Science & Technology Facilities Council

Technical Report
RAL-TR-2013-001

Metadata Objects for Linking Environmental Sciences (MOLES) Version 3.4 Users Guide

S Ventouras, BN Lawrence

March 2013

RAL Library
STFC Rutherford Appleton Laboratory
R61
Harwell Oxford
Didcot
OX11 0QX

Tel: +44(0)1235 445384
Fax: +44(0)1235 446403
email: libraryral@stfc.ac.uk

ISSN 1358- 6254

Metadata Objects for Linking Environmental Sciences

(MOLES)

Version 3.4

Users Guide

S. Ventouras & B.N. Lawrence

Centre for Environmental Data Archival
SCIENCE AND TECHNOLOGY FACILITIES COUNCIL
NATURAL ENVIRONMENT RESEARCH COUNCIL

British Atmospheric Data Centre
NATIONAL CENTRE FOR ATMOSPHERIC SCIENCE
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Earth Observation
NATURAL ENVIRONMENT RESEARCH COUNCIL

Table of Contents

1. Introduction and Motivation

A concluded inference, a conducted study or a decision made regarding a portion of the real world - as defined within an application domain, e.g. study of cloud properties - should be based on already existing data.

A portion of the real world can be considered a set of real world objects - which can be either physical or abstract - together with their properties and relationships relevant to the application domain. For example, in an atmospheric science application, a RADAR measuring echoes from clouds is a physical object whereas the object representing the measuring campaign is abstract. Furthermore, for this application, the manufacturer of the RADAR could be an appropriate property and the RADAR could be associated with the measuring campaign. For this campaign, the RADAR would have the role 'used instrument'.

A state of a portion of the real world is realised by values of properties and relationships of the objects making this portion of real world. Therefore, data (i.e. values of properties and relationships) can be defined as a set of isolated and unrelated raw facts with an implicit meaning, and can be anything such as name of a location, a number, images, sound, etc.

However, data by itself is not enough. It has to be processed and converted into a meaningful and useful form, known as information [*Khurana, 2010*]. This allows it to be fully understood and thus exploitable within an application. For example, 'air temperature', '25', 'London', '23rd of July 2010' etc., is the data whereas "the air temperature in London on 23rd of July 2010 was 25 degrees Celsius" is the relevant piece of information.

In other words, the data of a prime interest (i.e. the values of the RADAR echoes in the previous example) within an application needs supplementary data in order to complete a view of a portion of the real world. That means that the process of revealing the semantics behind the data of a prime interest and converting it to information requires some additional data known as 'data about data' or metadata [*ISO 19115:2005*].

It has to be emphasised here that the required supplementary data does not involve only simple and obvious descriptions of the values of a prime interest. Metadata depends on the application in which the data of primary interest is used, and thus may include values of properties and relationships of other objects of the relevant portion of the real world. Therefore, there is not an unambiguous distinction between data and metadata of an application. This is addressed clearly together with a detailed taxonomy of data of prime interest and its supplementary data by [*Lawrence et al, 2009*].

MOLES is an information model for describing metadata covering a broad range of applications within multiple disciplines. These are mainly, but not limited, those within the earth and physical sciences. MOLES is primarily of use to consumers of data, especially in an interdisciplinary context. It allows them to establish details of provenance, and to compare and contrast such information without recourse to discipline-specific metadata or private communications with the original investigators. MOLES can also be of use to the custodians of data, providing an organising paradigm for the data and metadata (which is how it is being deployed in the Centre for Environmental Data Archival, CEDA).

The model was originally developed within the Natural Environment Research Council (NERC) Data Grid project [NDG] to fill a missing part of the 'metadata spectrum' [*Lawrence* *et al, 2009*]. Its aim was to facilitate the use of data, which is increasingly being shared, exchanged and frequently used for purposes other than those for which it was originally intended.

This document describes MOLES3.4 which is a major revision of MOLES compared to the previous attempts of MOLES [*Lawrence et al, 2009; Latham et al., 2009*]. Whereas the user requirements remain the same, MOLES3.4 is built entirely on ISO 19100 series geographic information standards, particularly the ISO 19156 Observations and Measurements (O&M) standard.

This standardisation facilitates interoperability with other information systems and provides freedom to mix and match information system components without compromising overall success [*ISO 19101:2002*]. It makes MOLES3.4 able to achieve a common and correct understanding of the content and structure of data within a particular application field. It may also provide a computer readable schema for applying automated mechanisms for data management. In addition, the foundation of MOLES3.4 on other schemas (e.g. Observations and Measurements) helps to avoid the creation of a large and complex system.

Initial designs of MOLES3.4 were developed over the last few years [*Ventouras et al, 2009; 2010; 2011*], and MOLES3.4 has now been finalised to a generic and flexible model allowing many different situations to be modelled. If required, MOLES3.4 classes can be easily adjusted, following the guidelines of ISO 19106 and ISO 19109, to fit specific domain applications.

Section 2 of this document describes the foundations of the MOLES3.4 schema. It outlines the ISO 19156 Observations and Measurements standard from the ISO 19100 series which plays the role of the base standard in the development of this version of MOLES. A comprehensive description of the model is given in section 3. The model is organised into 10 leaf packages which facilitate maintenance and future improvement of the current version. This package organisation allows MOLES users to select specific modules, if they wish, instead of holding the whole schema. For the attributes taking predefined values, MOLES3.4 does not enforce the controlled vocabulary that should be used. Controlled vocabularies are selected according to the user requirements and are managed externally to MOLES3.4. Annex A includes existing code lists that could be used to provide predefined values. Annex B gives an introduction to the concept of “feature” which is the fundamental element in the development of this schema and the relevant UML notation. Annex B is recommended to readers who are not familiar with information modelling and particular with ISO 19100 series. It would help if it is read before reading the description of the model or it can be used as a quick and easy reference. For convenience, Annex C provides a brief visual description of types obtained from the ISO 19100 series standards which are being integrated with MOLES3.4. Finally, Annex D holds a discussion regarding MOLES3.4 and the storage of the data.

2. Model Foundations

2.1 Overview

A typical sequence of data capturing involves one or more projects/activities under which a number of actions are undertaken, using appropriate tools and methods to produce the datasets. MOLES is not particularly concerned with the details of the datasets structures but is aimed at capturing the important provenance elements associated with them. Key components of MOLES include:

- Project descriptions;
- the action itself and the processes used to acquire or generate the data;
- the collection of data into groups according to user requirements.

The general structure of the model gives a central place to the concept of “observation”. According to [*Fowler,1998*] an observation is an act that results in the estimation of the value of a feature property using a designated procedure, such as a sensor, instrument, algorithm or process chain. An observation is associated with a discrete time instant or period through which a number, term or other symbol is assigned to a phenomenon. The result of an observation is an estimate of the value of a property of some feature, so the details of the observation are metadata concerning the value of the feature property.

2.2 ISO 19100 series Standards

The conceptual schema specified in this document is rooted in the ISO 19100 series of standards from the ISO/TC 211 “Harmonised Model”. In particular, it has been created:

a) following the guidance provided by ISO/TC 211 (i.e. ISO 19101, 19106 and 19109) and has been formalised in the Unified Modelling Language (UML) ISO/IEC 19501, following the guidance of ISO/TS 19103;

b) by integrating (i.e., either exploiting existing concepts and relationships or specialising the information in these standards) reusable modules of conceptual schemas defined within ISO 19100 series e.g. temporal schema, metadata, etc.

A fundamental unit in the development of an application schema is the concept of feature which represents an abstraction of real-world phenomena. An overview of this concept as defined within ISO 19109 is given in Annex B.

2.2.1 ISO 19156 Observations and Measurements

ISO 19156 (Observations & Measurements) defines a conceptual schema, designed to apply across disciplines, for observations and for features involved in sampling when making observations. It serves as the base standard from ISO 19100 series in the development of MOLES3.4 schema.

Core class of Observations & Measurements model is the *OM_Observation* representing the action of the observation. Concepts defined and modelled within ISO 19156 which are directly associated with the concept of observation and used in MOLES3.4 schema are (see Figure 1):

Feature of Interest: a real-world object, i.e. a feature, carrying the properties which are under observation. It has the role *featureOfInterest* with respect to the observation. The feature of interest is the subject of the observation.

Observed Property: a phenomenon associated with the *featureOfInterest* for which the observation result provides an estimate of its value. The phenomenon has the role *observedProperty* with respect to the observation.

The observedProperty shall be a phenomenon associated with the *featureOfInterest*.

Observation Result: value of the *observedProperty* obtained by the action of observation. The value has the role *result* with respect to the observation. The *result* is of type “Any”, since it may represent the value of any feature property.

The result type shall be consistent with the observed property.

Procedure: a process used to generate the result. The process has the role *procedure* with respect to the observation and is represented by the class *OM_Process* which is a feature type, is abstract and has no attributes, operations or associations. It serves as the base class for observation processes. A process might be responsible for more than one observations. The procedure shall be suitable for the observed property.

EXAMPLE Measuring (*the act of the observation*) the rain rate at Chilbolton, UK at 16:00 GMT the *featureOfInterest* is the rain, the *observedProperty* is the rain rate, the *procedure* is a tipping bucket rain-gauge and the *result* is 40mm/hr.

Descriptions of the *OM_Observation* attributes are given in the following sections as adapted in the context of MOLES3.4

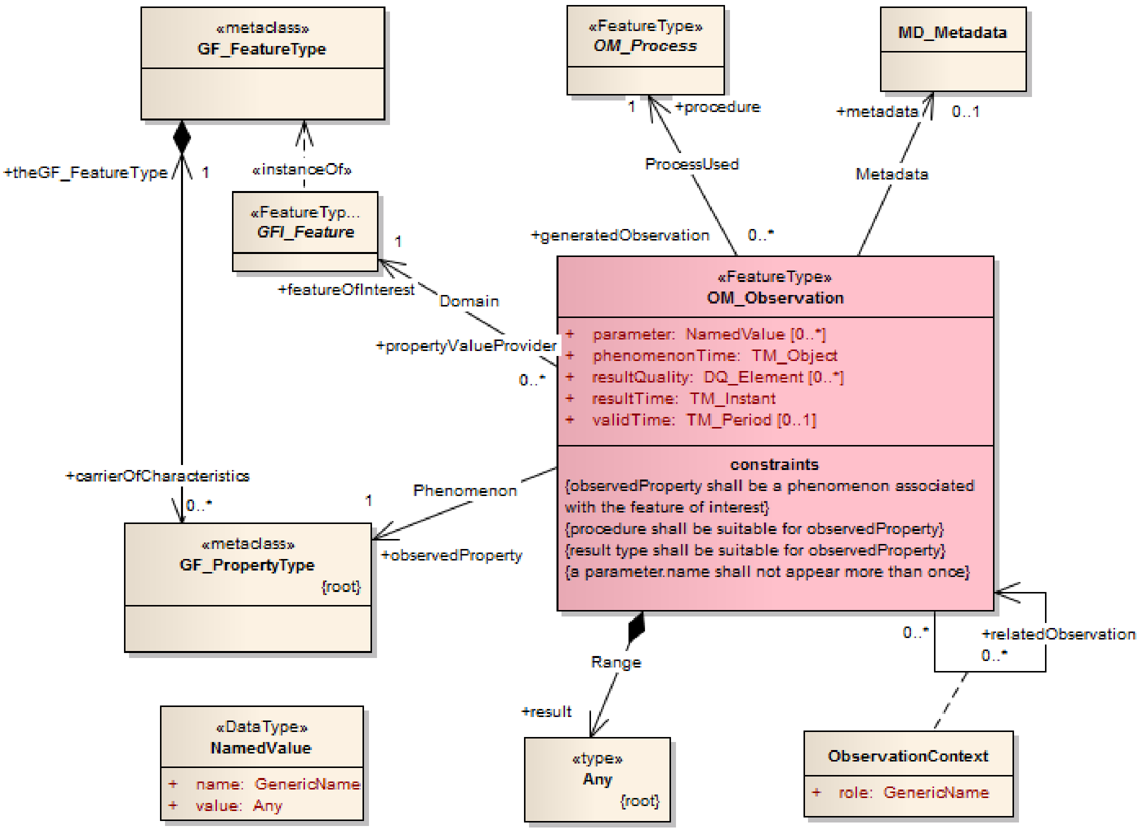

Figure 1 - The basic Observations & Measurements Model.

3. MOLES Application Schema

The schema is organised into 10 leaf packages: Observation, Feature Of Interest, Observable Property, Observation Process, Acquisition, Computation, Result, Project, Observation Collection and Utilities, as shown in Figure 2. The dependencies *"import"* in Figure 2 indicate the packages that MOLES3.4 application schema consists of.

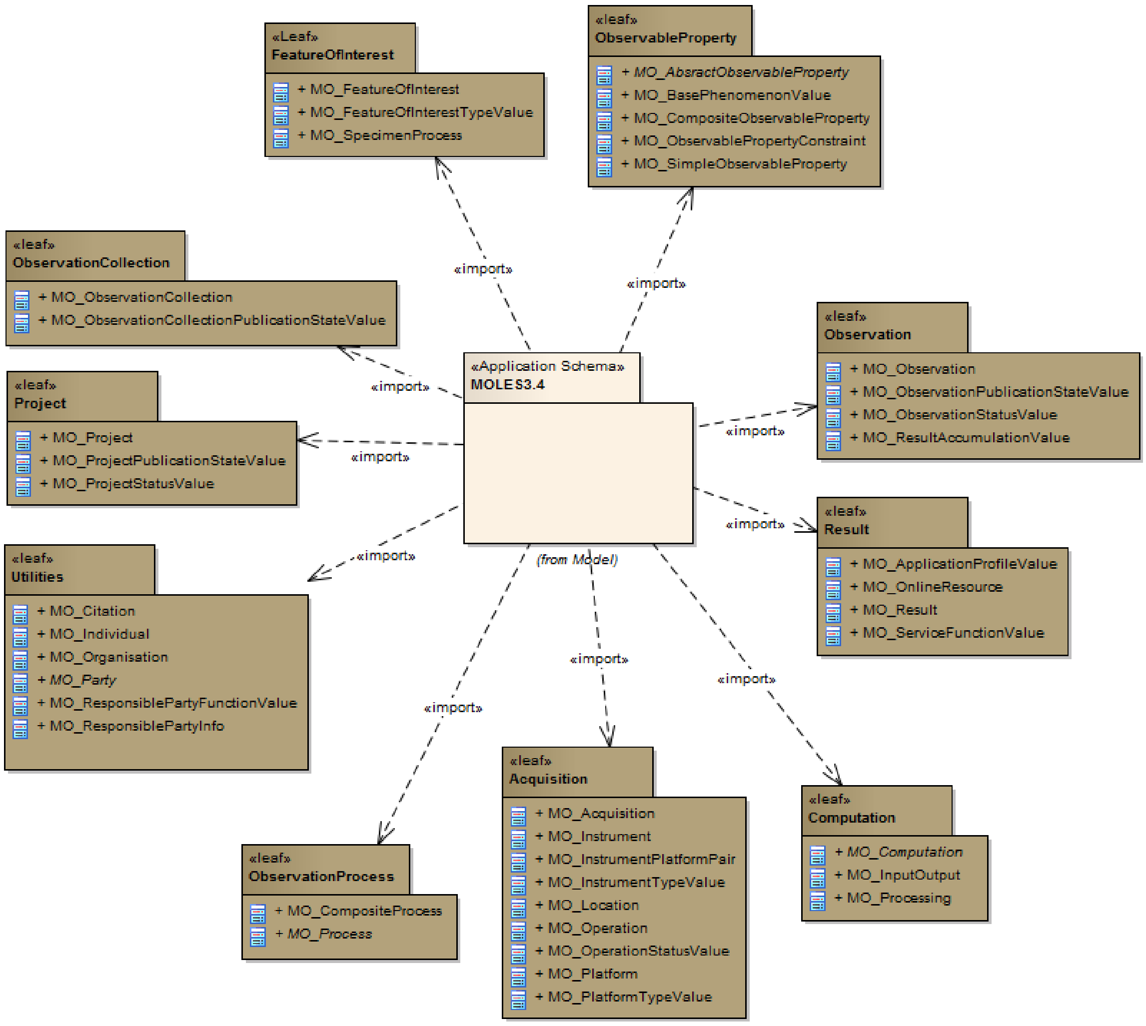

Figure 2 - Leaf packages of the MOLES3.4 schema.

