names can be specified as list with the locales argument.
`setPlannedAvailableDateTime(dateTime)` Sets planned available datetime, object of class c('POSIXct', 'POSIXlt')
`setOrderingInstructions(instructions, locales)` Sets ordering instructions, object of class character. Locale names can be specified as list with the locales argument.
`setTurnaround(turnaround, locales)` Sets turnaround, object of class character. Locale names can be specified as list with the locales argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@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

Format

[R6Class](#) object.

Value

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

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")
```

ISOStereoMate

ISOStereoMate

Description

ISOStereoMate

Format

[R6Class](#) object.

Value

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

Methods

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

Methods inherited from [ISOAbstractAggregate](#)

`addComposedOf(composedOf)` Adds a dataset, object of class [ISODataset](#)

`delComposedOf(composedOf)` Deletes a dataset, object of class [ISODataset](#)

`addSeriesMetadata(metadata)` Adds a series metadata, object of class [ISOMetadata](#)

`delSeriesMetadata(metadata)` Deletes a series metadata, object of class [ISOMetadata](#)

`addSubset(subset)` Adds a subset, object that inherits from [ISOAbstractAggregate](#)

`delSubset(subset)` Deletes a subset, object that inherits from [ISOAbstractAggregate](#)

`addSuperset(superset)` Adds a superset, object that inherits from [ISOAbstractAggregate](#)

`delSuperset(superset)` Deletes a superset, object that inherits from [ISOAbstractAggregate](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOTelephone

ISOTelephone

Description

ISOTelephone

Format

[R6Class](#) object.

Value

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

Fields

voice [[character](#)] phone number

facsimile [[character](#)] facsimile number

Methods

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

`setVoice(voice, locales)` Set voice phone number. Locale numbers can be specified as list with the `locales` argument.

`setFacsimile(voice, locales)` Set facsimile phone number. Locale numbers can be specified as list with the `locales` argument.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

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

ISOTemporalConsistency

ISOTemporalConsistency

Description

ISOTemporalConsistency

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOTemporalConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOTemporalConsistency$new()
dq$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()
```

ISOTemporalExtent

ISOTemporalExtent

Description

ISOTemporalExtent

Format

[R6Class](#) object.

Value

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

Fields

extent [[GMLTimeInstant](#)|[GMLTimePeriod](#)] the temporal extent (instant or period)

Methods

new(xml) This method is used to instantiate an [ISOTemporalExtent](#)

setTimeInstant(timeInstant) Sets a time instant, object of class [GMLTimeInstant](#)

setTimePeriod(timePeriod) Sets a time period, object of class [GMLTimePeriod](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@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)
```

ISOTemporalValidity *ISOTemporalValidity*

Description

ISOTemporalValidity

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

new(xml) This method is used to instantiate an [ISOTemporalValidity](#)

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOTemporalValidity$new()
dq$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()
```

ISOThematicClassificationCorrectness

ISOThematicClassificationCorrectness

Description

ISOThematicClassificationCorrectness

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOThematicClassificationCorrectness](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOThematicClassificationCorrectness$new()
dq$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()
```

 ISOTimePeriod

ISOTimePeriod

Description

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 *ISOTopicCategory*

Description

ISOTopicCategory

Format

[R6Class](#) object.

Value

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

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")
```

ISOTopologicalConsistency
ISOTopologicalConsistency

Description

ISOTopologicalConsistency

Format

[R6Class](#) object.

Value

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

Inherited methods

from [ISODataQualityAbstractElement](#)

Methods

`new(xml)` This method is used to instantiate an [ISOTopologicalConsistency](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

Examples

```
#encoding
dq <- ISOTopologicalConsistency$new()
dq$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()
```

ISOTopologyLevel	<i>ISOTopologyLevel</i>
------------------	-------------------------

Description

ISOTopologyLevel

Format

[R6Class](#) object.

Value

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

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	<i>ISOTypeName</i>
-------------	--------------------

Description

ISOTypeName

Format

[R6Class](#) object.

Value

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

Fields

aName [[character](#)] the type name

Methods

`new(xml, aName)` This method is used to instantiate an [ISOTypeName](#)

`setName(aName, locales)` Sets the aName. Locale names can be specified as `list` with the `locales` argument.

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

Format

[R6Class](#) object.

Value

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

Fields

value [[integer|numeric](#)] the integer

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

Format

[R6Class](#) object.

Value

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

Fields

value [[character](#)] the url

Methods

new(xml, value) This method is used to instantiate an [ISOURL](#)

setUrl(url) Set the url

Note

Class used by geometa internal XML decoder/encoder

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOUsage

ISOUsage

Description

ISOUsage

Format

[R6Class](#) object.

Value

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

Fields

specificUsage [[character](#)] a text giving a specific usage

usageDateTime [[POSIXt](#)] the datetime of the usage

userDeterminedLimitations [[character](#)] limitations determined by user

userContactInfo [[ISOResponsibleParty](#)] the user contact

Methods

new(xml) This method is used to instantiate an [ISOUsage](#)
 setSpecificUsage(specificUsage, locales) Set the specific usage, as [character](#) object.
 setUsageDateTime(usageDateTime) Set the usage date time, object of class [POSIXt](#)
 setUserDeterminedLimitations(userDeterminedLimitations, locales) Set the limitations determined by user
 addUserContact(contact) Adds user contact, object of class [ISOResponsibleParty](#)
 delUserContact(contact) Deletes user contact, object of class [ISOResponsibleParty](#)

Author(s)

Emmanuel Blondel <emmanuel.blondell@gmail.com>

References

ISO 19115:2003 - Geographic information – Metadata

ISOVectorSpatialRepresentation
ISOVectorSpatialRepresentation

Description

ISOVectorSpatialRepresentation

Format

[R6Class](#) object.

Value

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

Fields

topologyLevel [[character](#)] the topologic level
 geometricObjects [[ISOGeometricObjects](#)] giving type and number of geometries

Methods

new(xml) This method is used to instantiate an [ISOVectorSpatialRepresentation](#)
 setTopologyLevel(topologyLevel) Sets the topology level, object of class [character](#) or [ISOTopologyLevel](#).
 Recommended values among those listed by [ISOTopologyLevel\\$values\(\)](#).
 addGeometricObject(geometricObjects) Adds the geometricObjects, object of class [ISOGeometricObjects](#)
 setGeometricObject(geometricObjects) Sets the geometricObjects, object of class [ISOGeometricObjects](#)
 delGeometricObject(geometricObjects) Deletes the geometricObjects, object of class [ISOGeometricObjects](#)

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	<i>ISOVerticalExtent</i>
-------------------	--------------------------

Description

ISOVerticalExtent

Format

[R6Class](#) object.

Value

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

Fields

minimalValue [[numeric](#)] the minimum value for the vertical extent
maximalValue [[numeric](#)] the maximum value for the vertical extent
unitOfMeasure [[character](#)] the unit of measure
verticalCRS [[GMLVerticalCRS](#)] the vertical CRS

Methods

new(xml) This method is used to instantiate an [ISOVerticalExtent](#)
setMinimumValue(minimumValue) Sets the minimum value, object of class [numeric](#)
setMaximumValue(maximumValue) Sets the maximum value, object of class [numeric](#)
setUnitOfMeasure(uom) Sets the unit of measure, object of class [character](#)
setVerticalCRS(verticalCRS) Sets the vertical CRS, object of class [GMLVerticalCRS](#)

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()
```

pivot_format

pivot_format

Description

pivot_format

Format

[R6Class](#) object.

Value

Object of [R6Class](#) for modelling a mapping format

Fields

id (character) format identifier
pkg (character) package(s) required for the format
reader (function) handler used for reading a property
checker (function) handler used for checking a property value
constructor (function) handler used for object construction

Methods

`new(id, pkg, reader, checker, constructor)` This method is used to instantiate a `pivot_format`, given a unique `id`, the name of package used (for information only). A format is then defined by string expressions (using `sprintf` formatting) to read metadata properties (`reader`), one for checking existence of properties (`checker`), and an expression to create metadata objects (`constructor`). In case the `constructor` is `NULL`, then no conversion to this metadata format will be possible.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

```
#example on how geometa format is defined as pivot format
pivot_format$new(
  id = "geometa", pkg = "geometa",
  reader = "%s[[%s]]", checker = "!is.null(%s[[%s]])",
  constructor = "ISOMetadata$new"
)
```

readISO19139

readISO19139

Description

readISO19139 is a function to read a ISO 19139 from a file or url into an object in the **geometa** model.

Usage

```
readISO19139(file, url, raw)
```

Arguments

file	a valid file path, as object of class character
url	a valid URL, as object of class character
raw	indicates if the function should return the raw XML. By default this is set to FALSE and the function will try to map the xml data to the geometa data model.

Value

a **geometa** object inheriting ISOAbstractObject

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
mdfile <- system.file("extdata/examples", "metadata.xml", package = "geometa")
md <- readISO19139(mdfile)
```

registerISOCodelist *registerISOCodelist*

Description

registerISOCodelist allows to register a new codelist registered in **geometa**

Usage

```
registerISOCodelist(refFile, id, force)
```

Arguments

refFile	ISO XML file handling the ISO codelist
id	identifier of the ISO codelist
force	logical parameter indicating if registration has to be forced in case the identified codelist is already registered

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
registerISOCodelist(  
  refFile = "http://www.isotc211.org/2005/resources/Codelist/ML_gmxCodelists.xml",  
  id = "LanguageCode",  
  force = TRUE  
)
```

registerISOMetadataNamespace
 registerISOMetadataNamespace

Description

registerISOMetadataNamespace allows to register a new namespace in **geometa**

Usage

```
registerISOMetadataNamespace(id, uri, force)
```

Arguments

id	prefix of the namespace
uri	URI of the namespace
force	logical parameter indicating if registration has be to be forced in case the identified namespace is already registered

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
registerISOMetadataNamespace(id = "myprefix", uri = "http://someuri")
```

```
registerISOMetadataSchema  
    registerISOMetadataSchema
```

Description

registerISOMetadataSchema allows to register a new schema in **geometa**

Usage

```
registerISOMetadataSchema(xsdFile)
```

Arguments

xsdFile	the schema XSD file
---------	---------------------

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
registerISOMetadataSchema(xsdFile = "http://www.isotc211.org/2005/gmd/gmd.xsd")
```

registerMappingFormat *registerMappingFormat*

Description

registerMappingFormat allows to register a new mapping format in **geometa**

Usage

```
registerMappingFormat(mapping_format)
```

Arguments

mapping_format object of class pivot_format

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

registerMappings *registerMappings*

Description

MappingFile allows to register in **geometa** a data.frame containing mappings rules to convert from/to other metadata formats (currently EML/emld objects and NetCDF-CF/ncdf4 objects)

Usage

```
registerMappings(x)
```

Arguments

x a data.frame containing the metadata mapping rules

setGeometaOption *setGeometaOption*

Description

setGeometaOption allows to set an option from **geometa**

Usage

