Identifiers

I need to uniquely identify datasets as well as platforms, instruments, software, and other associated resources.

Overview

The need for identifiers in metadata records was first recognized in the DIF Standard and FGDC Remote Sensing Extensions. These standards introduced identifiers for the metadata records. In ISO 19115 this role is addressed by the fileIdentifier, a character string included in the MD_ or MI_Metadata object. This character string has been replaced with an MD_Identifier in 19115-1.

Including fileIdentifiers in the ISO metadata records gives metadata creators a mechanism for uniquely identifying them. This is becoming more important as metadata records evolve from single files into collections of related objects that can be harvested into repositories like geo.data.gov along multiple paths. There is no reliable way to identify duplicate records without a unique identifier in the actual record.

If the metadata records belongs to a parent metadata collection the parentIdentifier field can be used to reference the parent collection.

Identifiers are also used to reference resources associated with the data set or service described by the metadata. For example platforms, instruments, software, documentation, reports, partners and products can all be unambiguously referenced and described with the MD_Identifier object.

Digital Object Identifiers (DOIs) and Other Dataset Identifiers

Digital Object Identifiers are most commonly used to identify and cite published datasets. In the ISO standard these identifiers should be included as an MD_Identifier in the CI_Citation for the dataset. If the metadata record itself also had a DOI, that would be in the fileIdentifier.

As DOIs become more ubiquitous, the prefix doi: is becoming a standard internet protocol. This means that browsers and other tools will know that the string doi:10.5067/MEASURES/DMSP-F8/SSMI/DATA302 means the same thing as the URL: http://dx.doi.org/10.5067/MEASURES/DMSP-F8/SSMI/DATA302. As this becomes more common, it addresses the problem of identifiers with no straightforward mechanism for resolution.

Structure

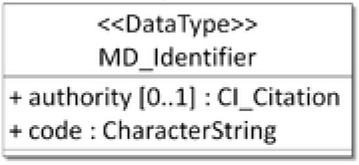
The MD_Identifier object in ISO 19115 includes two elements: a code and an authority. The code is an alphanumeric value identifying an object in a namespace that is maintained by the authority. In this case the CI_Citation cites the authority for the code. In many ways this is similar to the namespace for the code. It is the authority which understands and can explain or resolve the code.

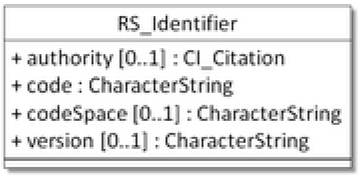
However, there is no agreed upon approach for how the namespace is described in the CI_Citation. ISO 19115-1 addresses this limitation by adding a codeSpace field to the MD_Identifier object. The codeSpace field unambiguously defines the namespace for the identifier. ISO 19115-1 also includes description and version fields in the MD_Identifier object. The description field enables a brief description of the code to be documented, and the version field enables the the Identifier version to be documented.

The RS_Identifier extends the MD_Identifier in ISO 19115 by adding a codeSpace and a version for the namespace. These additions address the lack of an agreed upon approach for describing a namespace using the authority/gco:CI_Citation alone. However, the standard only supports RS_Identifiers in the referenceSystemInfo class.

Many NASA identifiers include short and long names. In the translation to ISO the short name becomes the code and the long name becomes the description.

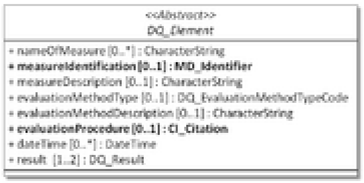
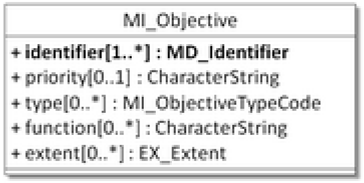
Note: In ISO 19115-1 the RS_Identifier object is replaced with an MD_Identifier object which includes codeSpace, version and description fields.