All the classes defined within MOLES3.4 schema have the prefix MO_. For the description of class attributes and role names we follow the notation:

<<stereotype>> *name : type* [multiplicity] {schema where the type is defined}

This indicates, besides the name, type, multiplicity and stereotype of the attribute or role name, the schema where the type is defined. If no explicit multiplicity exists, the multiplicity is assumed to be 1.

NOTE Only one stereotype, <<voidable>>, is used within MOLES3.4 for attributes and roles names (see section B.5).

3.1 Observation Package

This package adapts the action of the observation within the context of MOLES by defining the class *MO_Observation* as a specialisation of ISO 19156 class *OM_Observation*.

Figure 3 shows the ISO standards that the MOLES3.4 Observation leaf package is integrated with. In particular, the dependencies *"import"* mean that the Observation leaf package imports (i.e. uses) structures and definitions from the schemas of ISO 19115, ISO 19103 and the observation schema of ISO 19156.

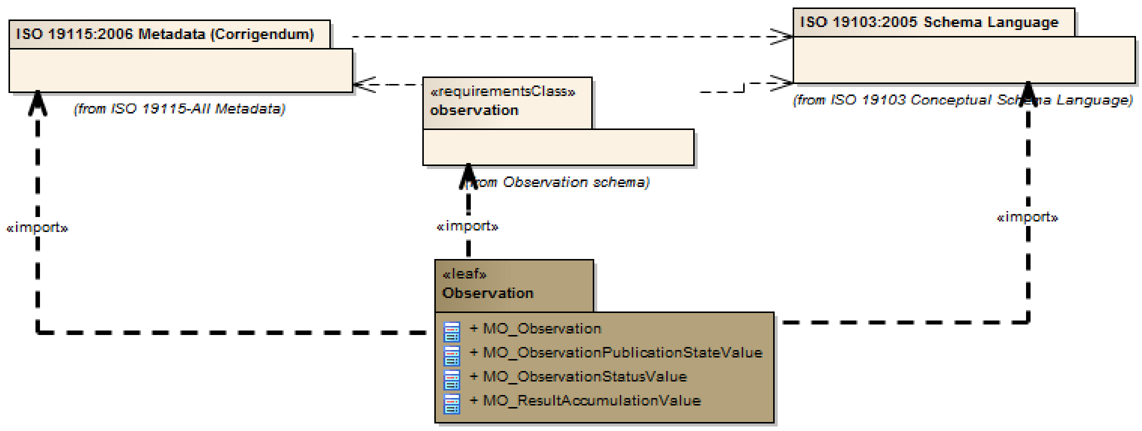

Figure 3 – Integration of MOLES3.4 Observation package with other ISO schemas.

3.1.1 MO_Observation Class

This class introduces eleven new attributes and one new association to the modelling of the act of the observation (see Figure 4). The descriptions of these properties, together with the descriptions of the inherited properties as they perceived in the MOLES domain, are given in this section.

3.1.1.1 Inherited Attributes

3.1.1.1.1 **phenomenonTime**

The attribute *phenomenonTime:TM_Object* {ISO 19108:2006 Temporal Schema} describes the extent of the result's temporal domain.

The type of the attribute, *TM Object*, suggests that the temporal extent of the result can be an instant or a time interval.

3.1.1.1.2 **resultTime**

As stated in ISO 19156, the attribute *resultTime:TM_Instant* {ISO 19108:2006 Temporal Schema} describes the time instant when the result became available, typically when the procedure associated with the observation was completed. For some observations, this is identical to the phenomenonTime. However, there are important cases where they differ. Simulations, for example, can estimate the values for phenomena in the future or past. The phenomenonTime is the time that the result applies to, while the resultTime is the time that the simulation was executed.

3.1.1.1.3 **validTime**

The attribute *validTime:TM_Period* [0..1] {ISO 19108:2006 Temporal Schema} describes the time period during which the result is intended to be used. For simulations, for example, this attribute describes the time window that the result is applicable.

3.1.1.1.4 **resultQuality**

The attribute *resultQuality:DQ_Element* [0..*] {ISO 19157:2012 Data Quality} describes the quality of the result. The quality of a result maybe assessed following the procedures in ISO 19157.

3.1.1.1.5 **parameter**

The attribute *parameter:NamedValue* [0..*] {ISO 19156:2011} describes an arbitrary event-specific parameter. This might be an environmental parameter, an instrument setting or input, or an event-specific sampling parameter that is not tightly bound to either the *featureOfInterest* or to the observation *procedure*. To avoid ambiguity, there shall be no more than one parameter with the same name.

The class *NamedValue* (Figure 4) defines the type of this attribute and supports two attributes:

- The attribute *name:GenericName* {ISO 19103:2005 Schema Language} shall indicate the meaning of the named value. Its value should be taken from a well-governed source if possible.

- The attribute *value:Any* {ISO 19103:2005 Schema Language} shall provide the value. The type "Any" should be substituted by a suitable concrete type, such as CI_ResponsibleParty or Measure.

EXAMPLE There are circumstances where an instance of *MO_Observation* utilises an instance of *MO_Processing* (see section 3.6.1) with the attribute *MO_Processing.processingInput* present. The attribute *processingInput* provides only a description of the input data whereas the actual data values, as they are specific to the event of the observation, if required, shall be stored in *MO_Observation* attribute *parameter,* i.e.

MO_Observation.parameter>NamedValue.value

with

MO_Observation.parameter>NamedValue.name =
MO_Processing.processingInput>MO_InputOutput.name

3.1.1.2 MOLES defined Attributes

3.1.1.2.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the observation including its intentions.

3.1.1.2.2 **publicationState**

The attribute *publicationState:MO_ObservationPublicationStateValue* [0..1] {MOLES3.4} indicates the state of the observation metadata record.

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ObservationPublicationStateValue* is empty serving as a super-class for code lists which can specify the attribute values, for example, the *CEDA_PublicationStateValue* code list (see section A.1).

3.1.1.2.3 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the observation.

The description of the class *MO_Citation* which defines the type of this attribute is given in section 3.10.1.

3.1.1.2.4 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO19115:2006 Metadata (Corrigendum)} provides an external identifier of the observation. The identifier may be used by an external application to reference an observation. For example, an identifier can be used for packaging an observation to an observation collection (see section 3.9).

3.1.1.2.5 **permission**

The attribute <<voidable>> *permission:MD_Constraints* [0..1] {ISO 19115:2006 Metadata (Corrigendum)} describes the restrictions on the access and use of observation's result. Restrictions on the access and use of the observation result are known without navigating from the observation to the result.

3.1.1.2.6 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding individuals or organisations related to the observation.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.1.1.2.7 **resolution**

The attribute <<voidable>> *resolution:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a narrative description of the result's domain density (temporal, spatial and/or spectral resolution).

3.1.1.2.8 **resultAccumulation**

The attribute <<voidable>> *resultFrequency:MO_ResultAccumulationValue* [0..1] {MOLES3.4} indicates the frequency with which additions are/were made to the observation's result.

EXAMPLE Measurements of rain rate at Chilbolton Observatoty,UK are being conducted since 2000. This is an *onGoing* observation with a result being growing on a daily basis.

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ResultAccumulationValue* is empty serving as a super-class for code lists which can specify the attribute values. A suggested code list is the ISO 19115:2006 Metadata (Corrigendum) code list *MD_MaintenanceFrequencyCode* (see section A.2)

3.1.1.2.9 **status**

The attribute <<voidable>> *status:MO_ObservationStatusValue* {MOLES3.4} indicates the status of the act of the observation.

> **EXAMPLE** Measurements of rain rate at Chilbolton Observatoty, UK are being conducted since 2000. The status of this observation is onGoing.

NOTE If an observation is *onGoing* then the relative Project shall be *onGoing* (see section 3.1.1.5.1).

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ObservationStatusValue* is empty serving as a super-class for code lists which can specify the attribute values. A suggested code list is the ISO 19115:2006 Metadata (Corrigendum) code list *MD_ProgressCode* (see section A.3)

3.1.1.2.10 **geographicExtent**

The attribute <<voidable>> *geographicExtent:EX_GeographicExtent* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} describes the geographic area, e.g. bounding polygon, within which the observation's result is available. *geographicExtent* is the envelope of the geographic areas of multiple coverages.

3.1.1.2.11 **verticalExtent**

The attribute <<voidable>> *verticalExtent:EX_VerticalExtent* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} describes the vertical extent of the observation's result. *verticalExtent* is the envelope of the vertical extents of multiple coverages.

3.1.1.3 Inherited Associations

In addition to the properties *featureOfInterest*, *observedProperty*, *procedure*, *result* which are inherited from the *OM_Observation* class, and their types are defined in sections 3.2, 3.3, 3.4 and 3.7 respectively, the class *MO_Observation* inherits also the properties *relatedObservation* and *metadata*.

3.1.1.3.1 **relatedObservation**

Some observations depend on other observations to provide a context which is important, sometimes essential, in understanding the result. These dependencies are stronger than mere spatiotemporal coincidences, requiring explicit representation. If present, the association class *ObservationContext* (see Figure 1) shall link an *OM_Observation* to another *OM_Observation*, with the role name *relatedObservation:OM_Observation* [0..*] {ISO 19156:2011} for the target. It shall support one attribute, *role:GenericName* which describes the relationship of the target *OM_Observation* to the source *OM_Observation*.

> **EXAMPLE** The related observation may provide input to a process that generates a new result.

3.1.1.4 metadata

The property *metadata:MD_Metadata* [0..1] {ISO 19115:2006 Metadata(Corrigendum)} provides descriptive metadata to the act of the observation.

3.1.1.5 MOLES defined Association

3.1.1.5.1 inSupportOf

The property <<voidable>> *inSupportOf:MO_Project* {MOLES3.4} describes a project requiring this observation to accomplish its objectives – it is the project which originates or manages the observation. The type of *inSupportOf* is defined in section 3.8**.**

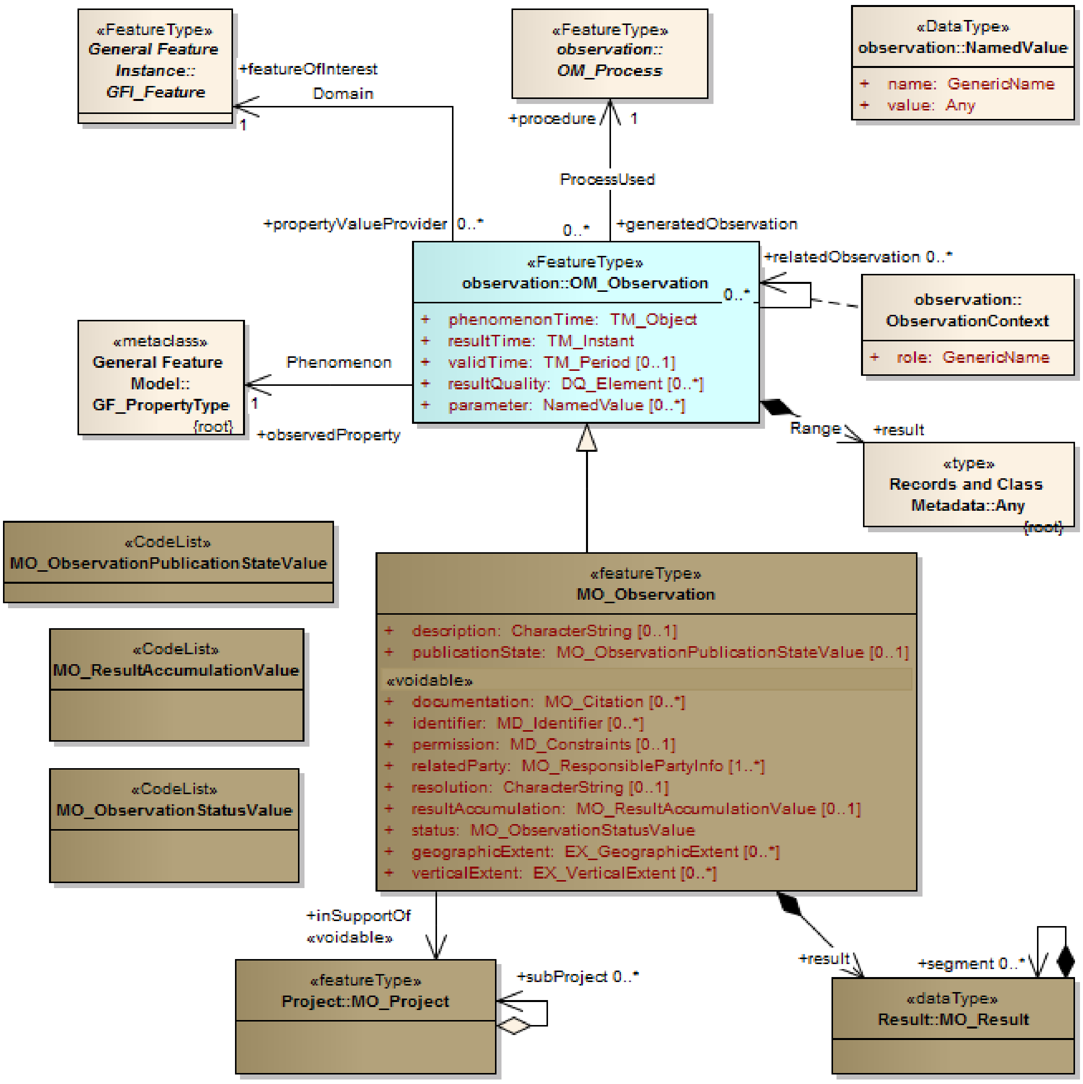

Figure 4 - Properties (Attributes and associations) and their types and stereotypes of *MO_Observation* class.

3.2 Feature Of Interest Package

This package provides the modelling of the subject of the observation by using the ISO 19156 class *SF_Specimen* and defining the class *MO_FeatureOfInterest*. It defines also the class *MO_SpecimenProcess* (a specialisation of ISO 19156 class *SF_Process)* for the modelling of processes associated with the design and preparation of specimens. The dependencies of Feature Of Interest Package to other schemas from the ISO 19100 series are shown in Figure 5.