```
setGeometaOption(option, value)
```

Arguments

option	the name of the option
value	the value to set for the option

Author(s)

Emmanuel Blondel, <emmanuel.blondel1@gmail.com>

Examples

```
setGeometaOption("schemaBaseUrl", "http://somealternativeurl")
```

setISOCodelists *setISOCodelists*

Description

setISOCodelists

Usage

```
setISOCodelists()
```

setISOMetadataNamespaces
setMetadataNamespaces

Description

setMetadataNamespaces

Usage

setISOMetadataNamespaces()

setISOMetadataSchemas *setISOMetadataSchemas*

Description

setISOMetadataSchemas

Usage

setISOMetadataSchemas()

setMappingFormats *setMappingFormats*

Description

setMappingFormats

Usage

setMappingFormats()

Index

- * **Abstract**
 - GMLAbstractGeneralParameterValue, [25](#)
- * **AcquisitionInformation**
 - ISOImageryAcquisitionInformation, [198](#)
- * **Aggregate**
 - GMLAbstractGeometricAggregate, [26](#)
- * **Binding**
 - ISOBinding, [115](#)
- * **BoundAssociationRole**
 - ISOBoundAssociationRole, [116](#)
- * **BoundFeatureAttribute**
 - ISOBoundFeatureAttribute, [117](#)
- * **CRS**
 - GMLAbstractCRS, [20](#)
 - GMLCompoundCRS, [38](#)
- * **Classification**
 - ISOClassification, [125](#)
- * **CoupledResource**
 - ISOCoupledResource, [136](#)
- * **CoverageContentType**
 - ISOCoverageContentType, [138](#)
- * **Coverage**
 - GMLCOVAbstractCoverage, [42](#)
 - GMLCOVExtension, [43](#)
- * **Curve**
 - GMLAbstractCurve, [20](#)
- * **DCP**
 - ISODCPList, [153](#)
- * **Datatype**
 - ISODatatype, [150](#)
- * **DimensionNameType**
 - ISODimensionNameType, [157](#)
- * **EnvelopeWithTimePeriod**
 - GMLEnvelopeWithTimePeriod, [49](#)
- * **Envelope**
 - GMLEnvelope, [48](#)
- * **EvaluationMethodType**
 - ISOEvaluationMethodType, [164](#)
- * **GMLCOV**
 - GMLCOVAbstractCoverage, [42](#)
 - GMLCOVExtension, [43](#)
- * **GML**
 - GMLAbstractCoordinateOperation, [17](#)
 - GMLAbstractCoordinateSystem, [18](#)
 - GMLAbstractCoverage, [19](#)
 - GMLAbstractCRS, [20](#)
 - GMLAbstractCurve, [20](#)
 - GMLAbstractDiscreteCoverage, [21](#)
 - GMLAbstractFeature, [22](#)
 - GMLAbstractGeneralConversion, [23](#)
 - GMLAbstractGeneralDerivedCRS, [23](#)
 - GMLAbstractGeneralOperationParameter, [24](#)
 - GMLAbstractGeneralParameterValue, [25](#)
 - GMLAbstractGeometricAggregate, [26](#)
 - GMLAbstractGeometricPrimitive, [26](#)
 - GMLAbstractGeometry, [27](#)
 - GMLAbstractGML, [28](#)
 - GMLAbstractImplicitGeometry, [29](#)
 - GMLAbstractObject, [29](#)
 - GMLAbstractReferenceableGrid, [30](#)
 - GMLAbstractRing, [31](#)
 - GMLAbstractSingleCRS, [32](#)
 - GMLAbstractSingleOperation, [32](#)
 - GMLAbstractSurface, [33](#)
 - GMLAbstractTimeObject, [34](#)
 - GMLAbstractTimePrimitive, [35](#)
 - GMLAffineCS, [36](#)
 - GMLBaseUnit, [37](#)
 - GMLCartesianCS, [38](#)
 - GMLCompoundCRS, [38](#)
 - GMLConventionalUnit, [39](#)
 - GMLConversion, [40](#)
 - GMLCoordinateSystemAxis, [41](#)
 - GMLCOVAbstractCoverage, [42](#)

- GMLCOVExtension, [43](#)
- GMLCylindricalCS, [43](#)
- GMLDefinition, [44](#)
- GMLDerivedCRS, [45](#)
- GMLDerivedUnit, [46](#)
- GMLElement, [47](#)
- GMLEllipsoidalCS, [47](#)
- GMLEnvelope, [48](#)
- GMLEnvelopeWithTimePeriod, [49](#)
- GMLGeneralGridAxis, [50](#)
- GMLGeodeticCRS, [51](#)
- GMLGrid, [52](#)
- GMLGridCoverage, [53](#)
- GMLGridFunction, [53](#)
- GMLLinearCS, [54](#)
- GMLLinearRing, [55](#)
- GMLLineString, [56](#)
- GMLMultiCurve, [57](#)
- GMLMultiCurveCoverage, [58](#)
- GMLMultiPoint, [58](#)
- GMLMultiPointCoverage, [59](#)
- GMLMultiSolidCoverage, [60](#)
- GMLMultiSurface, [61](#)
- GMLMultiSurfaceCoverage, [62](#)
- GMLObliqueCartesianCS, [63](#)
- GMLOperationMethod, [63](#)
- GMLOperationParameter, [64](#)
- GMLOperationParameterGroup, [65](#)
- GMLParameterValue, [66](#)
- GMLParameterValueGroup, [67](#)
- GMLPoint, [68](#)
- GMLPolarCS, [69](#)
- GMLPolygon, [70](#)
- GMLProjectedCRS, [71](#)
- GMLRectifiedGrid, [72](#)
- GMLRectifiedGridCoverage, [73](#)
- GMLReferenceableGridByArray, [74](#)
- GMLReferenceableGridByTransformation, [74](#)
- GMLReferenceableGridByVectors, [75](#)
- GMLSphericalCS, [76](#)
- GMLTemporalCRS, [77](#)
- GMLTemporalCS, [77](#)
- GMLTimeCS, [78](#)
- GMLUnitDefinition, [80](#)
- GMLUserDefinedCS, [81](#)
- GMLVerticalCRS, [82](#)
- GMLVerticalCS, [83](#)
- * GeneralGridAxis**
 - GMLGeneralGridAxis, [50](#)
- * GeneralParameterValue**
 - GMLAbstractGeneralParameterValue, [25](#)
- * Geometric**
 - GMLAbstractGeometricAggregate, [26](#)
 - GMLAbstractGeometricPrimitive, [26](#)
 - GMLAbstractRing, [31](#)
- * Geometry**
 - GMLAbstractGeometry, [27](#)
 - GMLAbstractImplicitGeometry, [29](#)
- * Georeferenceable**
 - ISOImageryGeoreferenceable, [214](#)
- * GridFunction**
 - GMLGridFunction, [53](#)
- * Grid**
 - GMLAbstractReferenceableGrid, [30](#)
 - GMLGrid, [52](#)
 - GMLRectifiedGrid, [72](#)
 - GMLReferenceableGridByArray, [74](#)
 - GMLReferenceableGridByTransformation, [74](#)
 - GMLReferenceableGridByVectors, [75](#)
- * INSPIRE**
 - INSPIREMetadataValidator, [84](#)
- * ISO**
 - GMLAbstractCoordinateOperation, [17](#)
 - GMLAbstractCoordinateSystem, [18](#)
 - GMLAbstractCoverage, [19](#)
 - GMLAbstractCRS, [20](#)
 - GMLAbstractCurve, [20](#)
 - GMLAbstractDiscreteCoverage, [21](#)
 - GMLAbstractFeature, [22](#)
 - GMLAbstractGeneralConversion, [23](#)
 - GMLAbstractGeneralDerivedCRS, [23](#)
 - GMLAbstractGeneralOperationParameter, [24](#)
 - GMLAbstractGeneralParameterValue, [25](#)
 - GMLAbstractGeometricAggregate, [26](#)
 - GMLAbstractGeometricPrimitive, [26](#)
 - GMLAbstractGeometry, [27](#)
 - GMLAbstractGML, [28](#)
 - GMLAbstractImplicitGeometry, [29](#)
 - GMLAbstractObject, [29](#)
 - GMLAbstractReferenceableGrid, [30](#)
 - GMLAbstractRing, [31](#)

- GMLAbstractSingleCRS, [32](#)
- GMLAbstractSingleOperation, [32](#)
- GMLAbstractSurface, [33](#)
- GMLAbstractTimeGeometricPrimitive, [34](#)
- GMLAbstractTimeObject, [34](#)
- GMLAbstractTimePrimitive, [35](#)
- GMLAffineCS, [36](#)
- GMLBaseUnit, [37](#)
- GMLCartesianCS, [38](#)
- GMLCompoundCRS, [38](#)
- GMLConventionalUnit, [39](#)
- GMLConversion, [40](#)
- GMLCoordinateSystemAxis, [41](#)
- GMLCylindricalCS, [43](#)
- GMLDefinition, [44](#)
- GMLDerivedCRS, [45](#)
- GMLDerivedUnit, [46](#)
- GMLElement, [47](#)
- GMLEllipsoidalCS, [47](#)
- GMLEnvelope, [48](#)
- GMLEnvelopeWithTimePeriod, [49](#)
- GMLGeodeticCRS, [51](#)
- GMLGrid, [52](#)
- GMLGridCoverage, [53](#)
- GMLGridFunction, [53](#)
- GMLLinearCS, [54](#)
- GMLLinearRing, [55](#)
- GMLLineString, [56](#)
- GMLMultiCurve, [57](#)
- GMLMultiCurveCoverage, [58](#)
- GMLMultiPoint, [58](#)
- GMLMultiPointCoverage, [59](#)
- GMLMultiSolidCoverage, [60](#)
- GMLMultiSurface, [61](#)
- GMLMultiSurfaceCoverage, [62](#)
- GMLObliqueCartesianCS, [63](#)
- GMLOperationMethod, [63](#)
- GMLOperationParameter, [64](#)
- GMLOperationParameterGroup, [65](#)
- GMLParameterValue, [66](#)
- GMLParameterValueGroup, [67](#)
- GMLPoint, [68](#)
- GMLPolarCS, [69](#)
- GMLPolygon, [70](#)
- GMLProjectedCRS, [71](#)
- GMLRectifiedGrid, [72](#)
- GMLRectifiedGridCoverage, [73](#)
- GMLReferenceableGridByArray, [74](#)
- GMLReferenceableGridByTransformation, [74](#)
- GMLReferenceableGridByVectors, [75](#)
- GMLSphericalCS, [76](#)
- GMLTemporalCRS, [77](#)
- GMLTemporalCS, [77](#)
- GMLTimeCS, [78](#)
- GMLTimeInstant, [79](#)
- GMLTimePeriod, [79](#)
- GMLUnitDefinition, [80](#)
- GMLUserDefinedCS, [81](#)
- GMLVerticalCRS, [82](#)
- GMLVerticalCS, [83](#)
- ISOAbsoluteExternalPositionalAccuracy, [85](#)
- ISOAbstractAggregate, [86](#)
- ISOAbstractCarrierOfCharacteristics, [87](#)
- ISOAbstractCatalogue, [88](#)
- ISOAbstractCompleteness, [89](#)
- ISOAbstractGenericName, [90](#)
- ISOAbstractLogicalConsistency, [90](#)
- ISOAbstractObject, [91](#)
- ISOAbstractPositionalAccuracy, [93](#)
- ISOAbstractPropertyType, [94](#)
- ISOAbstractReferenceSystem, [95](#)