RS_Identifier
+ authority [0..1] : CI_Citation
+ code : CharacterString
+ codeSpace [0..1] : CharacterString
+ version [0..1] : CharacterString

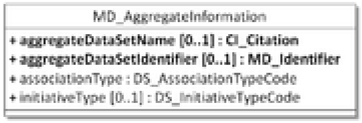
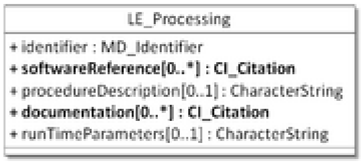
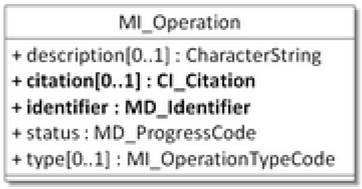
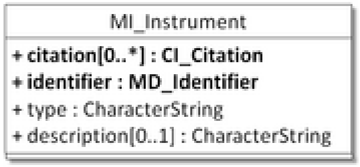
Usage

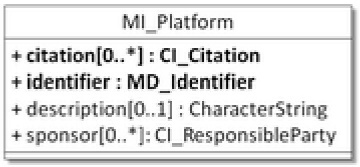
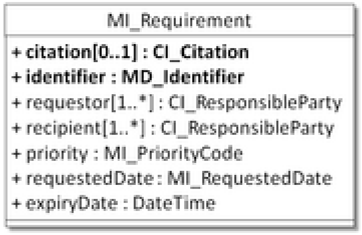
Identifiers occur in many places in the ISO standard. The identifiers in Citations are particularly important because Citations also occur in many locations throughout the standard.

Usage	Description and Xpath
Quality Measure Identifier  <<Abstract>> DQ_Element + nameOfMeasure [0..*] : CharacterString + measureIdentification [0..1] : MD_Identifier + measureDescription [0..1] : CharacterString + evaluationMethodType [0..1] : DQ_EvaluationMethodTypeCode + evaluationMethodDescription [0..1] : CharacterString + evaluationProcedure [0..1] : CI_Citation + dateTime [0..*] : DateTime + result [1..2] : DQ_Result	Provides identification information for a data quality measure, such as 'Percent Missing Data'. /gmi:MI_Metadata/gmd:dataQualityInfo/gmd:MD_DataQuality/gmd:report/gmd:DQ_Element /gmd:measureIdentification
Objective Identifier  MI_Objective + identifier[1..*] : MD_Identifier + priority[0..1] : CharacterString + type[0..*] : MI_ObjectiveTypeCode + function[0..*] : CharacterString + extent[0..*] : EX_Extent	Provides identification information for the operation objective. /gmi:MI_Metadata/gmd:acquisitionInformation /gmi:MI_AcquisitionInformation/gmi:objective/gmi:MI_Objective/gmi:citation

(CI_Citation + MD_Identifier)++

There are many cases in the ISO Standard where CI_Citations and MD_Identifiers are used together to reference and identify external resources. We term these (CI_Citation+MD_Identifier)++:

Usage	Xpath and Description
Aggregate Citation and Identifier  MD_AggregateInformation + aggregateDataSetName [0..1] : CI_Citation + aggregateDataSetIdentifier [0..1] : MD_Identifier + associationType : DS_AssociationTypeCode + initiativeType [0..1] : DS_InitiativeTypeCode	(CI_Citation + MD_Identifier) + associationType + initiativeType = MD_AggregationInformation /gmi:MI_Metadata/gmd:identificationInfo/gmd:MD_DataIdentification | srv:ServiceIdentification/gmd:aggregationInfo/gmd:MD_AggregateInformation/gmd:aggregateDataSetName
Software Citation and Identifier  LE_Processing + identifier : MD_Identifier + softwareReference[0..*] : CI_Citation + procedureDescription[0..1] : CharacterString + documentation[0..*] : CI_Citation + runTimeParameters[0..1] : CharacterString	(CI_Citation + MD_Identifier) + description + scaleDenominator + sourceReferenceSystem + sourceExtent + processedLevel + resolution + sourceStep = LE_Source /gmi:MI_Metadata/gmd:dataQualityInfo/gmd:MD_DataQuality /gmd:lineage/gmd:LE_Lineage/gmd:processStep /gmd:LE_ProcessStep/gmd:processingInformation /gmd:LE_Processing /gmd:softwareReference
Operation Citation and Identifier  MI_Operation + description[0..1] : CharacterString + citation[0..1] : CI_Citation + identifier : MD_Identifier + status : MD_ProgressCode + type[0..1] : MI_OperationTypeCode	(CI_Citation + MD_Identifier) + description + status + type = MI_Operation /gmi:MI_Metadata/gmd:acquisitionInformation /gmi:MI_AcquisitionInformation/gmi:operation /gmi:MI_Operation/gmi:citation
Instrument Citation and Identifier  MI_Instrument + citation[0..*] : CI_Citation + identifier : MD_Identifier + type : CharacterString + description[0..1] : CharacterString	(CI_Citation + MD_Identifier) + description + type = MI_Instrument /gmi:MI_Metadata/gmd:acquisitionInformation /gmi:MI_AcquisitionInformation /gmi:instrument/gmi:MI_Instrument/gmi:citation