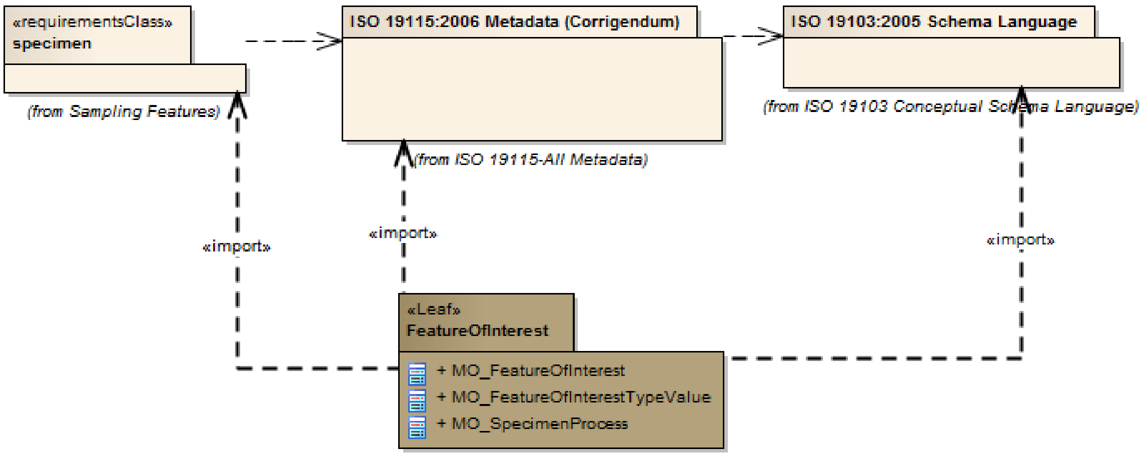

Figure 5 – Integration of MOLES3.4 Feature Of Interest Package with the schemas from ISO 19115, 19103 and the schema Sampling Features of ISO 19156.

The intention of the act of the observation is to provide values of some property carried by a domain-specific feature, e.g. atmosphere, a specific animal etc. However, the subject of the observation i.e. *featureOfInterest*, is not always coincident with the domain-specific feature except a kind of relationship. In particular, as stated in ISO 19156,

i) there are circumstances where the domain-specific feature, e.g. atmosphere, may not be fully accessible and a sampling regime has to be applied to provide a representative, i.e. a sample, of it. Therefore, the *featureOfInterest* is the sample whereas the domain-specific feature plays the role of the sampled feature with respect to the *featureOfInterest*. Typical examples of featureOfInterests after applying a sampling regime are: a specimen taken from a rock, a column in the atmosphere etc.

ii) estimates of the *observedProperty* might be obtained indirectly by targeting not a feature carrying the *observedProperty* but another proxy feature carrying a more convenient property associated with the o*bservedProperty*. Then, estimates of this property are converted to estimates of the wanted *observedProperty* by the application of an algorithm or processing chain which is part of the observation's procedure.

EXAMPLE A remote sensing observation might obtain the reflectance colour, when the investigation is actually interested in vegetation type and quality. The feature which contains reflectance colour is a scene or swath, while the feature carrying vegetation properties is a parcel or tract.

Furthermore, in some practical situations, both exceptions apply.

For this reason, ISO 19156 defines the concept of sampling feature (represented by the class *SF_SamplingFeature* – see Figure 6), a feature type which is involved in making observations concerning a domain-specific feature. In particular, Sampling Features provide:

i) an intermediate feature type that allows the assignment of primitive and intermediate properties within a processing chain, and

ii) a context for the description of sampling regimes.

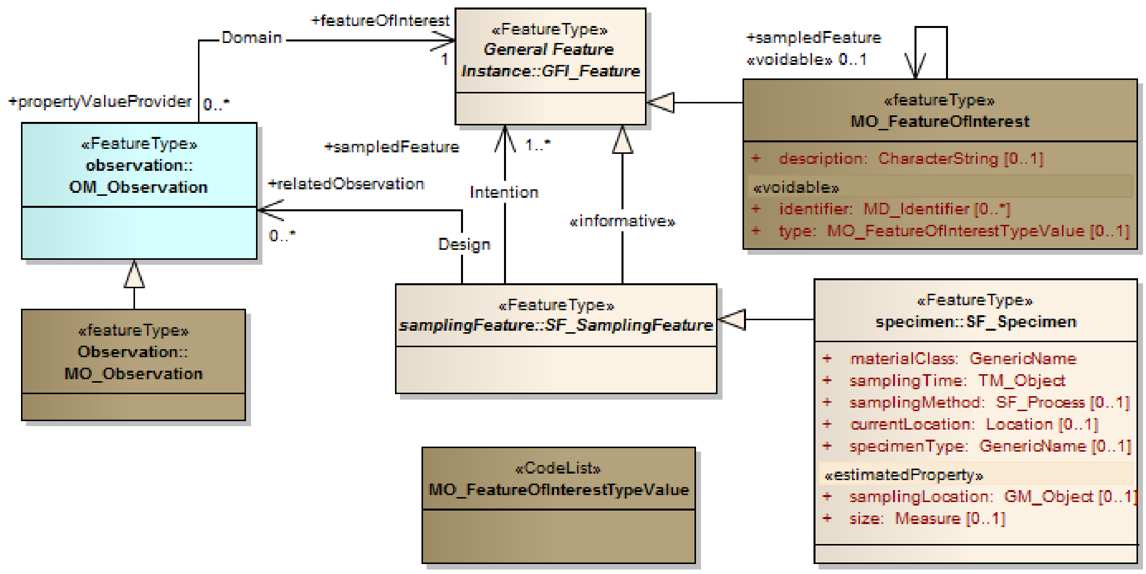

Figure 6 – The ISO 19156 class *SF_SamplingFeature* and the Properties (Attributes and associations) together their types and stereotypes of *MO_FeatureOfInterest* class.

Sampling features are artefacts of observational strategies and have no significant function outside of their role in the observation process. Where the *featureOfInterest* of an observation is a sampling feature, the *observedProperty* shall be a member of the sampling feature or of the sampled feature.

Figure 7 illustrates the concept of sampling feature in an example of time series measurements of air temperature (*observed property*) at a specific location (*a point sampling feature* which is the *feature of interest* of the observation) of the atmosphere above Chilbolton Observatory, UK (*sampled feature*).

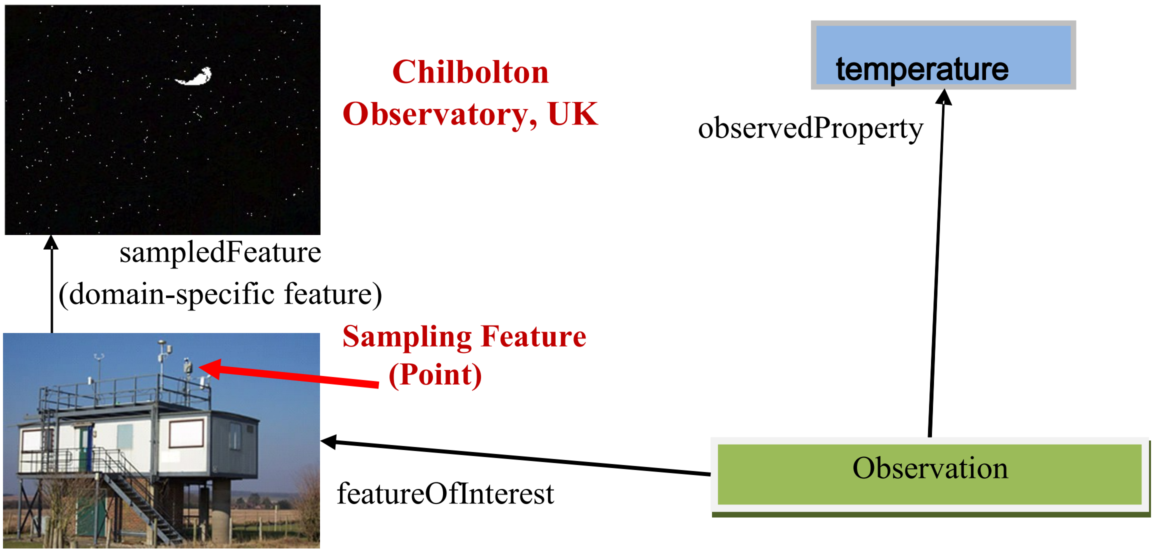

Figure 7 - Example of time series measurements of air temperature showing the use of the concepts: *sampling feature* and *sampled feature.*

Within the context of MOLES3.4 the subject of an observation is represented by (see Figure 6) :

i) the class *MO_FeatureOfInterest,* if it is a domain-specific or a sampling feature but not a specimen (see section 3.2.1);

ii) the ISO 19156 class *SF_Specimen*, if it is a physical sample (i.e. specimen), obtained for observation(s) carried out ex situ (see section 3.2.2) .

NOTE For the former case we do not use the ISO 19156 class *SF_SamplingFeature* and its specialisation *SF_SpatialSamplingFeature* class as there is no need for such a detailed model in the context of MOLES.

3.2.1 MO_FeatureOfInterest Class

The class *MO_FeatureOfInterest* (see Figure 8) represents a feature type that is the subject of the observation and carries the observed property. This feature is a domain-specific or a sampling feature.

3.2.1.1 Attributes

3.2.1.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the feature of interest or sampled feature.

3.2.1.1.2 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of feature of interest or sampled feature.

NOTE Sampling features not always have identifiers.

3.2.1.1.3 **type**

The attribute <<voidable>> *type:MO_FeatureOfInterestTypeValue* [0..1] {MOLES3.4} describes the type of the feature of interest or sampled feature. This property should reference a term from a controlled vocabulary. Typical examples are the spatiotemporal sampling geometries of CSML observation types [OGC 11-021 pending document].

The code list *MO_FeatureOfInterestTypeValue* is empty serving as a super-class for code lists which can specify the attribute values.

3.2.1.2 Associations

3.2.1.2.1 **sampledFeature**

If the class *MO_FeatureOfInterest* represents a sampling feature, the role <<voidable>> *sampledFeature:MO_FeatureOfInterest* [0..1] {MOLES 3.4} describes the principal domain feature.

3.2.2 SF_Specimen Class

The class *SF_Specimen* (See Figure 6) is defined within ISO 19156 and provides the modelling for physical samples derived from domain specific features. It is a specialisation of the base class *SF_SamplingFeature*.

3.2.2.1 Attributes

3.2.2.1.1 **materialClass**

The attribute *materialClass:GenericName* {ISO 19156:2011} shall provide a basic classification of the material type of the specimen.

EXAMPLE Soil, water, rock, aqueous, liquid, tissue, vegetation, food.

3.2.2.1.2 **samplingTime**

The attribute *samplingTime:TM_Object* {ISO 19108:2006 Temporal Schema} shall record when the specimen was retrieved from the sampled feature.

3.2.2.1.3 **samplingLocation**

If present, the attribute *samplingLocation:GM_Object* [0..1] {ISO 19107:2003 Spatial Schema} shall describe the location from where the specimen was obtained.

3.2.2.1.4 **samplingMethod**

If present, the attribute *samplingMethod:SF_Process* [0..1] {MOLES3.4} shall describe the method used to obtain the specimen from its sampled feature.

The type of this attribute is defined by the class *MO_SpecimenProcess* which is a specialisation of the ISO 19156 abstract class *SF_Process* (see section 3.2.3).

3.2.2.1.5 **currentLocation**

If present, the attribute *currentLocation:Location* [0..1] { ISO 19156:2011} shall describe the location of a physical specimen. This may be a storage location, such as a shelf in a warehouse or a drawer in a museum.

The union class Location (Figure 8) defines the type of this attribute and supports two attributes:

- The attribute *geometryLocation:GM_Object* {ISO 19107:2003 Spatial Schema} shall select a geometric representation of the location.
- The attribute *nameLocation:EX_GeographicDescription* {ISO19115:2006 Metadata (Corrigendum)} shall select a description of the location using text or an identifier.

3.2.2.1.6 **specimenType**

If present, the attribute *specimenType:GenericName* [0..1] {ISO 19103:2005 Schema Language} shall describe the basic form of the specimen.

EXAMPLE Polished section; core; pulp; solution.

3.2.2.1.7 **size**

If present, the attribute *size:Measure* [0..1] {ISO 19103:2005 Schema Language} shall describe a physical extent of the specimen. This may be length, mass, volume, etc., as appropriate for the specimen instance and its material class.

3.2.2.2 Associations

3.2.2.2.1 **processingDetails**

In the context of MOLES3.4 the property *processingDetails:MO_SpecimenProcess* [0..*] {MOLES3.4} describes the process which transforms an existing specimen. The type of this property, which is a specialisation of the ISO 19156 class *SF_Process*, is defined in section 3.2.3.

In many applications, specimen preparation procedures are applied to the material prior to its use in an observation. The association class *PreparationStep* (see Figure 8) is defined within ISO 19156 and shall link a *SF_Specimen* to a *SF_Process* that describes a phase of the specimen preparation. It shall support two attributes:

- The attribute *time:TM_Object* {ISO 19108:2006 Temporal Schema} shall describe the time that the SF_Process was applied to the SF_Specimen. It supports ordering of preparation steps.
- If present, the attribute *processOperator:CI_ResponsibleParty* [0..1] {ISO19115:2006 Metadata (Corrigendum)} shall describe the operator of the process involved in the preparation step.

3.2.3 MO_SpecimenProcess class

The ISO 19156 class *SF_Process* is abstract, and has no attributes, operations or associations. It serves as the base class for processes associated with the design and preparation of sampling features. Therefore, within MOLES3.4, this class is specialised to *MO_SpecimenProcess* class in order to model processes which generate or transform specimens.

MO_SpecimenProcess does not represent a process which produces a numerical artefact, i.e. an observation result, which is represented by the class *MO_Process* (see section 3.4). With the introduction of this class, MOLES3.4 addresses clearly the distinction between the acquisition of an observation (see section 3.5) and the acquisition of a specimen, from which an observation might be made.

3.2.3.1 Attributes

3.2.3.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the specimen process.

3.2.3.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the specimen process.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.2.3.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the specimen process.

3.2.3.1.4 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding individuals or organisations related to the specimen process.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.2.3.2 Associations

3.2.3.2.1 **component**

A specimen process may consist of more than one component. If so, the property *component:MO_SpecimenProcess* [0..*] {MOLES3.4} describes a specimen process component.