- ISOAbstractResult, [96](#)
- ISOAbstractTemporalAccuracy, [96](#)
- ISOAbstractThematicAccuracy, [97](#)
- ISOAccuracyOfATimeMeasurement, [98](#)
- ISOAddress, [99](#)
- ISOAggregateInformation, [100](#)
- ISOAnchor, [102](#)
- ISOAngle, [103](#)
- ISOApplicationSchemaInformation, [103](#)
- ISOAssociation, [104](#)
- ISOAssociationRole, [105](#)
- ISOAssociationType, [106](#)
- ISOAttributes, [107](#)
- ISOBand, [107](#)
- ISOBaseBoolean, [109](#)
- ISOBaseCharacterString, [110](#)
- ISOBaseDate, [111](#)
- ISOBaseDateTime, [111](#)
- ISOBaseDecimal, [112](#)
- ISOBaseInteger, [113](#)

- ISOBaseReal, [114](#)
- ISOBinary, [114](#)
- ISOBinding, [115](#)
- ISOBoundAssociationRole, [116](#)
- ISOBoundFeatureAttribute, [117](#)
- ISOBoundingPolygon, [117](#)
- ISOBrowseGraphic, [118](#)
- ISOCarrierOfCharacteristics, [119](#)
- ISOCellGeometry, [120](#)
- ISOCharacterSet, [121](#)
- ISOCitation, [121](#)
- ISOCitationSeries, [124](#)
- ISOClassification, [125](#)
- ISOCodelist, [125](#)
- ISOCodelistValue, [126](#)
- ISOCompletenessCommission, [127](#)
- ISOCompletenessOmission, [128](#)
- ISOConceptualConsistency, [129](#)
- ISOConformanceResult, [131](#)
- ISOConstraint, [132](#)
- ISOConstraints, [133](#)
- ISOContact, [134](#)
- ISOContentInformation, [135](#)
- ISOCountry, [136](#)
- ISOCoupledResource, [136](#)
- ISOCouplingType, [137](#)
- ISOCoverageContentType, [138](#)
- ISOCoverageDescription, [139](#)
- ISODataFile, [140](#)
- ISODataIdentification, [141](#)
- ISODataQuality, [146](#)
- ISODataQualityAbstractElement, [148](#)
- ISODataset, [149](#)
- ISODatatype, [150](#)
- ISODate, [151](#)
- ISODateType, [152](#)
- ISODCPList, [153](#)
- ISODefinitionReference, [153](#)
- ISODefinitionSource, [154](#)
- ISODigitalTransferOptions, [155](#)
- ISODimension, [156](#)
- ISODimensionNameType, [157](#)
- ISODistance, [158](#)
- ISODistribution, [158](#)
- ISODistributionUnits, [160](#)
- ISODistributor, [160](#)
- ISODomainConsistency, [162](#)
- ISOElementSequence, [163](#)
- ISOEvaluationMethodType, [164](#)
- ISOExtendedElementInformation, [165](#)
- ISOExtent, [167](#)
- ISOFeatureAssociation, [168](#)
- ISOFeatureAttribute, [169](#)
- ISOFeatureCatalogue, [170](#)
- ISOFeatureCatalogueDescription, [173](#)
- ISOFeatureOperation, [174](#)
- ISOFeatureType, [175](#)
- ISOFileName, [177](#)
- ISOFormat, [178](#)
- ISOFormatConsistency, [180](#)
- ISOFreeText, [181](#)
- ISOGeographicBoundingBox, [182](#)
- ISOGeographicDescription, [183](#)
- ISOGeographicExtent, [184](#)
- ISOGeometricObjects, [184](#)
- ISOGeometricObjectType, [185](#)
- ISOGeorectified, [186](#)
- ISOGeoreferenceable, [187](#)
- ISOGriddedDataPositionalAccuracy, [189](#)
- ISOGridSpatialRepresentation, [190](#)
- ISOHierarchyLevel, [192](#)
- ISOIdentification, [192](#)
- ISOIdentifier, [194](#)
- ISOImageDescription, [195](#)
- ISOImageryAbstractGeolocationInformation, [198](#)
- ISOImageryAcquisitionInformation, [198](#)
- ISOImageryAlgorithm, [200](#)
- ISOImageryBand, [201](#)
- ISOImageryBandDefinition, [203](#)
- ISOImageryContext, [204](#)
- ISOImageryCoverageDescription, [205](#)
- ISOImageryCoverageResult, [206](#)
- ISOImageryEnvironmentalRecord, [207](#)
- ISOImageryEvent, [208](#)
- ISOImageryGCP, [210](#)
- ISOImageryGCPCollection, [211](#)
- ISOImageryGeometryType, [212](#)
- ISOImageryGeorectified, [213](#)
- ISOImageryGeoreferenceable, [214](#)
- ISOImageryImageDescription, [215](#)
- ISOImageryInstrument, [216](#)
- ISOImageryMetadata, [218](#)

- ISOImageryNominalResolution, [225](#)
- ISOImageryObjective, [227](#)
- ISOImageryObjectiveType, [229](#)
- ISOImageryOperation, [230](#)
- ISOImageryOperationType, [231](#)
- ISOImageryPlan, [232](#)
- ISOImageryPlatform, [234](#)
- ISOImageryPlatformPass, [236](#)
- ISOImageryPolarisationOrientation, [237](#)
- ISOImageryPriority, [238](#)
- ISOImageryProcessing, [239](#)
- ISOImageryProcessStep, [241](#)
- ISOImageryProcessStepReport, [243](#)
- ISOImageryRangeElementDescription, [244](#)
- ISOImageryRequestedDate, [245](#)
- ISOImageryRequirement, [246](#)
- ISOImagerySensorType, [249](#)
- ISOImagerySequence, [250](#)
- ISOImagerySource, [251](#)
- ISOImageryTransferFunctionType, [252](#)
- ISOImageryTrigger, [253](#)
- ISOImageryUsability, [254](#)
- ISOImagingCondition, [254](#)
- ISOInheritanceRelation, [255](#)
- ISOInitiative, [256](#)
- ISOInitiativeType, [257](#)
- ISOKeywords, [258](#)
- ISOKeywordType, [259](#)
- ISOLanguage, [260](#)
- ISOLegalConstraints, [261](#)
- ISOLength, [263](#)
- ISOLineage, [263](#)
- ISOListedValue, [265](#)
- ISOLocale, [266](#)
- ISOLocaleContainer, [267](#)
- ISOLocalisedCharacterString, [268](#)
- ISOLocalName, [269](#)
- ISOMaintenanceFrequency, [269](#)
- ISOMaintenanceInformation, [270](#)
- ISOMeasure, [271](#)
- ISOMedium, [272](#)
- ISOMediumFormat, [273](#)
- ISOMediumName, [274](#)
- ISOMemberName, [274](#)
- ISOMetadata, [275](#)
- ISOMetadataExtensionInformation, [283](#)
- ISOMetadataNamespace, [285](#)
- ISOMetaIdentifier, [286](#)
- ISOMimeFileType, [286](#)
- ISOMultiplicity, [287](#)
- ISOMultiplicityRange, [288](#)
- ISONonQuantitativeAttributeAccuracy, [289](#)
- ISOobligation, [290](#)
- ISOOnlineFunction, [291](#)
- ISOOnlineResource, [292](#)
- ISOOperationMetadata, [293](#)
- ISOOtherAggregate, [294](#)
- ISOParameter, [295](#)
- ISOParameterDirection, [296](#)
- ISOPixelOrientation, [297](#)
- ISOPlatform, [298](#)
- ISOPortrayalCatalogueReference, [299](#)
- ISOPresentationForm, [300](#)
- ISOProcessStep, [301](#)
- ISOProductionSeries, [302](#)
- ISOPropertyType, [303](#)
- ISOQuantitativeAttributeAccuracy, [304](#)
- ISOQuantitativeResult, [305](#)
- ISORangeDimension, [306](#)
- ISORecord, [307](#)
- ISORecordType, [308](#)
- ISOReferenceIdentifier, [309](#)
- ISOReferenceSystem, [310](#)
- ISORepresentativeFraction, [311](#)
- ISOResolution, [312](#)
- ISOResponsibleParty, [313](#)
- ISORestriction, [314](#)
- ISORole, [315](#)
- ISORoleType, [316](#)
- ISOScale, [317](#)
- ISOScope, [317](#)
- ISOScopeDescription, [318](#)
- ISOScopedName, [319](#)
- ISOSecurityConstraints, [320](#)
- ISOSensor, [321](#)
- ISOSeries, [322](#)
- ISOServiceIdentification, [323](#)
- ISOSource, [325](#)
- ISOspatialRepresentation, [327](#)

- ISO SpatialRepresentationType, 328
- ISO SpatialTemporalExtent, 329
- ISO SRVServiceIdentification, 330
- ISO StandardOrderProcess, 335
- ISO Status, 337
- ISO StereoMate, 337
- ISO Telephone, 338
- ISO TemporalConsistency, 339
- ISO TemporalExtent, 340
- ISO TemporalValidity, 341
- ISO ThematicClassificationCorrectness, 342
- ISO TimePeriod, 343
- ISO TopicCategory, 344
- ISO TopologicalConsistency, 345
- ISO TopologyLevel, 346
- ISO TypeName, 347
- ISO UnlimitedInteger, 348
- ISO URL, 348
- ISO Usage, 349
- ISO VectorSpatialRepresentation, 350
- ISO VerticalExtent, 351
- * **Implicit**
 - GML AbstractImplicitGeometry, 29
- * **InheritanceRelation**
 - ISO InheritanceRelation, 255
- * **LineString**
 - GML LineString, 56
- * **LinearRing**
 - GML LinearRing, 55
- * **MultiCurve**
 - GML MultiCurve, 57
- * **MultiPoint**
 - GML MultiPoint, 58
- * **MultiSurface**
 - GML MultiSurface, 61
- * **OGC**
 - GML GeneralGridAxis, 50
- * **ObjectiveType**
 - ISO ImageryObjectiveType, 229
- * **Obligation**
 - ISO Obligation, 290
- * **ObliqueCartesian**
 - GML ObliqueCartesianCS, 63
- * **OnLineFunction**
 - ISO OnLineFunction, 291
- * **OperationMetadata**
 - ISO OperationMetadata, 293
- * **Operation**
 - GML AbstractCoordinateOperation, 17
 - GML AbstractSingleOperation, 32
 - ISO ImageryOperation, 230
 - ISO ImageryOperationType, 231
- * **Plan**
 - ISO ImageryPlan, 232
- * **PlatformPass**
 - ISO ImageryPlatformPass, 236
- * **Point**
 - GML Point, 68
 - GML Polygon, 70
- * **Polarisation**
 - ISO ImageryPolarisationOrientation, 237
- * **Primitive**
 - GML AbstractGeometricPrimitive, 26
- * **ProcessStepReport**