Platform Citation and Identifier  MI_Platform + citation[0..*] : CI_Citation + identifier : MD_Identifier + description[0..1] : CharacterString + sponsor[0..*]: CI_ResponsibleParty	(CI_Citation + MD_Identifier) + description + sponsor = MI_Platform /gmi:MI_Metadata/gmd:acquisitionInformation /gmi:MI_AcquisitionInformation /gmi:platform/gmi: MI_Platform/gmi:citation
Requirement Citation and Identifier  MI_Requirement + citation[0..1] : CI_Citation + identifier : MD_Identifier + requestor[1..*] : CI_ResponsibleParty + recipient[1..*] : CI_ResponsibleParty + priority : MI_PriorityCode + requestedDate : MI_RequestedDate + expiryDate : DateTime	(CI_Citation + MD_Identifier) + requestor + recipient + priority = MI_Requirement /gmi:MI_Metadata/gmd:acquisitionInformation /gmi:MI_AcquisitionInformation /gmi:requirement/gmi: MI_Requirement/gmi:citation

fileIdentifiers and parentIdentifiers

One of the ironic aspects of the ISO 19115 is that the identifiers for metadata records (/gmi:MI_Metadata/gmd:fileIdentifier/gmd:characterString and /gmi:MI_Metadata/gmd:parentIdentifier/gmd:characterString) are characterStrings rather than MI_Identifiers. In order to help ensure uniqueness these strings should include a namespace and a code guaranteed to be unique in that namespace. For example:

```
<gmd:fileIdentifier>
<gco:CharacterString>gov.noaa.class:AERO100</gco:CharacterString>
</gmd:fileIdentifier>.
```

In this case, gov.noaa.class is a namespace, and AERO100 is a code guaranteed to be unique in that namespace. In this case, the code is meaningful to the data provider. Creating meaningful identifiers that are unique over a large collection can many times be difficult. It might make sense to consider using UUIDs for file names and identifiers, although this takes some getting used to.

ISO 19115-1 overcomes this challenge by changing the type of fileIdentifiers and parentIdentifiers to MD_Identifiers, see Structure section above.

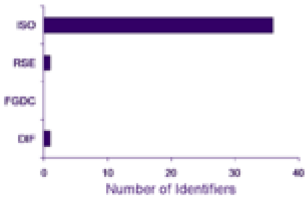
Should the fileIdentifier match the file name? There is no rule in ISO that specifies a relationship between the fileIdentifier and the file name. It is, however, very convenient to have the file name available from within the record, particularly for supporting access to the file name when transforming the XML into HTML.

Managing ISO Metadata in a Database

Including identifiers in ISO metadata records gives metadata creators a mechanism for uniquely identifying metadata records and pieces of those records for the first time. The importance of unique identifiers is well known to people that use relational database management systems, they are the primary keys that identify items and make relationships possible. This is also becoming more important as metadata records are harvested into repositories like Geospatial One-Stop along multiple paths. There is no reliable way to identify duplicate records without an identifier in the actual record.

The REST approach to web services is becoming more and more commonplace. The first principle of REST is “Give Everything an ID”. Given this principle, one measure of how “RESTful” a metadata standard might be is the number of id’s that are included in the standard. This compares the number of IDs included in four metadata standards. The original FGDC standard has none, the Directory Interchange Format (DIF) and the FGDC Remote Sensing Extensions (RSE) have one and the ISO 19115(-2) has 36.

These ids make it possible to reference many individual elements in an ISO metadata record directly, so the ISO standard is very compatible with a RESTful approach.



Identifiers in CMR Metadata

- Identifiers