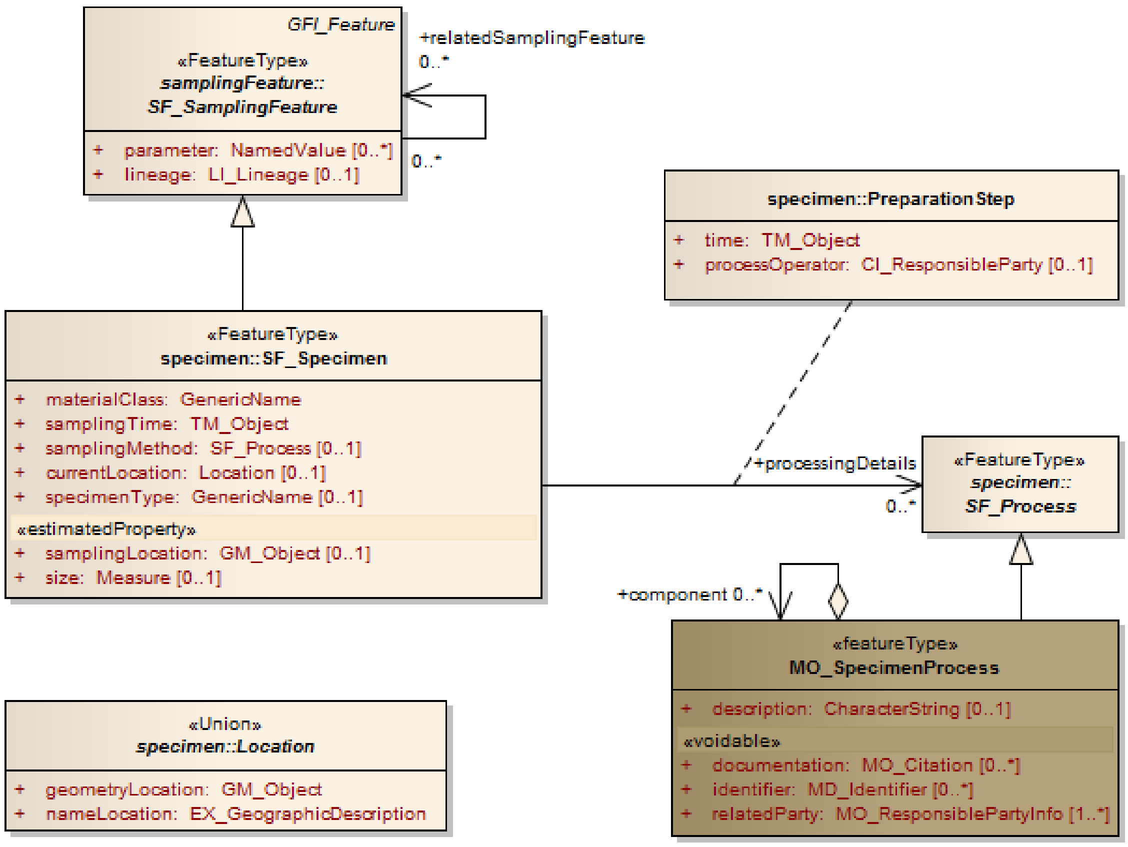

Figure 8 - Properties (Attributes and associations) and their types and stereotypes of *MO_SpecimenProcess* class.

3.3 Observable Property Package

This package provides the modelling of the phenomenon associated with the *featureOfInterest*. It defines the classes *MO_AbstractObservablePropert*, *MO_SimpleObservableProperty*, *MO_CompositeObservableProperty* and *MO_ObservablePropertyConstraint*. As shown in Figure 9 it uses elements only from the ISO 19103.

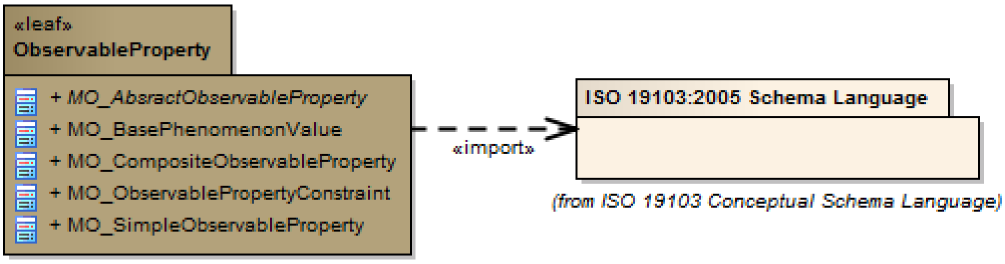

Figure 9 – Dependencies of Observable Property Package.

3.3.1 MO_AbstractObservableProperty Class

MO_AbstractObservableProperty (see Figure 10) is an abstract class that represents an *observedProperty* which can be either simple or composite represented by the classes *MO_SimpleObservableProperty* and *MO_CompositeObservableProperty* respectively. It supports two attributes.

3.3.1.1 Attributes

3.3.1.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a human readable description of *observedProperty*.

3.3.1.1.2 **label**

The attribute *label:CharacterString* [0..*] {ISO 19103:2005 Schema Language} indicates a human readable name by which an instance of *AbsractObservableProperty* is known.

EXAMPLE label can be: rain rate rainfall rate whereas *basePhenomenon* (see section 3.3.2) shall be: precipitationIntensity (obtained from a controlled vocabulary).

3.3.2 MO_SimpleObservableProperty Class

This concrete class (see Figure 10) describes an observable property which does not need to split further into components.

EXAMPLE temperature, total cloud cover, wind direction.

3.3.2.1 Attributes

3.3.2.1.1 **basePhenomenon**

The attribute *basePhenomenon:MO_BasePhenomenonValue* {MOLES 3.4} provides the name of a simple observable property (e.g. temperature) from a controlled vocabulary. The code list *MO_BasePhenomenonValue* is empty serving as a super-class for code lists which can specify the attribute values.

3.3.2.1.2 **constraint**

If present, the attribute *constraint:MO_ObservablePropertyConstraint* [0..1] {MOLES 3.4} shall provide further detail required for an instance of the *MO_SimpleObservableProperty*.

NOTE It is used when the *basePhenomenon* value of an instance of *MO_SimpleObservableProperty* from a controlled vocabulary is not sufficient to describe the *observedProperty*.

The type of this property is defined in section 3.3.4

EXAMPLE For an observation of time series of daily maximum temperature where only the term 'temperature" is available in a controlled vocabulary and not the term "daily maximum temperature":
(MO_SimpleObservableProperty.basePhenomenon>)temperature
(MO_SimpleObservableProperty.constraint>ObservablePropertyConstraint.label >)Daily maximum
(MO_SimpleObservableProperty.constraint>ObservablePropertyConstraint.description >)Maximum temperature recorded for each day

3.3.3 MO_CompositeObservableProperty Class

This concrete class (see Figure 10) describes observable properties consisting of two or more simple observable properties.

3.3.3.1 Associations

3.3.3.1.1 **component**

The property *component:MO_SimpleObservableProperty* [2..*] {MOLES 3.4} describes a simple observable property which is a component of the composite observable property.

3.3.4 MO_ObservablePropertyConstraint Class

The data type class MO_ObservablePropertyConstraint supports two attributes and one association.

3.3.4.1 Attributes

3.3.4.1.1 **label**

The attribute *description:CharacterString* {ISO 19103:2005 Schema Language} provides the text added to the name of an instance of *Simple Observable Property*.

3.3.4.1.2 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides the meaning added to an instance of *MO_SimpleObservableProperty* by the value of attribute label.

3.3.4.2 Association

3.3.4.2.1 **subConstraint**

An instance of MO_ObservablePropertyConstraint can be associated to other constraints. If so, the property *subConstraint:MO_ObservablePropertyConstraint* [0..*] {MOLES 3.4} shall describe the sub-constrains members.

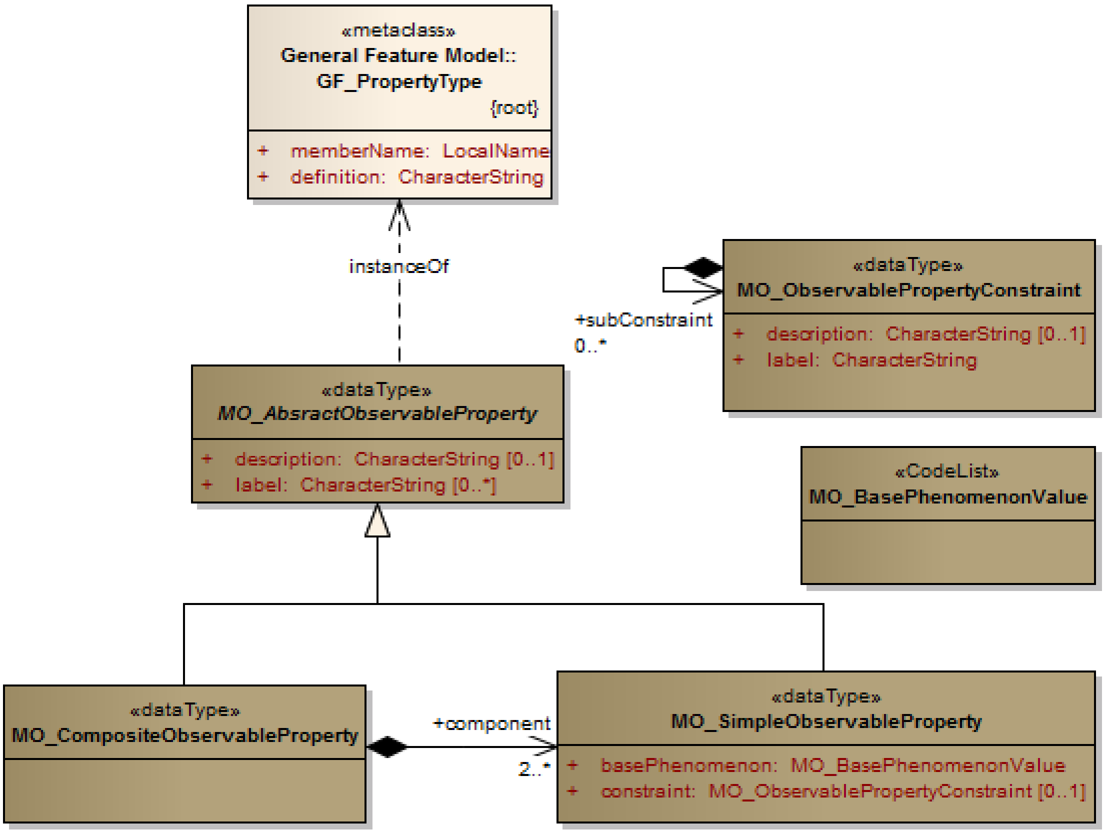

Figure 10 – Modelling of the Observable Property.

3.4 Observation Process Package

This package provides the modelling of the designated procedure used by the action of observation in order to assign a number, term or other symbol to a phenomenon. The procedure is represented in MOLES3.4 by the abstract class *MO_Process*. The dependencies of this package on the ISO 19100 series standards are shown in Figure 11.

A procedure is often an instrument or sensor but may be a process chain, human observer, an algorithm, a computation or simulator [*ISO 19156*].Therefore, a procedure may consist of more than one component. A component shall be either an acquisition or a numerical computation. An acquisition is defined as the procedure component that interacts with the *featureOfIinterest* (e.g. an atmospheric column, a specimen etc); typical examples include: instruments, sensors, human observer. A computation is defined as the procedure component which involves only pure computation. By definition a computation component does not interact with the *featureOfIinterest* but refers to it (e.g. predicted values of air temperature at a specific atmospheric column) and its output can be referred to the past, present or future compared with the time the computation was performed.

The class *MO_Process* (Figure 12) is a specialisation of the ISO 19156 *OM_Process* class without any attributes or operations and serves as the base class for all types of processes defined within MOLES, i.e. *MO_Acquisition* (see section 3.5), *MO_Computation* (see section 3.6) and *MO_CompositeProcess* (see section 3.4.1).

The procedure may be applied in-situ, remotely, or ex-situ with respect to the *featureOfInterest* and might be responsible for more than one observations. A description of the observation procedure provides or implies an indication of the reliability or quality of the observation result.

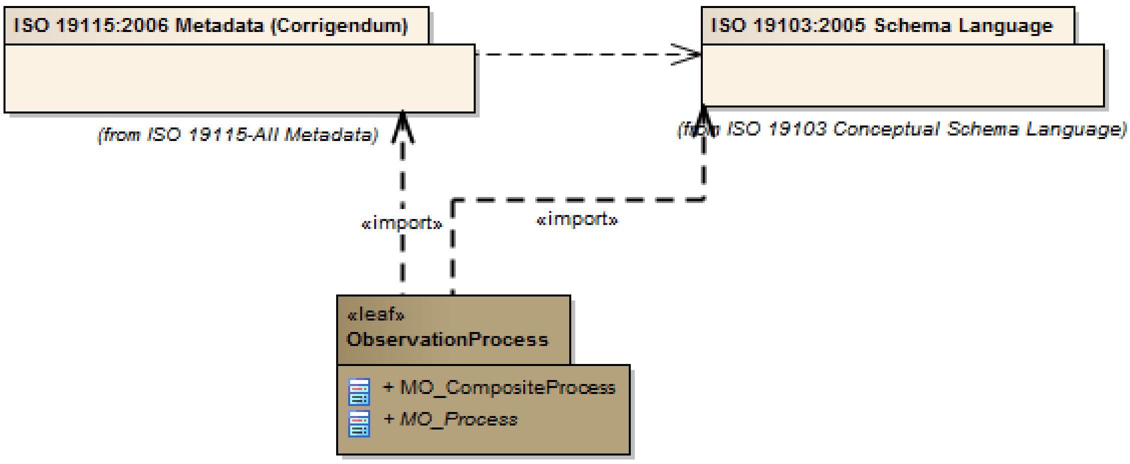

Figure 11 – Dependencies of Observation Process Package.

3.4.1 MO_CompositeProcess Class

The class *MO_CompositionProcess* represents processes consisting of more than one components. A component shall be either of acquisition or computation type. This aggregation (see Figure 12) may include only acquisition or computation types, or both. It supports 4 attributes and 2 associations.

3.4.1.1 Attributes

3.4.1.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the composite process.

3.4.1.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..1] {MOLES3.4} provides a reference to a documentation describing the composite process.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.4.1.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the composite process.

3.4.1.1.4 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding individuals or organisations related to the composite process.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.4.1.2 Associations

3.4.1.2.1 **computationComponent**

The property *computationComponent:MO_Computation* [0..*] {MOLES3.4} describes a computation component of the composite process.

The description of the class *MO_Computation* which defines the type of this property is given in section 3.6.

3.4.1.2.2 **acquisitionComponent**

The property *acquisitionComponent:MO_Acquisition* [0..*] {MOLES3.4} describes an acquisition component of the composite process.

The description of the class *MO_Acquisition* which defines the type of this property is given in section 3.5.

For any instance of *MO_CompositeProcess* the total number of computation and acquisition components shall be greater than 1.