 - ISO ImageryProcessStepReport, 243
- * **Quantitative**
 - ISO QuantitativeResult, 305
- * **Restriction**
 - ISO Restriction, 314
- * **Sensor**
 - ISO Sensor, 321
- * **Series**
 - ISO Series, 322
- * **StandardOrderProcess**
 - ISO StandardOrderProcess, 335
- * **Surface**
 - GML AbstractSurface, 33
- * **URL**
 - ISO URL, 348
- * **UserDefined**
 - GML UserDefinedCS, 81
- * **absolute**
 - ISO AbsoluteExternalPositionalAccuracy, 85
- * **abstract**
 - GML AbstractCoordinateOperation, 17
 - GML AbstractCoordinateSystem, 18
 - GML AbstractCoverage, 19
 - GML AbstractCRS, 20
 - GML AbstractDiscreteCoverage, 21
 - GML AbstractFeature, 22
 - GML AbstractGeneralConversion, 23
 - GML AbstractGeneralDerivedCRS, 23

- GMLAbstractGeneralOperationParameter, [24](#)
- GMLAbstractSingleCRS, [32](#)
- GMLAbstractSingleOperation, [32](#)
- GMLAbstractTimeObject, [34](#)
- GMLAbstractTimePrimitive, [35](#)
- ISOAbstractAggregate, [86](#)
- ISOAbstractCarrierOfCharacteristics, [87](#)
- ISOAbstractCatalogue, [88](#)
- ISOAbstractCompleteness, [89](#)
- ISOAbstractGenericName, [90](#)
- ISOAbstractLogicalConsistency, [90](#)
- ISOAbstractPositionalAccuracy, [93](#)
- ISOAbstractPropertyType, [94](#)
- ISOAbstractReferenceSystem, [95](#)
- ISOAbstractTemporalAccuracy, [96](#)
- ISOAbstractThematicAccuracy, [97](#)
- ISODataQualityAbstractElement, [148](#)
- ISOImageryAbstractGeolocationInformation, [198](#)
- * **accuracy**
 - ISOAbsoluteExternalPositionalAccuracy, [85](#)
 - ISOAbstractPositionalAccuracy, [93](#)
 - ISOAbstractTemporalAccuracy, [96](#)
 - ISOAbstractThematicAccuracy, [97](#)
 - ISOAccuracyOfATimeMeasurement, [98](#)
 - ISOGriddedDataPositionalAccuracy, [189](#)
 - ISONonQuantitativeAttributeAccuracy, [289](#)
 - ISOQuantitativeAttributeAccuracy, [304](#)
 - ISOThematicClassificationCorrectness, [342](#)
- * **address**
 - ISOAddress, [99](#)
 - ISOSpatialRepresentation, [327](#)
 - ISOVectorSpatialRepresentation, [350](#)
- * **affine**
 - GMLAffineCS, [36](#)
- * **aggregated**
 - ISOAggregateInformation, [100](#)
- * **aggregate**
 - ISOAbstractAggregate, [86](#)
 - ISOOtherAggregate, [294](#)
- * **algorithm**
 - ISOImageryAlgorithm, [200](#)
- * **anchor**
 - ISOAnchor, [102](#)
- * **angle**
 - ISOAngle, [103](#)
- * **application**
 - ISOApplicationSchemaInformation, [103](#)
- * **association**
 - ISOAssociation, [104](#)
 - ISOAssociationRole, [105](#)
 - ISOAssociationType, [106](#)
 - ISOFeatureAssociation, [168](#)
- * **attributes**
 - ISOAttributes, [107](#)
- * **attribute**
 - ISONonQuantitativeAttributeAccuracy, [289](#)
 - ISOQuantitativeAttributeAccuracy, [304](#)
- * **axis**
 - GMLCoordinateSystemAxis, [41](#)
- * **band**
 - ISOBand, [107](#)
 - ISOImageryBand, [201](#)
 - ISOImageryBandDefinition, [203](#)
- * **base**
 - GMLBaseUnit, [37](#)
- * **binary**
 - ISOBinary, [114](#)
- * **boolean**
 - ISOBaseBoolean, [109](#)
- * **bounding**
 - ISOBoundingPolygon, [117](#)
- * **browse**
 - ISOBrowseGraphic, [118](#)
 - ISOScopeDescription, [318](#)
- * **carrierOfCharacteristics**
 - ISOAbstractCarrierOfCharacteristics, [87](#)
 - ISOCarrierOfCharacteristics, [119](#)
- * **cartesian**
 - GMLCartesianCS, [38](#)
- * **catalogue**
 - ISOAbstractCatalogue, [88](#)
 - ISOFeatureCatalogue, [170](#)
 - ISOFeatureCatalogueDescription, [170](#)

- 173
- ISOPortrayalCatalogueReference, 299
- * **category**
 - ISOTopicCategory, 344
- * **cell**
 - ISOCellGeometry, 120
- * **characterSet**
 - ISOCharacterSet, 121
- * **character**
 - ISOBaseCharacterString, 110
 - ISOLocalisedCharacterString, 268
- * **charset**
 - ISOCharacterSet, 121
 - ISOTopologyLevel, 346
- * **citation**
 - ISOCitation, 121
 - ISOCitationSeries, 124
- * **classification**
 - ISOThematicClassificationCorrectness, 342
- * **codelist**
 - ISOCodelist, 125
- * **code**
 - ISOCodelistValue, 126
- * **collection**
 - ISOImageryGCP, 210
 - ISOImageryGCPCollection, 211
- * **commission**
 - ISOCompletenessCommission, 127
- * **completeness**
 - ISOAbstractCompleteness, 89
 - ISOCompletenessCommission, 127
 - ISOCompletenessOmission, 128
- * **compound**
 - GMLCompoundCRS, 38
- * **conceptual**
 - ISOConceptualConsistency, 129
- * **condition**
 - ISOImagingCondition, 254
- * **conformance**
 - ISOConformanceResult, 131
- * **consistency**
 - ISOAbstractLogicalConsistency, 90
 - ISOConceptualConsistency, 129
 - ISODomainConsistency, 162
 - ISOFormatConsistency, 180
 - ISOTemporalConsistency, 339
 - ISOTopologicalConsistency, 345
- * **constraints**
 - ISOConstraints, 133
 - ISOLegalConstraints, 261
 - ISOSecurityConstraints, 320
- * **constraint**
 - ISOConstraint, 132
- * **contact**
 - ISOContact, 134
- * **container**
 - ISOLocaleContainer, 267
- * **content**
 - ISOContentInformation, 135
- * **context**
 - ISOImageryContext, 204
- * **conventional**
 - GMLConventionalUnit, 39
- * **conversion**
 - GMLAbstractGeneralConversion, 23
 - GMLConversion, 40
- * **convert**
 - pivot_format, 352
- * **coordinate**
 - GMLAbstractCoordinateOperation, 17
 - GMLAbstractCoordinateSystem, 18
 - GMLAffineCS, 36
 - GMLCartesianCS, 38
 - GMLCoordinateSystemAxis, 41
 - GMLCylindricalCS, 43
 - GMLEllipsoidalCS, 47
 - GMLLinearCS, 54
 - GMLObliqueCartesianCS, 63
 - GMLPolarCS, 69
 - GMLSphericalCS, 76
 - GMLTemporalCS, 77
 - GMLTimeCS, 78
 - GMLUserDefinedCS, 81
 - GMLVerticalCS, 83
- * **correctness**
 - ISOThematicClassificationCorrectness, 342
- * **country**
 - ISOCountry, 136
- * **coupling**
 - ISOCouplingType, 137
- * **coverage**
 - GMLAbstractCoverage, 19
 - GMLAbstractDiscreteCoverage, 21

- GMLGridCoverage, [53](#)
- GMLMultiCurveCoverage, [58](#)
- GMLMultiPointCoverage, [59](#)
- GMLMultiSolidCoverage, [60](#)
- GMLMultiSurfaceCoverage, [62](#)
- GMLRectifiedGridCoverage, [73](#)
- ISOCoverageDescription, [139](#)
- ISOImageDescription, [195](#)
- ISOImageryCoverageResult, [206](#)
- * **crs**
 - GMLAbstractGeneralDerivedCRS, [23](#)
 - GMLAbstractSingleCRS, [32](#)
 - GMLDerivedCRS, [45](#)
 - GMLGeodeticCRS, [51](#)
 - GMLProjectedCRS, [71](#)
 - GMLTemporalCRS, [77](#)
 - GMLVerticalCRS, [82](#)
- * **cylindrical**
 - GMLCylindricalCS, [43](#)
- * **dataset**
 - ISODataset, [149](#)
- * **data**
 - ISOAbsoluteExternalPositionalAccuracy, [85](#)
 - ISOAbstractCompleteness, [89](#)
 - ISOAbstractLogicalConsistency, [90](#)
 - ISOAbstractPositionalAccuracy, [93](#)
 - ISOAbstractResult, [96](#)
 - ISOAbstractTemporalAccuracy, [96](#)
 - ISOAbstractThematicAccuracy, [97](#)
 - ISOAccuracyOfATimeMeasurement, [98](#)
 - ISOCompletenessCommission, [127](#)
 - ISOCompletenessOmission, [128](#)
 - ISOConceptualConsistency, [129](#)
 - ISODataFile, [140](#)
 - ISODataIdentification, [141](#)
 - ISODataQuality, [146](#)
 - ISODataQualityAbstractElement, [148](#)
 - ISODomainConsistency, [162](#)
 - ISOFormatConsistency, [180](#)
 - ISOGriddedDataPositionalAccuracy, [189](#)
 - ISONonQuantitativeAttributeAccuracy, [289](#)
 - ISOQuantitativeAttributeAccuracy, [304](#)
 - ISOTemporalConsistency, [339](#)
 - ISOTemporalValidity, [341](#)
 - ISOThematicClassificationCorrectness, [342](#)
 - ISOTopologicalConsistency, [345](#)
- * **datetime**
 - ISOBaseDateTime, [111](#)
- * **datatype**
 - ISODataType, [152](#)
- * **date**
 - ISOBaseDate, [111](#)
 - ISODate, [151](#)
 - ISOImageryRequestedDate, [245](#)
- * **decimal**
 - ISOBaseDecimal, [112](#)
- * **definition**
 - GMLBaseUnit, [37](#)
 - GMLConventionalUnit, [39](#)
 - GMLDefinition, [44](#)
 - GMLDerivedUnit, [46](#)
 - GMLUnitDefinition, [80](#)
 - ISODefinitionReference, [153](#)
 - ISODefinitionSource, [154](#)
 - ISOImageryBandDefinition, [203](#)
- * **derived**
 - GMLDerivedCRS, [45](#)
 - GMLDerivedUnit, [46](#)
- * **description**
 - ISOCoverageDescription, [139](#)
 - ISOFeatureCatalogueDescription, [173](#)
 - ISOGeographicDescription, [183](#)
 - ISOImageDescription, [195](#)
 - ISOImageryCoverageDescription, [205](#)
 - ISOImageryImageDescription, [215](#)
 - ISOImageryRangeElementDescription, [244](#)
- * **dimension**
 - ISODimension, [156](#)
 - ISORangeDimension, [306](#)
- * **direction**
 - ISOParameterDirection, [296](#)
- * **discrete**
 - GMLAbstractDiscreteCoverage, [21](#)
- * **distance**
 - ISODistance, [158](#)
- * **distribution**
 - ISODigitalTransferOptions, [155](#)
 - ISODistribution, [158](#)
 - ISODistributionUnits, [160](#)

- * **distributor**
 - ISODistributor, 160
- * **domain**
 - ISODomainConsistency, 162
- * **element**
 - GMLElement, 47
 - ISOAbstractObject, 91
 - ISOCodelistValue, 126
 - ISODataQualityAbstractElement, 148
 - ISOExtendedElementInformation, 165
 - ISOImageryAcquisitionInformation, 198
 - ISOImageryMetadata, 218
 - ISOImageryRangeElementDescription, 244
 - ISOMetadata, 275
- * **ellipsoidal**
 - GMLEllipsoidalCS, 47
- * **environmental**
 - ISOImageryEnvironmentalRecord, 207
- * **event**
 - ISOImageryEvent, 208
- * **extended**
 - ISOExtendedElementInformation, 165
- * **extension**
 - ISOMetadataExtensionInformation, 283
- * **extent**
 - ISOExtent, 167
 - ISOGeographicBoundingBox, 182
 - ISOGeographicExtent, 184
 - ISOResolution, 312
 - ISOScope, 317
 - ISOspatialTemporalExtent, 329
 - ISOTemporalExtent, 340
 - ISOVerticalExtent, 351
- * **external**
 - ISOAbsoluteExternalPositionalAccuracy, 85
- * **feature**
 - GMLAbstractFeature, 22
 - ISOConstraint, 132
 - ISOFeatureAssociation, 168
 - ISOFeatureAttribute, 169
 - ISOFeatureCatalogue, 170
 - ISOFeatureCatalogueDescription, 173
 - ISOFeatureOperation, 174
 - ISOFeatureType, 175
- * **file**
 - ISODataFile, 140
 - ISOFileName, 177
 - ISOMimeFileType, 286
 - ISOResponsibleParty, 313
 - ISOTelephone, 338
- * **format**
 - ISOFormat, 178
 - ISOFormatConsistency, 180
 - ISOMediumFormat, 273
- * **form**
 - ISOPresentationForm, 300
- * **fraction**
 - ISORepresentativeFraction, 311
- * **freeText**
 - ISOFreeText, 181
- * **function**
 - ISOImageryTransferFunctionType, 252
- * **gcp**
 - ISOImageryGCP, 210
 - ISOImageryGCPCollection, 211
- * **general**
 - GMLAbstractGeneralConversion, 23
 - GMLAbstractGeneralOperationParameter, 24
- * **generic**
 - ISOAbstractGenericName, 90
- * **geodetic**
 - GMLGeodeticCRS, 51
- * **geographic**
 - ISOGeographicDescription, 183
- * **geolocation**
 - ISOImageryAbstractGeolocationInformation, 198
- * **geometric**
 - ISOGeometricObjectType, 185
- * **geometry**
 - ISOCellGeometry, 120
 - ISOGeometricObjects, 184
 - ISOImageryGeometryType, 212
- * **georectified**
 - ISOGeorectified, 186
 - ISOImageryGeorectified, 213
- * **georeferenceable**
 - ISOGeoreferenceable, 187
- * **graphic**

- ISOBrowseGraphic, 118
- ISOScopeDescription, 318
- * **gridded**
 - ISOGriddedDataPositionalAccuracy, 189
- * **grid**
 - GMLGridCoverage, 53
 - GMLRectifiedGridCoverage, 73
 - ISOGeoreferenceable, 187
 - ISOGridSpatialRepresentation, 190
- * **group**
 - GMLOperationParameterGroup, 65
 - GMLParameterValueGroup, 67
- * **hierarchyLevel**
 - ISOHierarchyLevel, 192
 - ISOMaintenanceFrequency, 269
- * **identification**
 - ISODataIdentification, 141
 - ISOIdentification, 192
 - ISOServiceIdentification, 323
 - ISOSRVServiceIdentification, 330
- * **identifier**
 - ISOIdentifier, 194
 - ISOMetaIdentifier, 286
 - ISOReferenceIdentifier, 309
 - ISOResponsibleParty, 313
 - ISOTelephone, 338
- * **imagery**
 - ISOImageryAbstractGeolocationInformation, 198
 - ISOImageryAcquisitionInformation, 198
 - ISOImageryAlgorithm, 200
 - ISOImageryBand, 201
 - ISOImageryBandDefinition, 203
 - ISOImageryContext, 204
 - ISOImageryCoverageDescription, 205
 - ISOImageryCoverageResult, 206
 - ISOImageryEnvironmentalRecord, 207
 - ISOImageryEvent, 208
 - ISOImageryGCP, 210
 - ISOImageryGCPCollection, 211
 - ISOImageryGeometryType, 212
 - ISOImageryGeoreferenceable, 214
 - ISOImageryImageDescription, 215
 - ISOImageryInstrument, 216
 - ISOImageryMetadata, 218
 - ISOImageryNominalResolution, 225
 - ISOImageryObjective, 227
 - ISOImageryObjectiveType, 229
 - ISOImageryOperation, 230
 - ISOImageryOperationType, 231
 - ISOImageryPlan, 232
 - ISOImageryPlatform, 234
 - ISOImageryPlatformPass, 236
 - ISOImageryPolarisationOrientation, 237
 - ISOImageryPriority, 238
 - ISOImageryProcessing, 239
 - ISOImageryProcessStep, 241
 - ISOImageryProcessStepReport, 243
 - ISOImageryRangeElementDescription, 244
 - ISOImageryRequestedDate, 245
 - ISOImageryRequirement, 246
 - ISOImagerySensorType, 249
 - ISOImagerySequence, 250
 - ISOImagerySource, 251
 - ISOImageryTransferFunctionType, 252
 - ISOImageryTrigger, 253
 - ISOImageryUsability, 254
- * **image**
 - ISOImageryCoverageDescription, 205
 - ISOImageryGeorectified, 213
 - ISOImageryImageDescription, 215
- * **imaging**
 - ISOImagingCondition, 254
- * **information**
 - ISOAggregateInformation, 100
 - ISOApplicationSchemaInformation, 103
 - ISOContentInformation, 135
 - ISOExtendedElementInformation, 165
 - ISOImageryAbstractGeolocationInformation, 198
 - ISOMaintenanceInformation, 270
 - ISOMetadataExtensionInformation, 283
- * **initiative**
 - ISOInitiativeType, 257
- * **initiative**
 - ISOInitiative, 256
- * **instant**
 - GMLTimeInstant, 79
- * **integer**

- ISOBaseInteger, 113
- ISOUnlimitedInteger, 348
- * **keywords**
 - ISOKeywords, 258
- * **keywordtype**
 - ISOKeywordType, 259
- * **language**
 - ISOLanguage, 260
- * **legal**
 - ISOLegalConstraints, 261
- * **length**
 - ISODistance, 158
 - ISOLength, 263
- * **level**
 - ISOTopologyLevel, 346
- * **lineage**
 - ISOLineage, 263
- * **linear**
 - GMLLinearCS, 54
- * **listed**
 - ISOListedValue, 265
- * **locale**
 - ISOLocale, 266
 - ISOLocaleContainer, 267
- * **localised**
 - ISOLocalisedCharacterString, 268
- * **local**
 - ISOLocalName, 269
- * **logger**
 - geometaLogger, 11
- * **logical**
 - ISOAbstractLogicalConsistency, 90
- * **maintenance**
 - ISOMaintenanceInformation, 270
- * **mapping**
 - pivot_format, 352
- * **mate**
 - ISOStereoMate, 337
- * **measurement**
 - ISOAccuracyOfATimeMeasurement, 98
- * **measure**
 - ISOAngle, 103
 - ISODistance, 158
 - ISOLength, 263
 - ISOMeasure, 271
 - ISOScale, 317
- * **medium**
 - ISOMedium, 272
 - ISOMediumFormat, 273
 - ISOMediumName, 274
- * **metadata**
 - INSPIREMetadataValidator, 84
 - ISOAbstractObject, 91
 - ISOImageryMetadata, 218
 - ISOMetadata, 275
 - ISOMetadataExtensionInformation, 283
 - ISOMetadataNamespace, 285
 - pivot_format, 352
- * **meta**
 - ISOMetaIdentifier, 286
- * **method**
 - GMLOperationMethod, 63
- * **mime**
 - ISOMimeFileType, 286
- * **multicurve**
 - GMLMultiCurveCoverage, 58
- * **multiplicity**
 - ISOMultiplicity, 287
 - ISOMultiplicityRange, 288
- * **multipoint**
 - GMLMultiPointCoverage, 59
- * **multisolid**
 - GMLMultiSolidCoverage, 60
- * **multisurface**
 - GMLMultiSurfaceCoverage, 62
- * **namespace**
 - ISOMetadataNamespace, 285
- * **name**
 - ISOAbstractGenericName, 90
 - ISOFileName, 177
 - ISOLocalName, 269
 - ISOMediumName, 274
 - ISOScopedName, 319
- * **nominal**
 - ISOImageryNominalResolution, 225
- * **nonquantitative**
 - ISONonQuantitativeAttributeAccuracy, 289
- * **objective**
 - ISOImageryObjective, 227
- * **objects**
 - ISOGeometricObjects, 184
- * **object**
 - GMLAbstractTimeObject, 34
 - ISOGeometricObjectType, 185

- * **omission**
 - ISOCompletenessOmission, 128
- * **online**
 - ISOOnlineResource, 292
- * **operation**
 - GMLAbstractGeneralOperationParameter, 24
 - GMLOperationMethod, 63
 - GMLOperationParameter, 64
 - GMLOperationParameterGroup, 65
 - ISOFeatureAttribute, 169
 - ISOFeatureOperation, 174
- * **orientation**
 - ISOImageryPolarisationOrientation, 237
 - ISOPixelOrientation, 297
- * **other**
 - ISOOtherAggregate, 294
- * **parameter**
 - GMLAbstractGeneralOperationParameter, 24
 - GMLOperationParameter, 64
 - GMLOperationParameterGroup, 65
 - GMLParameterValue, 66
 - GMLParameterValueGroup, 67