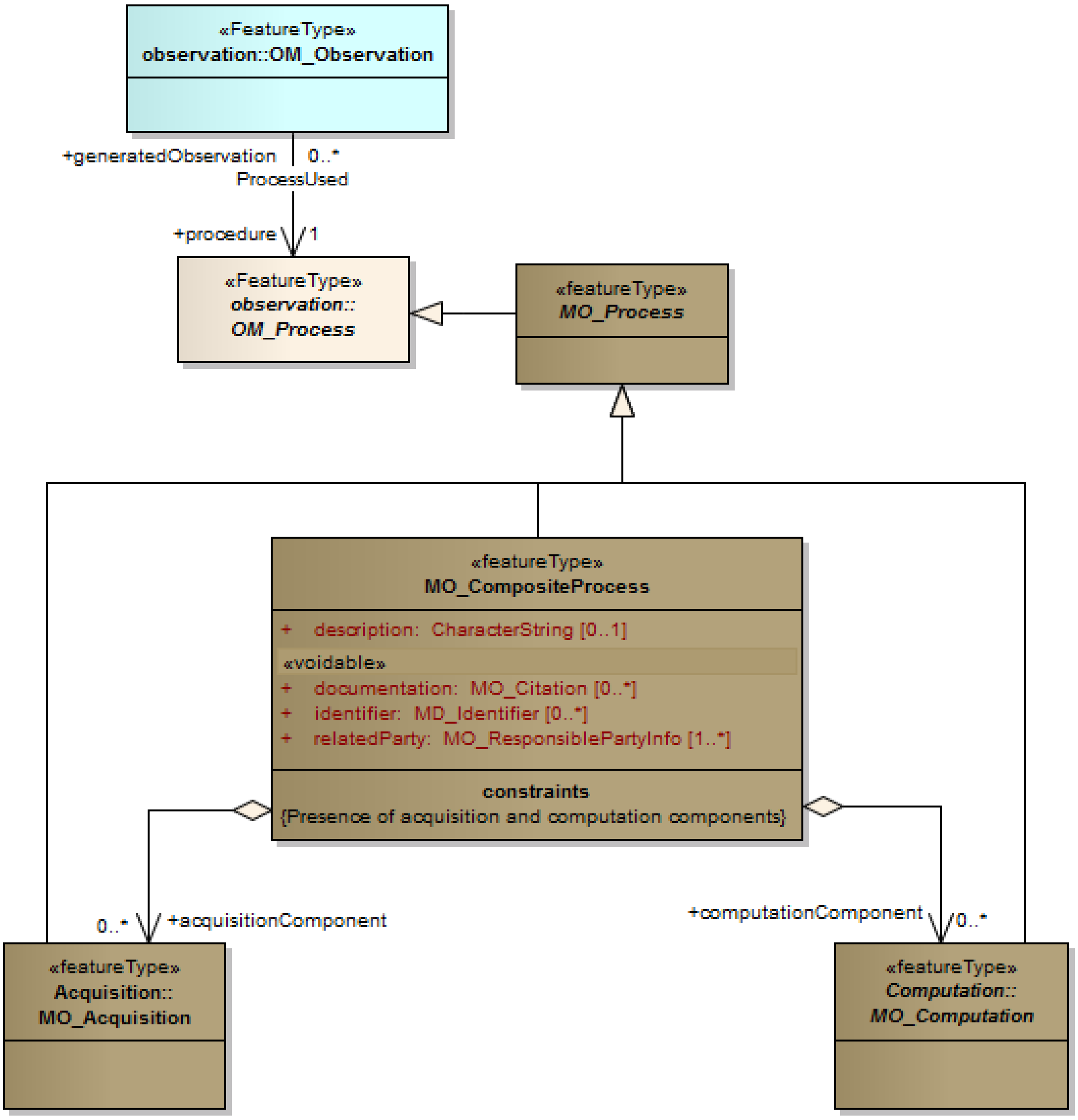

Figure 12 – Modelling of the Observation Process.

3.5 Acquisition Package

Acquisition is regarded the process which interacts with the feature of interest to provide a result. An acquisition always involves one or more instruments or sensors and might involve platforms and platform operations. Within the context of MOLES3.4 the concepts of instrument, platform and platform operation have the following definitions:

Instrument: Designations for the measuring instruments which interact with the feature of interest; this definition has been adopted from ISO 19115-2:2009.

Platform: an identifiable object which can either
1. bring the acquisition instrument(s) to the appropriate environment (e.g. aircraft, ground station);
2. create the appropriate environment for the acquisition instrument(s) (e.g. a cloud chamber) ;
3. provide the appropriate operational conditions to the instruments (e.g. a ground station provides electricity)

in order data to be acquired according to the observation objectives.

Operation: Information about a platform operation - e.g. flight of an aircraft - which needed for the data acquisition. The concept of platform operation applies only to platforms that are in motion during the acquisition e.g. a satellite – geostationary or orbital, an aircraft, a ship and not the operation of a static platform such as a ground station. Therefore not all platforms require an operation in MOLES context.

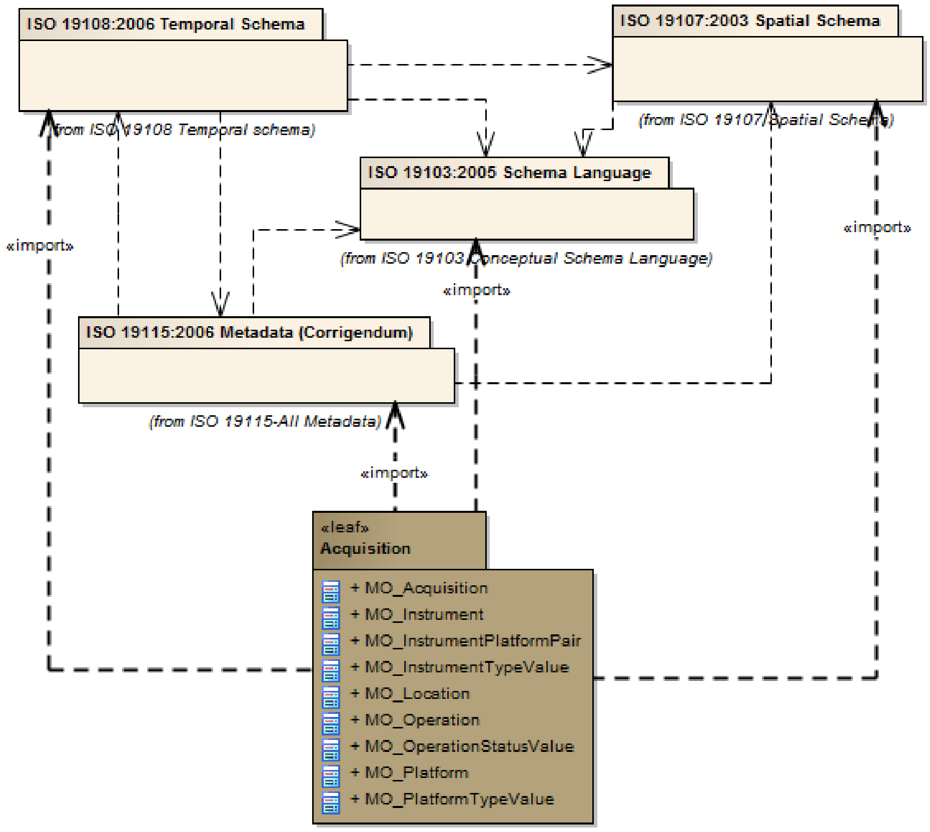

Figure 13 – Dependencies of Acquisition Package.

Figure shows 13 the dependencies of the Acquisition Package whereas Figure 14 illustrates the structure of the acquisition modelling with the classes *MO_Insrument*, *MO_Platform* and *MO_Operation* representing the concepts of *Instrument*, *Platform* and *Operation* respectively.

3.5.1 MO_Acquisition class

3.5.1.1 Attributes

3.5.1.1.1 description

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the acquisition.

3.5.1.1.2 documentation

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the acquisition.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.5.1.1.3 identifier

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the acquisition.

3.5.1.1.4 instrumentPlatformPair

The attribute <<voidable>> *instrumentPlatformPair:MO_InstrumentPlatformPair* [0..*] {MOLES 3.4} provides a notification of the platform and the mounted on it instrument for the specified acquisition component.

The type of this attribute is provided by the class *MO_InstrumentPlatformPair* which is defined in section 3.5.5.

3.5.1.1.5 outputDescription

The attribute <<voidable>> *outputDescription:MO_InputOutput* [0..1] {MOLES 3.4} provides a description of the acquisition output. The description does not involve the actual values of the output data which are specific to the observation action.

The structure of the value of this attribute is provided by the class *MO_InputOutput* which is defined in section 3.6.2 .

3.5.1.1.6 relatedParty

The attribute <<voidable>> *relatedParty:MO_ResponiblePartyInfo* [0..*] {MOLES3.4} provides information regarding individuals or organisations related to the acquisition.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.5.1.2 Associations

3.5.1.2.1 **instrument**

The association *UsedInstrument* shall link the *MO_Acquisition* to the *MO_Instrument* used to generate the result. Instances of *MO_Instrument* have the role *instrument* (*instrument:MO_Instrument* [1..*] {MOLES3.4}) *w*ith respect to the acquisition.

NOTE Any instance of *MO_Acquisition* shall have at least one instrument which might be used in more than one acquisitions.

3.5.1.2.2 **platform**

If present, the association *UsedPlatform* shall link the *MO_Acquisition* to the involved *MO_Platform* used to generate the result. Instances of *MO_Platform* have the role *platform* (<<voidable>> *platform:MO_Platform* [0..*] {MOLES3.4}) with respect to the acquisition.

3.5.1.2.3 **operation**

If present, the association *UsedOperation* shall link the *MO_Acquisition* to the involved *MO_Operation* used to generate the result. Instances of *MO_Operation* have the role *operation* (<<voidable>> *platform:MO_Operation* [0..*] {MOLES3.4}) with respect to the acquisition.

3.5.2 MO_Instrument Class

3.5.2.1 Attributes

3.5.2.1.1 **description**

The attribute *description:CharacterString* [0..1]{ISO 19103:2005 Schema Language} provides a free text description of the instrument.

3.5.2.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the instrument.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.5.2.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the instrument.

3.5.2.1.4 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding organisations and/or individuals related to the instrument.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.5.2.1.5 **type**

The attribute <<voidable>> *type:MO_InstrumentTypeValue* {MOLES3.4} describes the type of the instrument. This property should reference a term from a controlled vocabulary (e.g. radiometer, rain gauge).

The code list *MO_InstrumentTypeValue* is empty serving as a super-class for code lists which can specify the attribute values.

3.5.2.2 Associations

3.5.2.2.1 **member**

An instance of *MO_Instrument* class can be an aggregation of instruments. If so, instances of *MO_Instrument* have the role *member* (<<voidable>> *member:MO_Instrument* [0..*] {MOLES3.4}) with respect to the aggregation.

EXAMPLE FAAM (Facility for Airborne Atmospheric Measurements) CORE Radiometers, performing total, red and infrared upwelling/downwelling radiation measurements. The FAAM broadband radiometer instruments include 2 clear dome pyranometers, 2 red dome pyranometers and two pyrgeometers.

3.5.3 MO_Platform Class

3.5.3.1 Attributes

3.5.3.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the platform.

3.5.3.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the platform.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.5.3.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the platform.

3.5.3.1.4 **location**

The attribute <<voidable>> *location:MO_Location* [0..1] {MOLES 3.4} describes the location of the platform. Location is only applicable to static platforms or geostationary satellites.

The type of the attribute, *MO_Location*, is defined in section 3.5.6

3.5.3.1.5 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding organisations and/or individuals related to the platform.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.5.3.1.6 **type**

The attribute <<voidable>> *type:MO_PlatformTypeValue* {MOLES3.4} describes the type of the platform. This property should reference a term from a controlled vocabulary (e.g. “balloon”, “satellite”).

The code list *MO_PlatformTypeValue* is empty serving as a super-class for code lists which can specify the attribute values.

3.5.3.2 Association

3.5.3.2.1 **childPlatform**

A platform may be a member of a complex of parent and child platforms. If so, instances of *MO_Platform* have the role *childPlatform* (<<voidable>> *childPlatform:MO_Platform* [0..*] {MOLES3.4}) with respect to the parent platform MO_Platform.

3.5.4 MO_Operation class

3.5.4.1 Attributes

3.5.4.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the platform operation.

3.5.4.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*]{MOLES3.4} provides a reference to a documentation describing the platform operation.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.5.4.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the platform operation.

3.5.4.1.4 **location**

The attribute <<voidable>> *location:MO_Location* {MOLES 3.4} describes the location of the platform operation.

EXAMPLE A flight line or a ship track for a platform such as an aircraft or a ship respectively.

The type of the attribute, *MO_Location*, is defined in section 3.5.6

3.5.4.1.5 **operationTime**

The attribute <<voidable>> *operationTime: TM_Object* {ISO 19108:2006 Temporal Schema} records the time when the operation occurred.

NOTE The act of an observation using a specified acquisition procedure occurs during the operation of the platform involved in the data acquisition.

EXAMPLE During the operation of ENVISAT satellite, the onboard AATSR (Advanced Along Track Scanning Radiometer) acquired data needed for measuring the sea surface temperature.

3.5.4.1.6 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding organisations and/or individuals related to the platform operation.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.5.4.1.7 **status**

The attribute <<voidable>> *status:MO_OperationStatusValue* {MOLES3.4} describes the status of the platform operation.

The values of the attribute should be taken from a controlled vocabulary (e.g. “ongoing”, “completed”). The code list *MO_OperationStatusValue* is empty serving as a super-class for code lists which can specify the attribute values. A suggested code list is the ISO 19115:2006 Metadata (Corrigendum) code list *MD_ProgressCode* (see section A.3)

3.5.4.2 Associations

3.5.4.2.1 **childOperation**

An instance of *MO_Operation* class can be an aggregation of platform operations. If so, instances of *MO_Operation* have the role *childOperation* (<<voidable>> *childOperation:MO_Operation* [0..*] {MOLES3.4}) with respect to the aggregation.

3.5.4.2.2 **platform**

The property <<voidable>> *platform:MO_Platform* [1..*] {MOLES 3.4} describes the operated platform.

3.5.5 MO_InstrumentPlatformPair Class

The data type class *MO_InstrumentPlatformPair* specifies a pair consisting of a platform and an instrument mounted on the platform. It supports two attributes.

3.5.5.1 Attributes

3.5.5.1.1 **platform**

The attribute *platform:MO_Platform* {MOLES 3.4} describes the platform hosting the specified instrument.

3.5.5.1.2 **instrument**

The attribute *instrument:MO_Instrument* {MOLES 3.4} describes the instrument mounted on the specified platform.

3.5.6 MO_Location Class

The data type class *MO_Location* provides the location of an object by a geometric or geographic description or both. It supports two attributes.

3.5.6.1 Attributes

3.5.6.1.1 **geometryLocation**

The attribute *geometryLocation:GM_O*bject [0..1] {ISO 19107:2003 Spatial Schema} provides a geometric representation of the location.

3.5.6.1.2 **nameLocation**

The attribute *geometryLocation:EX_GeographicDescription* [0..1] {ISO 19115:2006 Metadata (Corrigendum)} provides a geographic description of the location using text or an identifier.

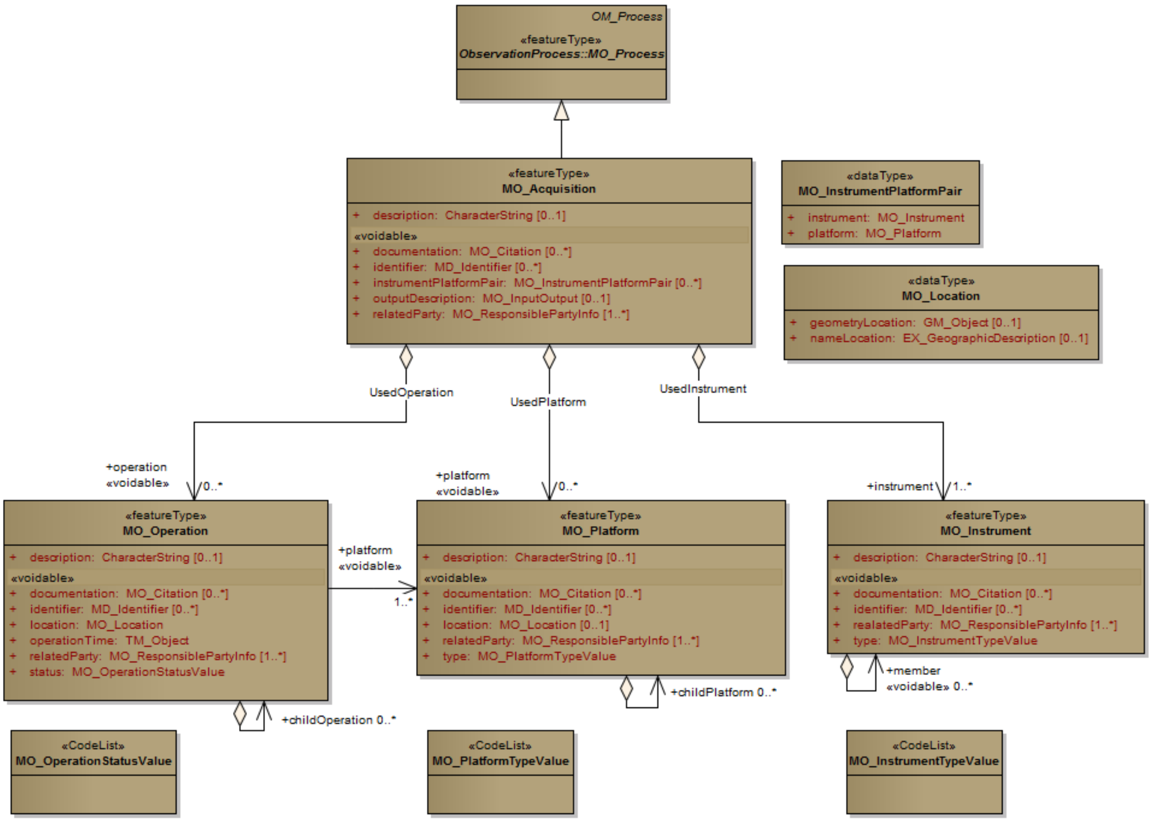

Figure 14 - Acquisition Model.

3.5.7 Location of the acquisition

Regarding the geospatial location of an acquisition it should be taken in to account that an acquisition consists of one or more of the mutually exclusive combinations:

a) {use of instrument only};
b) {use of instrument mounted on a platform without requiring operation}; and
c) {use of instrument mounted on platform requiring operation}.

Therefore, if the location (or locations) associated with an acquisition is of interest for the observation utilising the acquisition then:
For case a) the location is specific to the event of the observation, as the instrument can be used in other observations as well, thus the acquisition location should be provided by the attribute *MO_Observation.parameter* (see section 3.1.1.1.5).

> **EXAMPLE** if two observations, Obs1 and Obs2, use the same instrument (e.g. INT567) at the locations specified by the geographic descriptions "Location1" and "Location2" respectively (instances of *EX_GeographicDescription* {ISO 19115:2006}) , then:
>
> For Obs1:
> (MO_Observation.parameter>NamedValue. name>)LocationOf Instrument_INT567
> (MO_Observation.parameter>NamedValue. value>)"Location1"
>
> For Obs2:
> (MO_Observation.parameter>NamedValue. name>)LocationOf Instrument_INT567
> (MO_Observation.parameter>NamedValue. value>)"Location2"

For case b) the instrument is mounted on a static platform (or a geostationary satellite) and the attribute *MO_Platform.location* (see section 3.5.3.1.4) provides the location of the acquisition.

For case c) the instrument is mounted on a platform which is in motion during the acquisition and the attribute *MO_Operation.location* (see section 3.5.4.1.4) provides the location of the acquisition.

The location (or locations) associated with the acquisition should not be confused with the location of the *featureOfInterest (*i.e. spatial domain of the *featureOfInterest)* of the observation utilising the acquisition. There may be circumstances where these two locations are the same but not always. In remote sensing, for example, an instrument is located remotely from the *featureOfInterest*.

3.5.8 ISO 19115-2 pattern for acquisition

ISO 19115-2:2009 introduces the classes (see Figure 15):
MI_Instrument : designations for the measuring instruments.
MI_Platform: designation of the platform used to acquire the dataset
MI_Operation: representing designations for the operation used to acquire the dataset.

MOLES model follows the ISO 19115-2 pattern but introduces its own concepts of instrument, platform and platform operation and provides a more generic approach of the acquisition modelling. Thus, an instrument could be mounted on more than one platform.

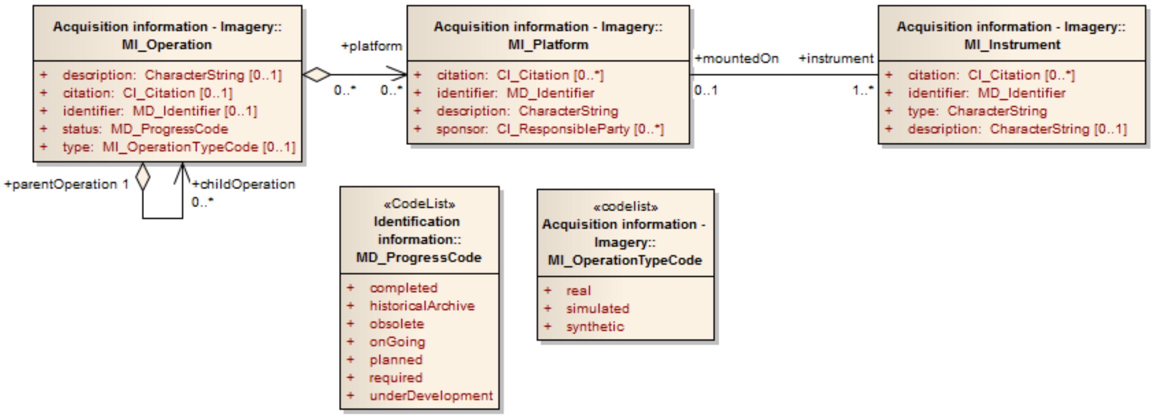

Figure 15 - Properties (Attributes and associations) of ISO 19115-2 classes: MI_Instrument, MI_Platform and MI_Operation.

3.6 Computation Package

This package defines the class *MO_Computation* for modelling processes which involve only numerical computation. It is abstract without attributes or operations and serves as the base class for non-acquisition process components. The dependencies of this package on the ISO 19100 series standards are shown in Figure 16.

In this schema *MO_Computation* (see Figure 17) is specialised to the concrete class *MO_Processing* which provides a brief description of a computation component together with a description of the relevant anticipated output and required input.

NOTE If a more thorough description is required than the one provided by the class *MO_Processing* classes developed within METAFOR framework [*Lawrence et al, 2012*] can be used to implement the abstract class *MO_Computation*.

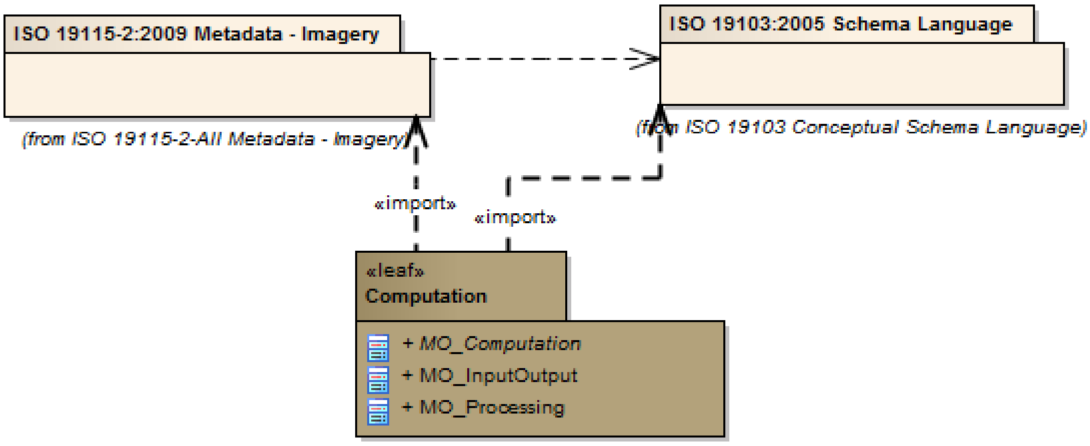

Figure 16 – Dependencies of Computation Package.

3.6.1 MO_Processing class

3.6.1.1 Attributes

3.6.1.1.1 **description**

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the computation component.

3.6.1.1.2 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the computation component.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.6.1.1.3 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the computation component.

3.6.1.1.4 **processingInput**

The attribute <<voidable>> *processingInput:MO_InputOutput* [0..*] {MOLES 3.4} provides a description of the required input of *MO_Processing*. The description does not involve the values of the input data. The actual values, if required, shall be stored in *OM_Observation.parameter>NamedValue* as they are specific to the event of the observation and not to the applied process (see section 3.1.1.1.5).

The class *MO_InputOutput* which defines the type of this attribute is described in section 3.6.2.

3.6.1.1.5 **processingOutput**

The attribute <<voidable>> *processingOutput:MO_InputOutput* {MOLES 3.4} provides a description of *MO_Processing* output. The description does not involve the actual values of the output data.

The class *MO_InputOutput* which defines the type of this attribute is described in section 3.6.2.

3.6.1.1.6 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding organisations and/or individuals related to the *MO_Processing* instance.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.6.1.1.7 **softwareReference**

The attribute <<voidable>> *softwareReference:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the relevant processing software.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.6.1.2 Associations

3.6.1.2.1 **algorithm**

The property <<voidable>> *algorithm:LE_Algorithm* [0..*] {ISO 19115-2:2009 Metadata-Imagery} describes the algorithm(s) on which the processing is based on.

3.6.2 MO_InputOutput class

The data type class *MO_InputOutput* provides information describing the required input or produced output by the application of a process component. The description does not involve the values of the data. The class supports two attributes.

3.6.2.1 Attributes

3.6.2.1.1 name

The attribute *name:CharacterSting* [0..1]{ISO 19103:2005 Schema Language} provides a name or other identification of input or output data relevant to a process component.

3.6.2.1.2 description

The attribute *description:LE_Source* {ISO 19115-2:2009 Metadata-Imagery} provides a structured description of input or output data relevant to a process component.

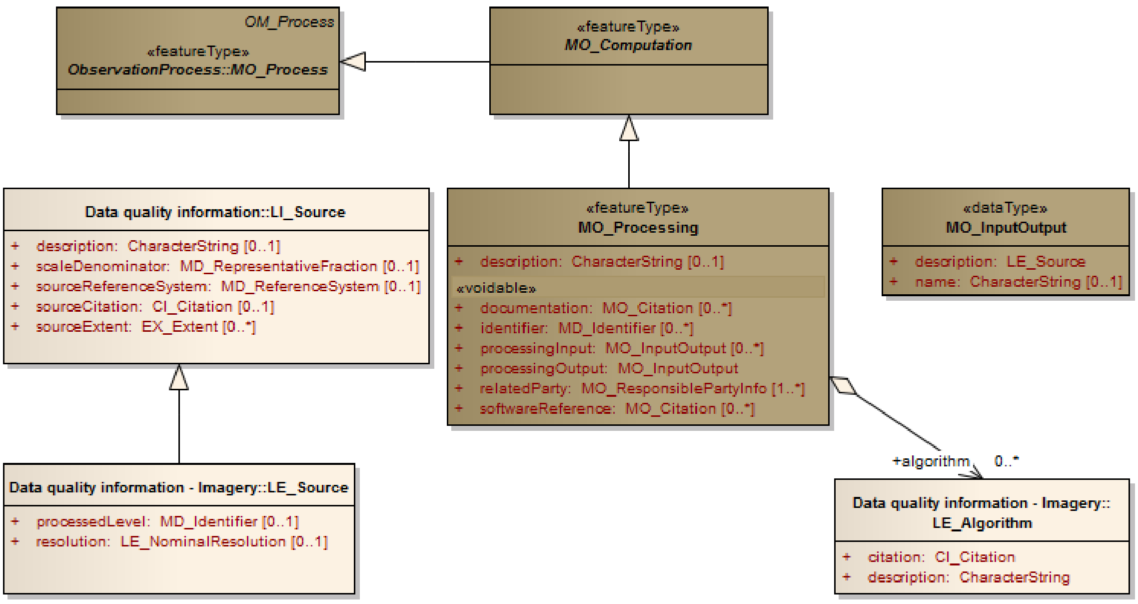

Figure 17 - Computation model.

3.7 Result Package

The act of the observation produces a numerical artefact, i.e. the observation result. MOLES is not particularly concerned with details of observation result structures but is aimed at providing the required metadata in order to make this result fully understandable and exploitable. Therefore, in MOLES environment, observation result is regarded the set of metadata for accessing and obtaining the actual values assigned to a phenomenon.

For a discussion how MOLES3.4 result is related to the actual obtained values see Annex D.

This package uses elements from ISO 19115 (see Figure 18) and defines the data type classes *MO_Result* and *MO_OnlineResource*.

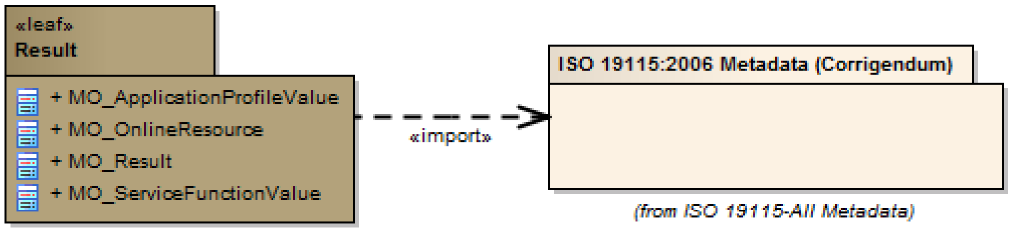

Figure 18 – Dependencies of the Result Package.

3.7.1 MO_Result class

An instance of MO_Result (see Figure 19) class represents a list of resources for on line access to the observation result. The result can be accessed as a whole or partly. The result may be consisted of segments accessible with their own resources.

3.7.1.1 Attributes

3.7.1.1.1 source

The attribute *source:MO_OnlineResource* [0..*] {MOLES 3.4} provides information about resources which are used to access the observation result (as a whole or partly).

3.7.1.1.2 sample

The attribute <<voidable>> *sample:MO_OnlineResource* [0..*] {MOLES 3.4} provides information about resources which are used to access a sample of the observation result.

3.7.1.2 Association

3.7.1.2.1 segment

The result may be consisted of segments accessible with their own resources. If so, instances of *MO_Result* have the role *segment* (*segment:MO_Result* [0..*] {MOLES3.4}) with respect to the aggregating result.

3.7.2 MO_OnlineResource class

This data type class (see Figure 19) represents information about on line sources. It is a specialisation of the ISO 19115:2006 Metadata (Corrigendum) *CI_OnlineResource* class. A description of these properties, together with a description of the inherited properties as they perceived in the MOLES domain, are given in this section.

3.7.2.1 Inherited Attributes

3.7.2.1.1 applicationProfile

The attribute *applicationProfile:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides the type (format) of the stored result accessible from the online resource.

This property should reference a term from a controlled vocabulary (e.g. NetCDF, csv). Therefore, the value of the attribute, which is of type *CharacterString*, shall be obtained from the code list *MO_ApplicationProfileValue*. This is empty and serves as a super-class for code lists which can specify the attribute values.

3.7.2.1.2 description

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a detailed text description of what the online resource is/does.

3.7.2.1.3 function

The attribute *description:CI_OnLineFunctionCode* [0..1] {ISO 19115:2006 Metadata (Corrigendum)} indicates the function performed by the online resource.