 - ISOParameter, 295
 - ISOParameterDirection, 296
- * **period**
 - GMLTimePeriod, 79
 - ISOTimePeriod, 343
- * **pixel**
 - ISOPixelOrientation, 297
- * **platform**
 - ISOImageryInstrument, 216
 - ISOImageryPlatform, 234
 - ISOPlatform, 298
- * **polar**
 - GMLPolarCS, 69
- * **polygon**
 - ISOBoundingPolygon, 117
- * **portrayal**
 - ISOPortrayalCatalogueReference, 299
- * **positional**
 - ISOAbsoluteExternalPositionalAccuracy, 85
 - ISOAbstractPositionalAccuracy, 93
 - ISOGriddedDataPositionalAccuracy, 189
- * **presentation**
 - ISOPresentationForm, 300
- * **primitive**
 - GMLAbstractTimeGeometricPrimitive, 34
 - GMLAbstractTimePrimitive, 35
- * **priority**
 - ISOImageryPriority, 238
- * **processing**
 - ISOImageryProcessing, 239
- * **process**
 - ISOImageryProcessStep, 241
 - ISOProcessStep, 301
- * **production**
 - ISOProductionSeries, 302
- * **projected**
 - GMLProjectedCRS, 71
- * **property**
 - ISOAbstractPropertyType, 94
 - ISOPropertyType, 303
- * **quality**
 - ISOAbsoluteExternalPositionalAccuracy, 85
 - ISOAbstractCompleteness, 89
 - ISOAbstractLogicalConsistency, 90
 - ISOAbstractPositionalAccuracy, 93
 - ISOAbstractResult, 96
 - ISOAbstractTemporalAccuracy, 96
 - ISOAbstractThematicAccuracy, 97
 - ISOAccuracyOfATimeMeasurement, 98
 - ISOCompletenessCommission, 127
 - ISOCompletenessOmission, 128
 - ISOConceptualConsistency, 129
 - ISODataQuality, 146
 - ISODataQualityAbstractElement, 148
 - ISODomainConsistency, 162
 - ISOFormatConsistency, 180
 - ISOGriddedDataPositionalAccuracy, 189
 - ISONonQuantitativeAttributeAccuracy, 289
 - ISOQuantitativeAttributeAccuracy, 304
 - ISOTemporalConsistency, 339
 - ISOTemporalValidity, 341
 - ISOThematicClassificationCorrectness, 342

- ISOTopologicalConsistency, 345
- * **quantitative**
 - ISOQuantitativeAttributeAccuracy, 304
- * **range**
 - ISOImageryRangeElementDescription, 244
 - ISOMultiplicityRange, 288
 - ISORangeDimension, 306
- * **real**
 - ISOBaseReal, 114
- * **record**
 - ISOElementSequence, 163
 - ISOImageryEnvironmentalRecord, 207
 - ISOMemberName, 274
 - ISORecord, 307
 - ISORecordType, 308
- * **rectified**
 - GMLRectifiedGridCoverage, 73
- * **reference**
 - ISOAbstractReferenceSystem, 95
 - ISODefinitionReference, 153
 - ISOPortrayalCatalogueReference, 299
 - ISOReferenceIdentifier, 309
 - ISOReferenceSystem, 310
- * **representation**
 - ISOGeoreferenceable, 187
 - ISOGridSpatialRepresentation, 190
 - ISOSpatialRepresentationType, 328
- * **representative**
 - ISORepresentativeFraction, 311
- * **requested**
 - ISOImageryRequestedDate, 245
- * **requirement**
 - ISOImageryRequirement, 246
- * **resolution**
 - ISOImageryNominalResolution, 225
- * **resource**
 - ISOOnlineResource, 292
- * **result**
 - ISOAbstractResult, 96
 - ISOConformanceResult, 131
 - ISOImageryCoverageResult, 206
 - ISOQuantitativeResult, 305
- * **ring**
 - GMLAbstractRing, 31
- * **roleType**
 - ISORoleType, 316
- * **role**
 - ISOAssociationRole, 105
 - ISORole, 315
- * **scale**
 - ISOScale, 317
- * **schema**
 - ISOApplicationSchemaInformation, 103
- * **scoped**
 - ISOScopedName, 319
- * **security**
 - ISOSecurityConstraints, 320
- * **sensor**
 - ISOImagerySensorType, 249
- * **sequence**
 - ISOImagerySequence, 250
- * **series**
 - ISOCitationSeries, 124
 - ISOProductionSeries, 302
- * **service**
 - ISOServiceIdentification, 323
 - ISOSRVServiceIdentification, 330
- * **single**
 - GMLAbstractGeneralDerivedCRS, 23
 - GMLAbstractSingleCRS, 32
 - GMLAbstractSingleOperation, 32
- * **source**
 - ISODefinitionSource, 154
 - ISOImagerySource, 251
 - ISOSource, 325
- * **spatialtemporal**
 - ISOSpatialTemporalExtent, 329
- * **spatial**
 - ISOGeoreferenceable, 187
 - ISOGridSpatialRepresentation, 190
 - ISOSpatialRepresentationType, 328
- * **spherical**
 - GMLSphericalCS, 76
- * **status**
 - ISOStatus, 337
- * **step**
 - ISOImageryProcessStep, 241
 - ISOProcessStep, 301
- * **stereo**
 - ISOStereoMate, 337
- * **string**
 - ISOBaseCharacterString, 110

- ISOLocalisedCharacterString, 268
- * **system**
 - GMLAbstractCoordinateSystem, 18
 - GMLAffineCS, 36
 - GMLCartesianCS, 38
 - GMLCoordinateSystemAxis, 41
 - GMLCylindricalCS, 43
 - GMLEllipsoidalCS, 47
 - GMLLinearCS, 54
 - GMLObliqueCartesianCS, 63
 - GMLPolarCS, 69
 - GMLSphericalCS, 76
 - GMLTemporalCS, 77
 - GMLTimeCS, 78
 - GMLUserDefinedCS, 81
 - GMLVerticalCS, 83
 - ISOAbstractReferenceSystem, 95
 - ISOReferenceSystem, 310
- * **temporal**
 - GMLAbstractTimeGeometricPrimitive, 34
 - GMLTemporalCRS, 77
 - GMLTemporalCS, 77
 - ISOAbstractTemporalAccuracy, 96
 - ISOAccuracyOfATimeMeasurement, 98
 - ISOTemporalConsistency, 339
 - ISOTemporalExtent, 340
 - ISOTemporalValidity, 341
- * **thematic**
 - ISOAbstractThematicAccuracy, 97
 - ISOThematicClassificationCorrectness, 342
- * **time**
 - GMLAbstractTimeObject, 34
 - GMLAbstractTimePrimitive, 35
 - GMLTimeCS, 78
 - GMLTimeInstant, 79
 - GMLTimePeriod, 79
 - ISOAccuracyOfATimeMeasurement, 98
 - ISOTimePeriod, 343
- * **topic**
 - ISOTopicCategory, 344
- * **topological**
 - ISOTopologicalConsistency, 345
- * **topology**
 - ISOTopologyLevel, 346
- * **transfer**
 - ISOImageryTransferFunctionType, 252
- * **trigger**
 - ISOImageryTrigger, 253
- * **typename**
 - ISOTypeName, 347
- * **type**
 - ISOAbstractPropertyType, 94
 - ISOAssociationType, 106
 - ISOCouplingType, 137
 - ISOFeatureType, 175
 - ISOGeometricObjectType, 185
 - ISOImageryGeometryType, 212
 - ISOImageryOperationType, 231
 - ISOImagerySensorType, 249
 - ISOImageryTransferFunctionType, 252
 - ISOInitiativeType, 257
 - ISOMimeType, 286
 - ISOPropertyType, 303
 - ISOSpatialRepresentationType, 328
- * **units**
 - ISODistributionUnits, 160
- * **unit**
 - GMLBaseUnit, 37
 - GMLConventionalUnit, 39
 - GMLDerivedUnit, 46
 - GMLUnitDefinition, 80
- * **unlimited**
 - ISOUnlimitedInteger, 348
- * **usability**
 - ISOImageryUsability, 254
- * **usage**
 - ISOUsage, 349
- * **validation**
 - INSPIREMetadataValidator, 84
- * **validator**
 - INSPIREMetadataValidator, 84
- * **validity**
 - ISOTemporalValidity, 341
- * **value**
 - GMLParameterValue, 66
 - GMLParameterValueGroup, 67
 - ISOListedValue, 265
- * **vertical**
 - GMLVerticalCRS, 82
 - GMLVerticalCS, 83
 - ISOVerticalExtent, 351
- cacheISOClasses, 9, 9

- character, [79–81](#), [88](#), [94](#), [99](#), [104](#), [115](#), [118](#),
[122](#), [124](#), [126](#), [131–133](#), [137](#), [139](#),
[142](#), [148](#), [154](#), [155](#), [165](#), [169](#), [174](#),
[175](#), [179](#), [186](#), [188](#), [193](#), [195](#), [200](#),
[217](#), [227](#), [230](#), [234](#), [239](#), [243](#), [244](#),
[255](#), [258](#), [261](#), [264–267](#), [269–272](#),
[275](#), [276](#), [292](#), [293](#), [295](#), [301](#),
[306–309](#), [313](#), [319](#), [320](#), [326](#), [330](#),
[336](#), [338](#), [347](#), [349–351](#)
- convert_metadata, [9](#)
- data.frame, [126](#)
- Date, [79](#), [122](#), [276](#)
- geometa, [10](#)
- geometa-package (geometa), [10](#)
- geometa_coverae (geometa_coverage), [11](#)
- geometa_coverage, [11](#)
- geometaLogger, [11](#)
- getGeometaOption, [12](#)
- getISOClasses, [9](#), [13](#)
- getISOCodelist, [13](#)
- getISOCodelists, [14](#)
- getISOInternalCodelists, [14](#)
- getISOMetadataNamespace, [15](#)
- getISOMetadataNamespaces, [15](#)
- getISOMetadataSchemas, [16](#)
- getMappingFormats, [10](#), [16](#)
- getMappings, [9](#), [10](#), [17](#)
- GMLAbstractCoordinateOperation, [17](#)
- GMLAbstractCoordinateSystem, [18](#)
- GMLAbstractCoverage, [19](#)
- GMLAbstractCRS, [20](#)
- GMLAbstractCurve, [20](#)
- GMLAbstractDiscreteCoverage, [21](#)
- GMLAbstractFeature, [22](#)
- GMLAbstractGeneralConversion, [23](#)
- GMLAbstractGeneralDerivedCRS, [23](#)
- GMLAbstractGeneralOperationParameter,
[24](#)
- GMLAbstractGeneralParameterValue, [25](#),
[40](#)
- GMLAbstractGeometricAggregate, [26](#)
- GMLAbstractGeometricPrimitive, [26](#)