The values provided by the code list *CI_OnLineFunctionCode* are described in section A.4.

3.7.2.1.4 linkage

The attribute *linkage:URL* {ISO 19115:2006 Metadata (Corrigendum)} provides a location (address) for on line access using a Uniform Resource Locator/Uniform Resource Identifier address or similar addressing scheme such as http://www.statkart.no/isotc211.

3.7.2.1.5 name

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a name of the online resource.

3.7.2.1.6 protocol

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} indicates the connection protocol to be used e.g. http, ftp, file.

3.7.2.2 MOLES defined Attribute

3.7.2.2.1 serviceFunction

The attribute *serviceFunction:MO_ServiceFunctionValue* [0..1] {MOLES 3.4} specifies a distinct part of the functionality that is provided by the resource through interfaces, e.g. OGC services [*ISO 19119*], in addition to functions described by the inherited attribute *function:CI_OnlineFunctionCode*.

This property should reference a term from a controlled vocabulary (e.g. wcs). The code list *MO_ServiceFunctionValue* is empty serving as a super-class for code lists which can specify the attribute values.

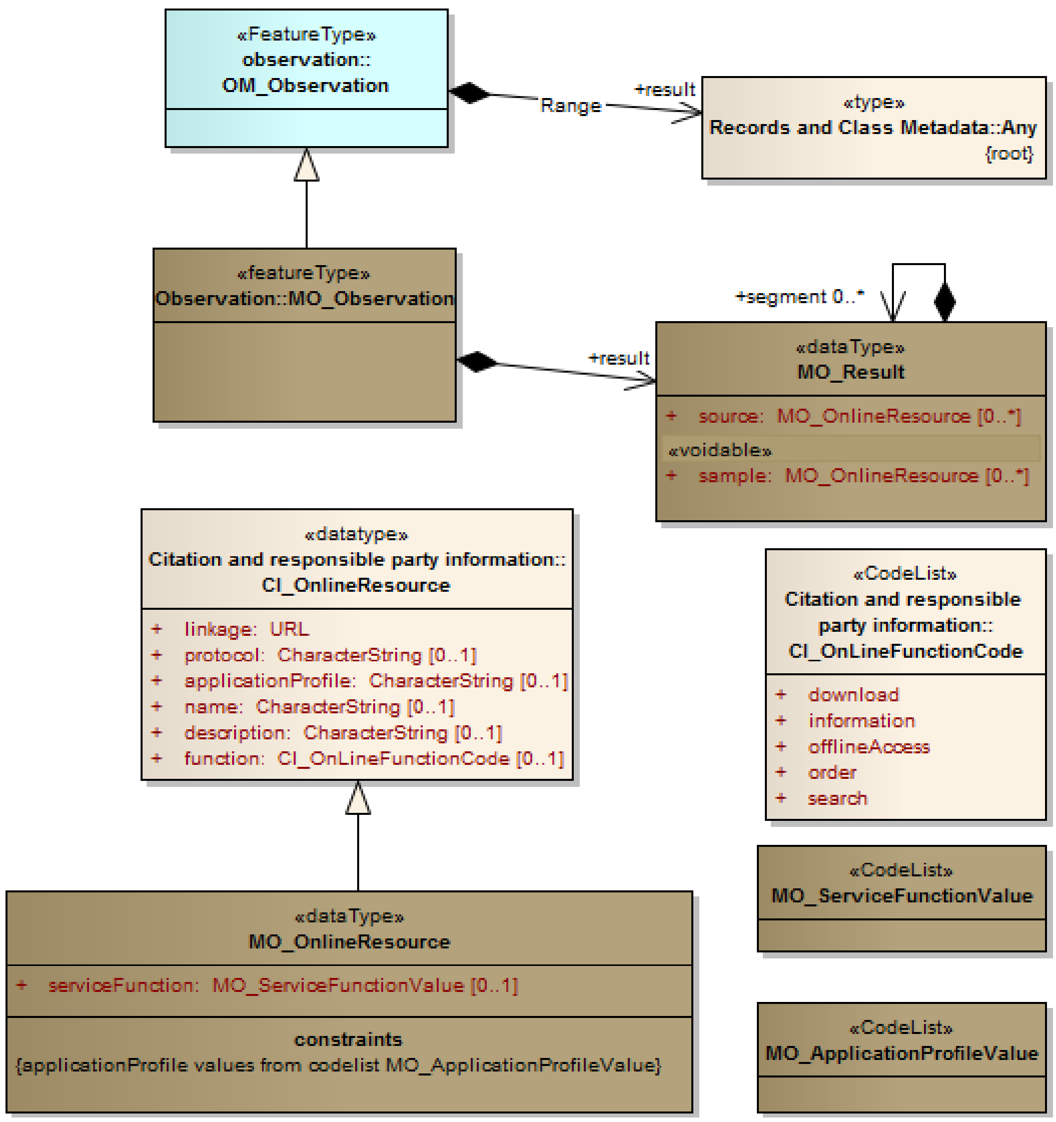

Figure 19 – MOLES3.4 Result model.

3.8 Project Package

This package provides the modelling of the activities under which actions are undertaken to produce datasets. It utilises elements from the ISO 19103 and 19115 (see Figure 20) and defines the class *MO_Project*.

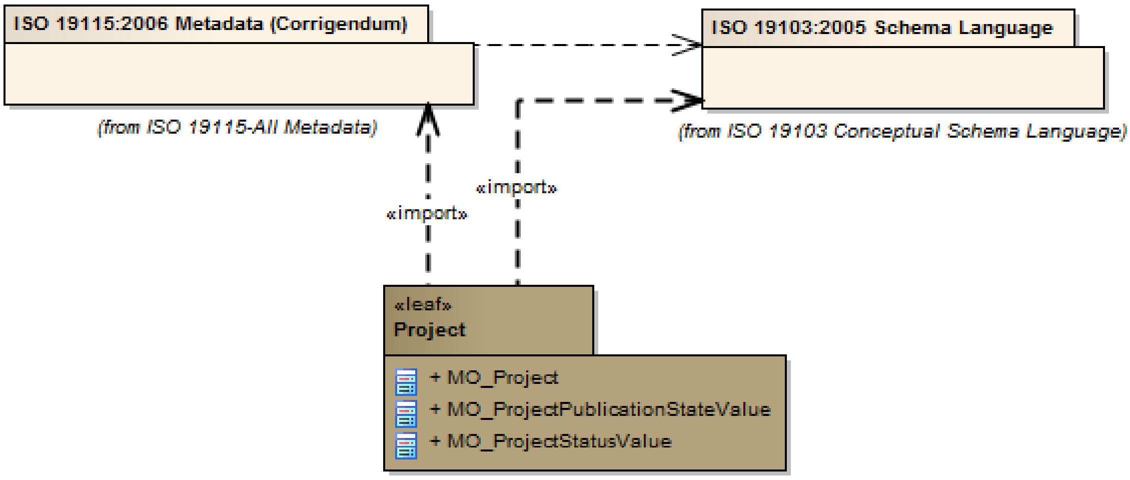

Figure 20 – Dependencies of the Project Package.

3.8.1 MO_Project class

An instance of *MO_Project* class (see Figure 21) represents an identifiable activity/project designed to accomplish a set of objectives. A typical sequence of data capturing involves one or more projects for which a number of observations are conducted, using appropriate tools and methods, to produce data sets.

EXAMPLE The UK Surface Ocean /Lower Atmosphere Study (UK SOLAS) was a Directed Mode programme of the Natural Environment Research Council, UK (NERC). The overall aim of UK SOLAS was to advance understanding of environmentally significant interactions between the atmosphere and ocean, focusing on material exchanges that involve ocean productivity, atmospheric composition and climate.

3.8.1.1 Attributes

3.8.1.1.1 abstract

The attribute *abstract:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a brief narrative summary of the project.

3.8.1.1.2 description

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the project including its intentions.

3.8.1.1.3 publicationState

The attribute *publication:MO_ProjectPublicationStateValue* [0..1] {MOLES3.4} indicates the state of project metadata record.

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ProjectPublicationStateValue* is empty serving as a super-class for code lists which can specify the attribute values, for example, the *CEDA_PublicationStateValue* code list (see section A.1).

3.8.1.1.4 **dmp**

The attribute <<voidable>> *dmp:MO_Citation* [0..1] {MOLES 3.4} provides a reference to the documentation describing the Data Management Plan of the project. Data Management Plan provides the planning which must cover the practical arrangements while the project is running and the subsequent maintenance and long-term curation of the data sets.

3.8.1.1.5 **documentation**

The attribute <<voidable>> *documentation:MO_Citation* [0..*] {MOLES3.4} provides a reference to a documentation describing the project.

For the class *MO_Citation* which defines the type of this attribute see section 3.10.1.

3.8.1.1.6 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [1..*]{ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the project.

3.8.1.1.7 **keywords**

The attribute <<voidable>> *keywords:MD_Keywords* [1..*] {ISO 19115:2006 Metadata (Corrigendum)} provides category keywords, their type, and reference source.

3.8.1.1.8 **relatedParty**

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding individuals or organisations related to the project.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.8.1.1.9 **status**

The attribute <<voidable>> *status:MO_ProjectStatusValue* {MOLES3.4} indicates the status of the project regarding its objectives.

EXAMPLE Measurements of rain rate at Chilbolton Observatoty,UK are being conducted since 2000. The status of this project is *onGoing*.

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ProjectStatusValue* is empty serving as a super-class for code lists which can specify the attribute values. A suggested code list is the ISO 19115:2006 Metadata (Corrigendum) code list *MD_ProgressCode* (see section A.3)

3.8.1.2 Associations

3.8.1.2.1 **subProject**

A project may be a member of a complex of parent and child-projects. If so, the property *subProject:MO_Project* [0..*] {MOLES3.4} shall describe a Sub-project of the project.

3.8.1.2.2 observationCollection

A project may contribute to a number of Observation collections. The property *observationCollection:MO_ObservationCollection* [0..*] {MOLES3.4} describes a collection of existing observations associated with the project. The type of the attribute is defined in section 3.9.

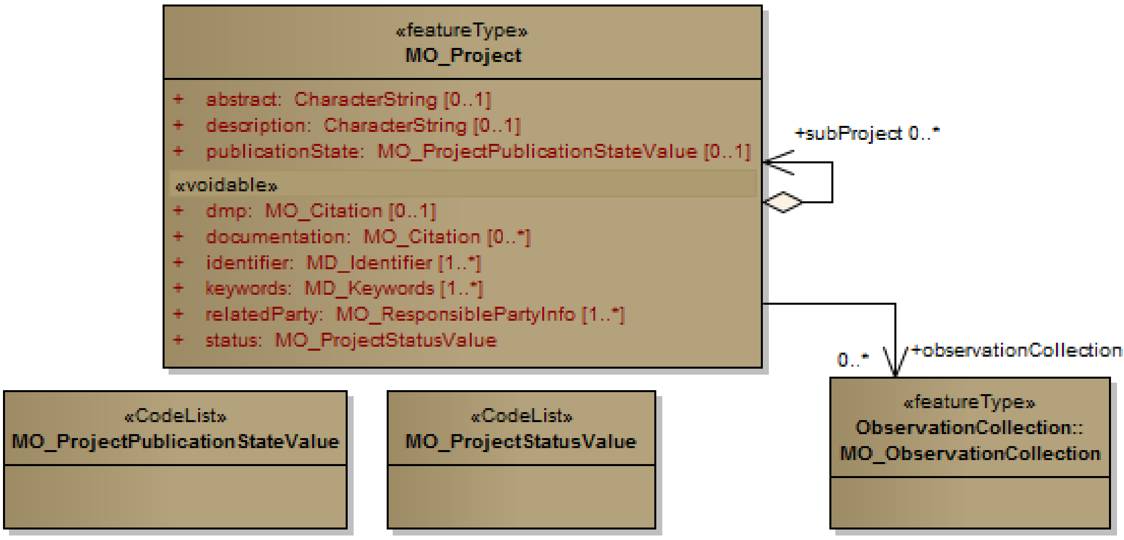

Figure 21 - Properties (Attributes and associations) and their types and stereotypes of *MO_Project* class.

3.9 Observation Collection Package

The data can be organised in groups according to user requirements or the wishes of data the manager. The class *MO_ObservationCollection* represents collections of existing observations. Observations can be aggregated in collections organised with significantly more flexibility than including only those observations conducted in support of the same project.

EXAMPLE A collection including all the conducted observations using the FAAM (Facility for Airborne Atmospheric Measurements) aircraft operated in 2005.

An observation may be aggregated in more than one observation collections.

The dependencies of this package on the ISO 19100 series standards are shown in Figure 22.

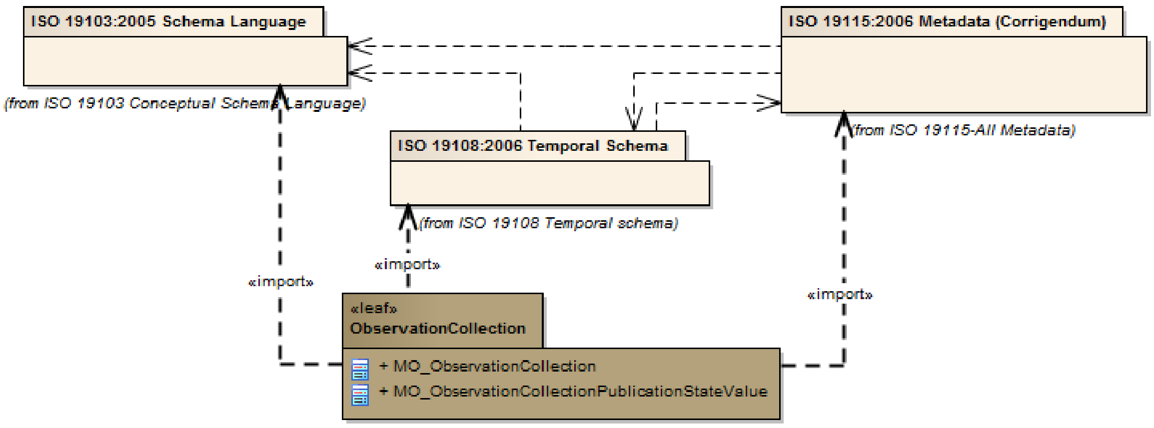

Figure 22 – Dependencies of Observation Collection Package.

3.9.1 MO_ObservationCollection Class

3.9.1.1 Attributes

3.9.1.1.1 description

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the observation collection including the criteria for its creation. Criteria could be a common observation process, an observable property etc.

3.9.1.1.2 publicationState

The attribute *publication:MO_ObservationCollectionStateValue* [0..1] {MOLES3.4} indicates the state of observation collection metadata record.

The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ObservationCollectionPublicationStateValue* is empty serving as a super-class for code lists which can specify the attribute values, for example, the *CEDA_PublicationStateValue* code list (see section A.1).

3.9.1.1.3 **additionalMetadata**

The attribute <<voidable>> *additionalMetadata:MO_Citation* [0..*] {MOLES3.4} provides the resources for any non ISO conformed metadata about the Observation Collection. This attributes complements the property *isoMetadata*.

The class *MO_Citation* which defines the type of this attribute is described in section 3.10.1.

3.9.1.1.4 **identifier**

The attribute <<voidable>> *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the observation collection.