- GMLAbstractGeometry, [27](#), [118](#)
- GMLAbstractGML, [28](#)
- GMLAbstractImplicitGeometry, [29](#)
- GMLAbstractObject, [29](#)
- GMLAbstractReferenceableGrid, [30](#)
- GMLAbstractRing, [31](#)
- GMLAbstractSingleCRS, [32](#)
- GMLAbstractSingleOperation, [32](#)
- GMLAbstractSurface, [33](#)
- GMLAbstractTimeGeometricPrimitive, [34](#)
- GMLAbstractTimeObject, [34](#)
- GMLAbstractTimePrimitive, [35](#)
- GMLAffineCS, [36](#)
- GMLBaseUnit, [37](#)
- GMLCartesianCS, [38](#), [51](#), [71](#)
- GMLCompoundCRS, [38](#)
- GMLConventionalUnit, [39](#)
- GMLConversion, [40](#)
- GMLCoordinateSystemAxis, [41](#)
- GMLCOVAbstractCoverage, [42](#)
- GMLCOVExtension, [43](#)
- GMLCylindricalCS, [43](#)
- GMLDefinition, [44](#)
- GMLDerivedCRS, [45](#)
- GMLDerivedUnit, [46](#)
- GMLElement, [17](#), [28](#), [39–41](#), [45](#), [47](#), [49](#), [50](#),
[64–66](#)
- GMLEllipsoidalCS, [47](#), [51](#)
- GMLEnvelope, [48](#)
- GMLEnvelopeWithTimePeriod, [49](#)
- GMLGeneralGridAxis, [50](#)
- GMLGeodeticCRS, [51](#), [71](#)
- GMLGrid, [52](#)
- GMLGridCoverage, [53](#)
- GMLGridFunction, [53](#)
- GMLLinearCS, [54](#)
- GMLLinearRing, [55](#)
- GMLLineString, [56](#)
- GMLMultiCurve, [57](#)
- GMLMultiCurveCoverage, [58](#)
- GMLMultiPoint, [58](#)
- GMLMultiPointCoverage, [59](#)
- GMLMultiSolidCoverage, [60](#)
- GMLMultiSurface, [61](#)
- GMLMultiSurfaceCoverage, [62](#)
- GMLObliqueCartesianCS, [63](#)
- GMLOperationMethod, [63](#)
- GMLOperationParameter, [64](#), [64](#), [65](#), [66](#)
- GMLOperationParameterGroup, [64](#), [65](#), [65](#),
[67](#)
- GMLParameterValue, [66](#), [67](#)
- GMLParameterValueGroup, [67](#)
- GMLPoint, [68](#), [186](#)

- GMLPolarCS, [69](#)
- GMLPolygon, [70](#)
- GMLProjectedCRS, [71](#)
- GMLRectifiedGrid, [72](#)
- GMLRectifiedGridCoverage, [73](#)
- GMLReferenceableGridByArray, [74](#)
- GMLReferenceableGridByTransformation, [74](#)
- GMLReferenceableGridByVectors, [75](#)
- GMLSphericalCS, [51](#), [76](#)
- GMLTemporalCRS, [77](#)
- GMLTemporalCS, [77](#)
- GMLTimeCS, [77](#), [78](#)
- GMLTimeInstant, [79](#), [329](#), [341](#)
- GMLTimePeriod, [79](#), [329](#), [341](#)
- GMLUnitDefinition, [80](#), [108](#), [169](#), [202](#), [306](#)
- GMLUserDefinedCS, [81](#)
- GMLVerticalCRS, [82](#), [351](#)
- GMLVerticalCS, [82](#), [83](#)

- INSPIREMetadataValidator, [84](#)
- integer, [108](#), [156](#), [165](#), [185](#), [191](#), [195](#), [272](#), [288](#), [311](#), [348](#)
- ISOAbsoluteExternalPositionalAccuracy, [85](#), [85](#)
- ISOAbstractAggregate, [86](#), [150](#), [295](#), [298](#), [303](#), [321](#), [322](#), [338](#)
- ISOAbstractCarrierOfCharacteristics, [87](#), [87](#), [94](#), [119](#), [303](#)
- ISOAbstractCatalogue, [88](#), [88](#), [171](#)
- ISOAbstractCompleteness, [89](#), [89](#)
- ISOAbstractGenericName, [90](#)
- ISOAbstractLogicalConsistency, [90](#), [91](#)
- ISOAbstractObject, [91](#)
- ISOAbstractPositionalAccuracy, [93](#), [93](#)
- ISOAbstractPropertyType, [94](#), [94](#), [304](#)
- ISOAbstractReferenceSystem, [95](#)
- ISOAbstractResult, [96](#), [207](#)
- ISOAbstractTemporalAccuracy, [96](#), [97](#)
- ISOAbstractThematicAccuracy, [97](#), [97](#)
- ISOAccuracyOfATimeMeasurement, [98](#), [98](#)
- ISOAddress, [99](#), [99](#), [134](#)
- ISOAggregateInformation, [100](#), [101](#), [193](#), [194](#)
- ISOAnchor, [102](#), [102](#)
- ISOAngle, [103](#), [156](#)
- ISOApplicationSchemaInformation, [103](#), [104](#)
- ISOAssociation, [104](#), [105](#)
- ISOAssociationRole, [105](#), [105](#), [168](#)
- ISOAssociationType, [100](#), [106](#), [106](#)
- ISOAttributes, [107](#)
- ISOBand, [107](#), [108](#), [202](#)
- ISOBaseBoolean, [109](#)
- ISOBaseCharacterString, [110](#), [208](#)
- ISOBaseDate, [111](#), [151](#)
- ISOBaseDateTime, [111](#)
- ISOBaseDecimal, [112](#)
- ISOBaseInteger, [113](#)
- ISOBaseReal, [114](#)
- ISOBinary, [104](#), [114](#), [115](#)
- ISOBinding, [115](#), [116](#)
- ISOBoundAssociationRole, [116](#), [116](#)
- ISOBoundFeatureAttribute, [117](#), [117](#)
- ISOBoundingPolygon, [117](#), [118](#)
- ISOBrowseGraphic, [118](#), [118](#), [142](#), [193](#), [194](#), [323](#), [331](#)
- ISOCarrierOfCharacteristics, [119](#), [119](#), [176](#)
- ISOCellGeometry, [120](#), [120](#), [188](#), [191](#)
- ISOCharacterSet, [121](#), [121](#), [142](#), [143](#), [266](#), [276](#)
- ISOCitation, [64](#), [100](#), [101](#), [104](#), [121](#), [122](#), [131](#), [142](#), [154](#), [173](#), [174](#), [188](#), [193](#), [195](#), [200](#), [217](#), [230](#), [232](#), [234](#), [239](#), [247](#), [258](#), [299](#), [323](#), [326](#), [330](#)
- ISOCitationSeries, [123](#), [124](#)
- ISOClassification, [125](#), [125](#), [320](#)
- ISOCodelist, [125](#), [126](#)
- ISOCodelistValue, [126](#)
- ISOCompletenessCommission, [127](#), [127](#)
- ISOCompletenessOmission, [128](#), [129](#)
- ISOConceptualConsistency, [129](#), [130](#)
- ISOConformanceResult, [131](#), [131](#), [148](#)
- ISOConstraint, [87](#), [94](#), [119](#), [132](#), [132](#), [303](#)
- ISOConstraints, [133](#), [176](#), [193](#), [194](#), [330](#), [331](#)
- ISOContact, [134](#), [313](#)
- ISOContentInformation, [135](#), [135](#)
- ISOCountry, [99](#), [136](#), [136](#), [266](#)
- ISOCoupledResource, [136](#), [137](#), [330](#), [332](#)
- ISOCouplingType, [137](#), [138](#), [330](#)
- ISOCoverageContentType, [138](#), [138](#), [196](#)
- ISOCoverageDescription, [139](#), [196](#), [207](#), [219](#), [276](#), [277](#)
- ISODataFile, [140](#), [141](#), [206](#), [207](#)
- ISODataIdentification, [122](#), [141](#), [143](#), [330](#), [332](#)

- ISODataQuality, *146, 146, 276*
- ISODataQualityAbstractElement, *85, 89, 91, 93, 97, 98, 127, 129, 130, 148, 162, 180, 189, 254, 289, 304, 339, 341, 343, 345*
- ISODataSet, *149, 150, 295, 298, 303, 321, 322, 338*
- ISODatatype, *150, 166*
- ISODate, *122, 151, 151, 267*
- ISODateType, *151, 152, 152, 165*
- ISODCPList, *153, 153, 293*
- ISODefinitionReference, *94, 153, 154, 176, 265, 304*
- ISODefinitionSource, *154, 154, 171*
- ISODigitalTransferOptions, *155, 155, 159*
- ISODimension, *156, 156, 187, 188, 191*
- ISODimensionNameType, *156, 157, 157*
- ISODistance, *156, 158, 226, 312*
- ISODistribution, *158, 159, 276*
- ISODistributionUnits, *160, 160*
- ISODistributor, *159, 160, 161, 179*
- ISODomainConsistency, *146, 162, 162*
- ISOElementSequence, *163*
- ISOEvaluationMethodType, *148, 164, 164*
- ISOExtendedElementInformation, *165, 165, 284*
- ISOExtent, *142, 144, 167, 168, 227, 326, 330, 331*
- ISOFeatureAssociation, *105, 168*
- ISOFeatureAttribute, *169, 169*
- ISOFeatureCatalogue, *94, 170, 176, 304*
- ISOFeatureCatalogueDescription, *173, 173, 219, 276, 277*
- ISOFeatureOperation, *174*
- ISOFeatureType, *87, 94, 105, 116, 119, 171, 175, 255, 256, 303*
- ISOFileName, *141, 177, 178*
- ISOFormat, *141, 159, 161, 178, 193, 194, 207*
- ISOFormatConsistency, *180, 180*
- ISOFreeText, *181*
- ISOGeographicBoundingBox, *182, 182*
- ISOGeographicDescription, *183*
- ISOGeographicExtent, *167, 168, 184, 329*
- ISOGeometricObjects, *184, 185, 350*
- ISOGeometricObjectType, *185, 185, 186*
- ISOGeorectified, *186, 187, 213*
- ISOGeoreferenceable, *187, 188, 214*
- ISOGriddedDataPositionalAccuracy, *189, 189*
- ISOGridSpatialRepresentation, *187, 190, 191, 213, 214*
- ISOHierarchyLevel, *192, 192, 276, 318*
- ISOIdentification, *192, 193, 276, 323*
- ISOIdentifier, *194, 195*
- ISOImageDescription, *195, 196*
- ISOImageryAbstractGeolocationInformation, *198, 198, 214*
- ISOImageryAcquisitionInformation, *198, 199, 218*
- ISOImageryAlgorithm, *200, 200, 239*
- ISOImageryBand, *201, 202*
- ISOImageryBandDefinition, *202, 203, 203*
- ISOImageryContext, *204, 204, 209*
- ISOImageryCoverageDescription, *205, 205*
- ISOImageryCoverageResult, *206, 207*
- ISOImageryEnvironmentalRecord, *199, 207, 208*
- ISOImageryEvent, *208, 209, 227, 228, 230, 231, 236*
- ISOImageryGCP, *210, 210, 211, 213*
- ISOImageryGCPCollection, *210, 211, 211*
- ISOImageryGeometryType, *212, 212, 232*
- ISOImageryGeorectified, *213, 213*
- ISOImageryGeoreferenceable, *214, 214*
- ISOImageryImageDescription, *215, 215*
- ISOImageryInstrument, *199, 209, 216, 227, 228, 234*
- ISOImageryMetadata, *218, 218*
- ISOImageryNominalResolution, *225, 226, 251*
- ISOImageryObjective, *199, 209, 227, 227, 230, 231*
- ISOImageryObjectiveType, *227, 229, 229*
- ISOImageryOperation, *199, 230, 230, 231, 232*
- ISOImageryOperationType, *230, 231, 231*
- ISOImageryPlan, *199, 230–232, 232, 247*
- ISOImageryPlatform, *199, 217, 230, 231, 234, 234*
- ISOImageryPlatformPass, *209, 227, 228, 236, 236*
- ISOImageryPolarisationOrientation, *202, 237, 237*
- ISOImageryPriority, *238, 238, 247*
- ISOImageryProcessing, *239, 239, 241*
- ISOImageryProcessStep, *241, 241*

- ISOImageryProcessStepReport, [241](#), [243](#), [244](#)
- ISOImageryRangeElementDescription, [205](#), [215](#), [244](#), [245](#)
- ISOImageryRequestedDate, [245](#), [246](#), [247](#)
- ISOImageryRequirement, [199](#), [246](#), [247](#)
- ISOImagerySensorType, [249](#), [249](#)
- ISOImagerySequence, [209](#), [250](#), [250](#)
- ISOImagerySource, [241](#), [251](#), [251](#)
- ISOImageryTransferFunctionType, [202](#), [252](#), [252](#)
- ISOImageryTrigger, [209](#), [253](#), [253](#)
- ISOImageryUsability, [254](#), [254](#)
- ISOImagingCondition, [195](#), [196](#), [254](#), [255](#)
- ISOInheritanceRelation, [176](#), [255](#), [256](#)
- ISOInitiative, [256](#), [256](#)
- ISOInitiativeType, [100](#), [257](#), [257](#)
- ISOKeywords, [142](#), [143](#), [193](#), [194](#), [258](#), [324](#), [330](#), [331](#)
- ISOKeywordType, [258](#), [259](#), [260](#)
- ISOLanguage, [142](#), [143](#), [171](#), [173](#), [260](#), [261](#), [266](#), [276](#)
- ISOLegalConstraints, [133](#), [143](#), [194](#), [261](#), [262](#), [324](#), [331](#)
- ISOLength, [156](#), [263](#)
- ISOLineage, [146](#), [263](#), [264](#)
- ISOListedValue, [169](#), [265](#), [265](#)
- ISOLocale, [266](#), [267](#)
- ISOLocaleContainer, [267](#), [267](#)
- ISOLocalisedCharacterString, [181](#), [200](#), [208](#), [217](#), [227](#), [230](#), [234](#), [239](#), [243](#), [268](#), [268](#)
- ISOLocalName, [94](#), [141](#), [175](#), [176](#), [269](#), [304](#)
- ISOMaintenanceFrequency, [269](#), [270](#)
- ISOMaintenanceInformation, [142](#), [193](#), [194](#), [270](#), [270](#), [276](#), [323](#), [331](#)
- ISOMeasure, [156](#), [271](#), [271](#)
- ISOMedium, [272](#), [272](#)
- ISOMediumFormat, [272](#), [273](#), [273](#)
- ISOMediumName, [272](#), [274](#), [274](#)
- ISOMemberName, [274](#), [275](#), [307](#)
- ISOMetadata, [9](#), [150](#), [275](#), [276](#), [295](#), [298](#), [303](#), [321](#), [322](#), [338](#)
- ISOMetadataExtensionInformation, [276](#), [283](#), [284](#)
- ISOMetadataNamespace, [285](#)
- ISOMetaIdentifier, [100](#), [101](#), [122](#), [148](#), [195](#), [209](#), [217](#), [227](#), [230](#), [234](#), [236](#), [239](#), [247](#), [251](#), [286](#), [286](#)
- ISOMimeType, [141](#), [286](#), [287](#)
- ISOMultiplicity, [94](#), [287](#), [287](#)
- ISOMultiplicityRange, [287](#), [288](#), [288](#)
- ISONonQuantitativeAttributeAccuracy, [289](#), [289](#)
- ISOobligation, [165](#), [166](#), [290](#), [290](#)
- ISOOnlineFunction, [291](#), [292](#)
- ISOOnlineResource, [134](#), [155](#), [284](#), [292](#), [292](#), [293](#), [294](#)
- ISOOperationMetadata, [293](#), [293](#), [294](#), [330](#), [332](#)
- ISOOtherAggregate, [294](#), [294](#)
- ISOParameter, [293](#), [294](#), [295](#), [296](#)
- ISOParameterDirection, [295](#), [296](#), [296](#), [297](#)
- ISOPixelOrientation, [186](#), [187](#), [297](#), [297](#)
- ISOPlatform, [298](#), [298](#)
- ISOPortrayalCatalogueReference, [299](#), [299](#)
- ISOPresentationForm, [122](#), [123](#), [300](#), [300](#)
- ISOProcessStep, [241](#), [264](#), [301](#), [301](#), [326](#)
- ISOProductionSeries, [302](#), [302](#)
- ISOPropertyType, [115](#), [116](#), [303](#), [303](#), [304](#)
- ISOQuantitativeAttributeAccuracy, [304](#), [304](#)
- ISOQuantitativeResult, [305](#), [306](#)
- ISORangeDimension, [139](#), [196](#), [306](#)
- ISORecord, [188](#), [244](#), [245](#), [306](#), [307](#)
- ISORecordType, [139](#), [196](#), [306](#), [308](#)
- ISOReferenceIdentifier, [309](#), [309](#), [310](#)
- ISOReferenceSystem, [210](#), [211](#), [276](#), [310](#), [310](#), [326](#)
- ISORepresentativeFraction, [311](#), [311](#), [312](#), [326](#)
- ISOResolution, [142](#), [143](#), [312](#), [312](#)
- ISOResponsibleParty, [122](#), [142](#), [161](#), [165](#), [166](#), [171](#), [193](#), [234](#), [247](#), [267](#), [276](#), [301](#), [302](#), [313](#), [323](#), [331](#), [349](#), [350](#)
- ISORestriction, [261](#), [262](#), [314](#), [314](#)
- ISORole, [313](#), [315](#), [315](#)
- ISORoleType, [105](#), [316](#), [316](#)
- ISOScale, [156](#), [317](#), [317](#)
- ISOScope, [146](#), [317](#), [318](#)
- ISOScopeDescription, [318](#), [318](#)
- ISOScopedName, [319](#), [319](#)
- ISOSecurityConstraints, [194](#), [320](#)
- ISOsensor, [321](#), [321](#)
- ISOseries, [322](#), [322](#)

- ISOServiceIdentification, [323](#), [330](#)
- ISOSource, [241](#), [264](#), [302](#), [325](#), [326](#)
- ISOSpatialRepresentation, [207](#), [276](#), [327](#), [327](#)
- ISOSpatialRepresentationType, [142](#), [143](#), [206](#), [207](#), [328](#), [328](#)
- ISOSpatialTemporalExtent, [329](#), [329](#)
- ISOSRVServiceIdentification, [330](#), [331](#)
- ISOStandardOrderProcess, [330](#), [331](#), [335](#), [336](#)
- ISOStatus, [142](#), [193](#), [230](#), [232](#), [323](#), [331](#), [337](#), [337](#)
- ISOStereoMate, [337](#)
- ISOTelephone, [134](#), [338](#), [339](#)
- ISOTemporalConsistency, [339](#), [339](#)
- ISOTemporalExtent, [167](#), [168](#), [329](#), [340](#), [341](#)
- ISOTemporalValidity, [341](#), [341](#)
- ISOThematicClassificationCorrectness, [342](#), [343](#)
- ISOTimePeriod, [343](#)
- ISOTopicCategory, [142](#), [143](#), [344](#), [344](#)
- ISOTopologicalConsistency, [345](#), [345](#)
- ISOTopologyLevel, [346](#), [346](#), [350](#)
- ISOTypeName, [117](#), [275](#), [295](#), [296](#), [347](#), [347](#)
- ISOUnlimitedInteger, [348](#), [348](#)
- ISOURL, [292](#), [348](#), [349](#)
- ISOUsage, [193](#), [349](#), [350](#)
- ISOVectorSpatialRepresentation, [350](#), [350](#)
- ISOVerticalExtent, [167](#), [168](#), [351](#), [351](#)
- list, [126](#)
- logical, [105](#), [131](#), [173](#), [175](#), [186](#), [188](#), [191](#), [196](#), [255](#), [295](#)
- matrix, [48](#)
- numeric, [79](#), [108](#), [182](#), [195](#), [202](#), [208](#), [272](#), [348](#), [351](#)
- pivot_format, [352](#)
- POSIXt, [79](#), [80](#), [122](#), [148](#), [209](#), [246](#), [247](#), [276](#), [301](#), [336](#), [349](#), [350](#)
- R6Class, [11](#), [17–91](#), [93–100](#), [102–107](#), [109–122](#), [124–128](#), [130–142](#), [146](#), [148–158](#), [160](#), [162–165](#), [167–170](#), [173–175](#), [178](#), [180–190](#), [192–195](#), [198–201](#), [203–215](#), [217](#), [218](#), [226](#), [227](#), [229–232](#), [234](#), [236–239](#), [241](#), [243–246](#), [249–258](#), [260](#), [261](#), [263](#), [265–275](#), [284–323](#), [325–330](#), [336–342](#), [344–352](#)
- readISO19139, [9](#), [353](#)
- registerISOCodelist, [354](#)
- registerISOMetadataNamespace, [354](#)
- registerISOMetadataSchema, [355](#)
- registerMappingFormat, [356](#)
- registerMappings, [356](#)
- setGeometaOption, [357](#)
- setISOCodelists, [357](#)
- setISOMetadataNamespaces, [358](#)
- setISOMetadataSchemas, [358](#)
- setMappingFormats, [358](#)