3.9.1.1.5 **phenomenonTime**

The attribute <<voidable>> *phenomenonTime:TM_Object* {ISO 19108:2006 Temporal Schema} describes the extent of the temporal domains for all the observations of the collection.

3.9.1.1.6 **geographicExtent**

The attribute <<voidable>> *geographicExtent:EX_GeographicExtent* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} describes the geographic area, e.g. bounding polygon, within which the results of all members of an observation collection are available.

3.9.1.1.7 **vericalExtent**

The attribute <<voidable>> verticalExtent:EX_VerticalExtent [0..*] {ISO 19115:2006 Metadata (Corrigendum)} describes the vertical extent within which the results of all members of an observation collection are available.

3.9.1.2 Associations

3.9.1.2.1 **member**

The property *member:MO_Observation* [1..*] {MOLES 3.4} describes the observation(s) included in the observation collection. The type of this property is defined in section 3.1.1. An instance of *MO_Observation* may be member of more than one observation collections.

3.9.1.2.2 **subCollection**

An observation collection may be a member of a complex of parent and child-projects. If so, the property *subCollection:MO_ObservationCollection* [0..*] {MOLES3.4} describes a sub-collection of the parent collection.

3.9.1.2.3 **isoMetadata**

The property *isoMetadata:MD_Metadata* [0..*] {ISO 19115:2006 Metadata(Corrigendum)} provides descriptive ISO conformant metadata to the observation collection.

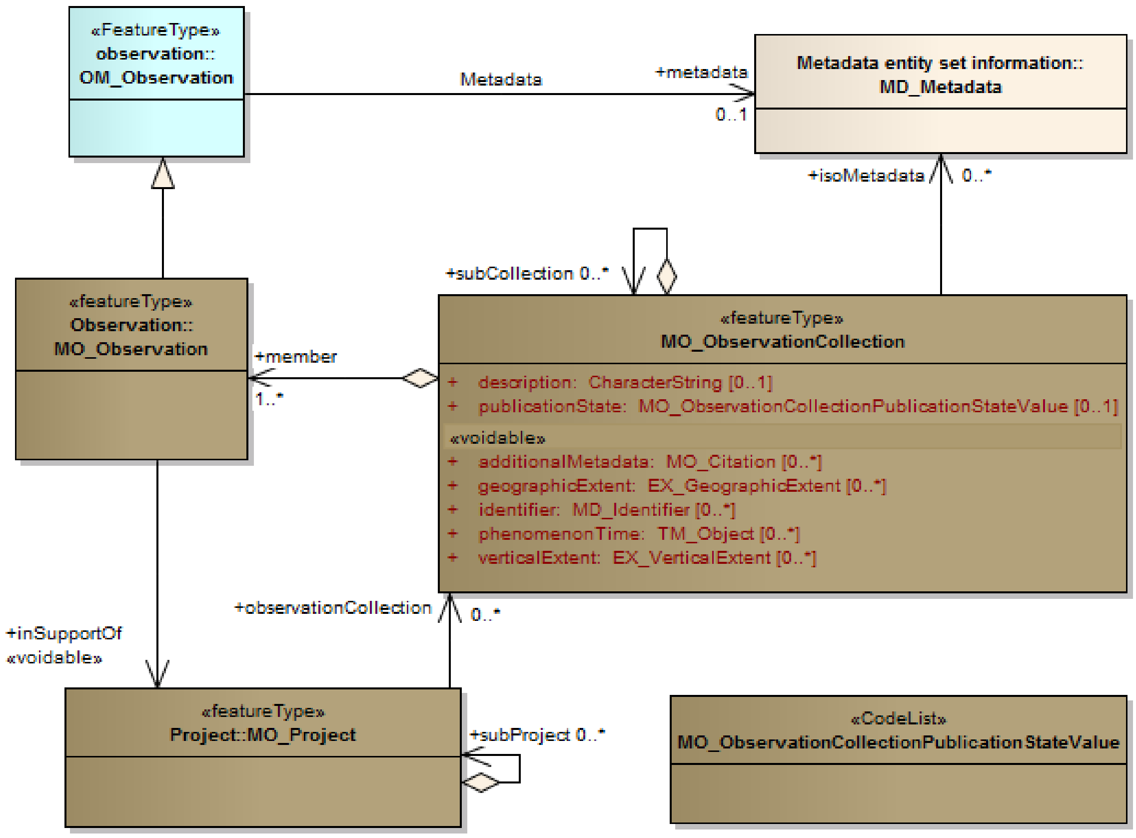

Figure 23 - ObjervationCollection model.

3.10 Utilities Package

ISO 19115:2006 in the package *Citation and responsible party information* provides a standardised method for citing a resource, as well as information about the party responsible for a resource. However, the model of ISO 19115 does not allow multiple parties for one role and as a result a new instance referring the same responsible party needs to be created every time a party has a new role. In addition, the citation does not include online reference to the cited resource.

The model provided in this package cure these problems by introducing the new class *MO_Citation* (see Figure 25) and the modelling shown in Figure 26 regarding the responsible party information.

NOTE The Draft ISO 19115-1 which has been prepared to cancel and replace ISO 19115:2003 is in line with the MOLES3.4 approach regarding *Citation and responsible party information*. Therefore, the MOLES 3.4 classes introduced in this package will be replaced with their counterparts in the Draft ISO 19115-1 when this draft is published as an ISO Standard.

The dependencies of this package to other schemas are shown in Figure 24.

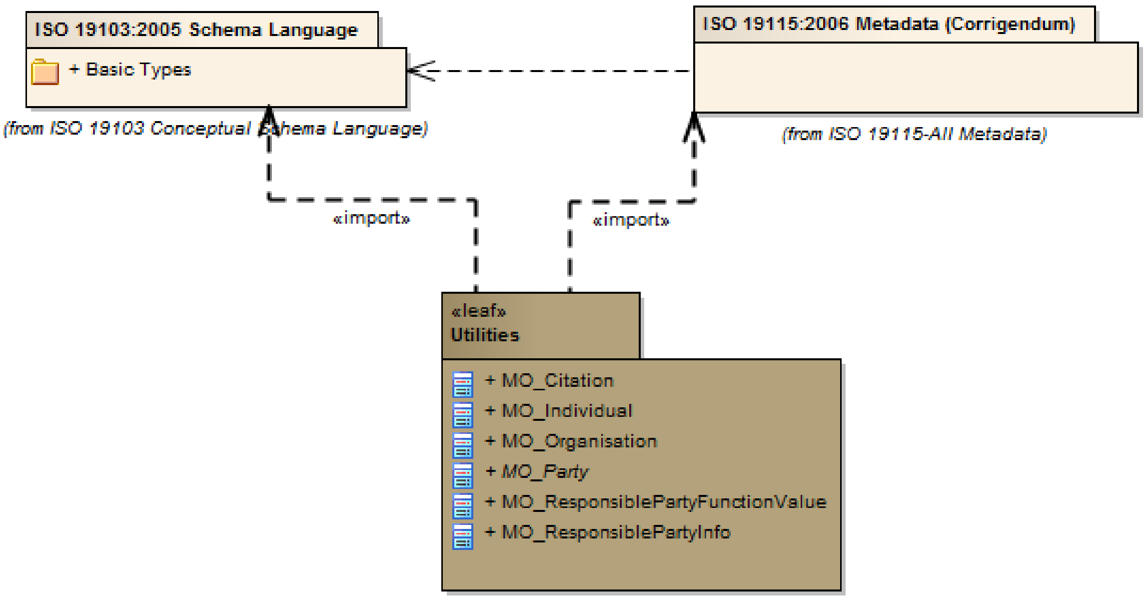

Figure 24 – Dependencies of the Utilities Package.

3.10.1 MO_Citation class

The class *MO_Citation*, which is a specialisation of the ISO 19115:2006 class *CI_Citation* (Figure 25), provides a standardized online/offline resource reference. In addition to the inherited attributes, it supports the attribute *onlineResource*.

3.10.1.1 Attributes

3.10.1.1.1 **onlineResource**

The attribute *onlineResource:CI_OnlineResource_Observation* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an online reference to the cited resource.

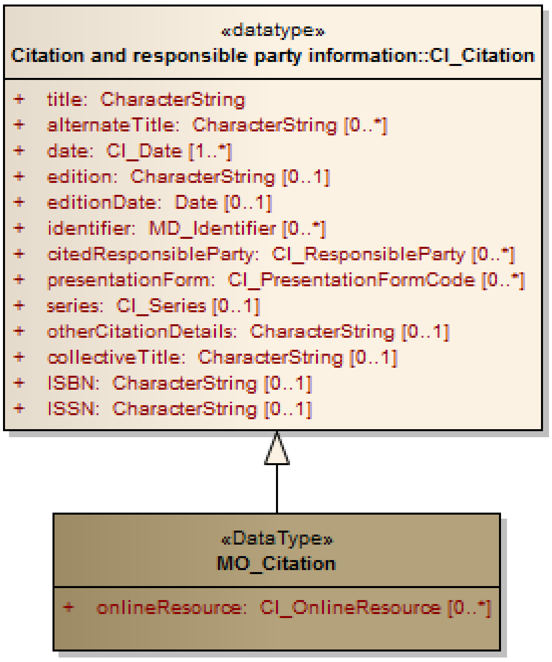

Figure 25 – MO_Citation class.

3.10.2 MO_ResponsiblePartyInfo Class

This class (see Figure 26) provides information about responsibility, identification of, and means of communication with, person(s) and organisations associated with a real world object.

3.10.2.1 Attributes

3.10.2.1.1 role

The attribute *role:MO_ResponsiblePartyFunctionValue* {MOLES 3.4} indicates the function performed by the responsible party. The values of the attribute should be taken from a controlled vocabulary. The code list *MO_ResponsiblePartyFunctionValue* is empty serving as a super-class for code lists which can specify the attribute values. A suggested code list is the ISO 19115:2006 Metadata (Corrigendum) code list *CI_RoleCode* (see section A.7).

3.10.2.1.2 extent

The attribute *extent:EX_Extent* [0..*] {ISO 19115:2006 Metadata(Corrigendum)} provides the temporal and/or spatial extent of the responsibility.

3.10.2.2 Associations

3.10.2.2.1 party

The attribute *party:MO_party* [1..*] {MOLES 3.4} indicates an individual or organisation having a responsibility regarding a real world object.

3.10.3 MO_Party Class

MO_Party is an abstract class that provides information about an individual or organisation which has a responsibility regarding a real world object. It has two specialisations: *MO_Individual* and *MO_Organisation*.

3.10.3.1 Attributes

3.10.3.1.1 name

The attribute *name:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides the name of the responsible party.

3.10.3.1.2 contactInfo

The attribute *contactInfo:CI_ContactInfo* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides the required information enabling contact with the responsible party.

3.10.4 MO_Individual Class

A concrete class representing an individual having a particular role associated with a real world object.

3.10.4.1 Attributes

3.10.4.1.1 positionName

The attribute *positionName:CharacterString* [0..1] {ISO 19103:2005 Schema Language} indicates the position of the individual within the named organisation.

3.10.5 MO_Organisation Class

This concrete class represents a body/organization having a particular role associated with a real world object.

3.10.5.1 Attributes

3.10.5.1.1 description

The attribute *description:CharacterString* [0..1] {ISO 19103:2005 Schema Language} provides a free text description of the organisation.

3.10.5.1.2 identifier

The attribute *identifier:MD_Identifier* [0..*] {ISO 19115:2006 Metadata (Corrigendum)} provides an external identifier of the organisation.

3.10.5.1.3 relatedParty

The attribute <<voidable>> *relatedParty:MO_ResponsiblePartyInfo* [1..*] {MOLES3.4} provides information regarding individuals or organisations related to the organisation.

The class *MO_ResponsiblePartyInfo* which defines the type of this attribute is described in section 3.10.2.

3.10.5.2 Association

3.10.5.2.1 **individual**

The property *individual:MO_Individual* [0..*]{MOLES3.4} indicates an individual in the named organisation.

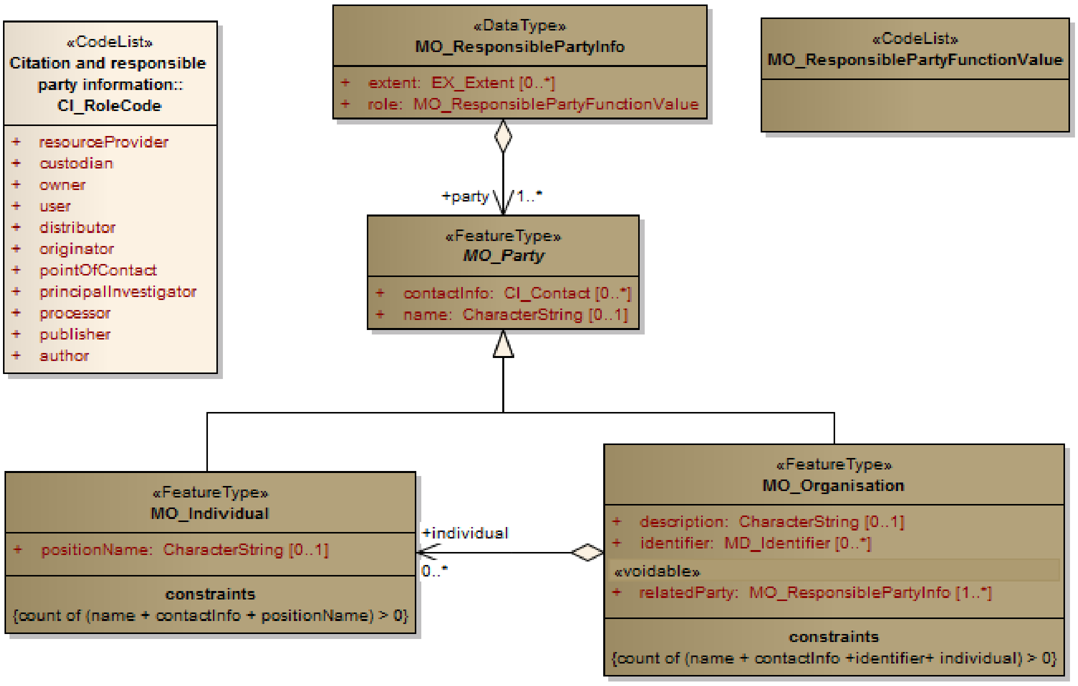

Figure 26 – Model for the responsible party information.

Annex A

Controlled Vocabulary

For each of the feature attributes taking predefined values (i.e. values from controlled vocabularies) an empty code list is assigned within MOLES3.4 application schema. The empty code list serves as a base class for selected code lists used to specify the attribute's values at the implementation stage according to the user requirements. MOLES3.4 does not impose the code list that shall be used; the MOLES3.4 application schema is independent of the used controlled vocabulary, which is managed externally.

The following section describes existing code lists that can be used (suggested) by this application schema. In the following tables the columns have the meanings:

- **Value**: The value (code) in lowerCamelCase (except for acronyms, which may be ALL UPPER CASE)
- **Name**: A human-readable name for the value.
- **Identifier**: Stable identifier for the value.

A.1 CEDA_PublicationStateValue

Code List: CEDA_PublicationStateValue Publication Status of metadata records.			
Value	**Name**	**Definition**	**Identifier**
citable	citable	For mature records that are unlikely to change and should be available via the discovery services.	
old	old	For redundant records.	
published	published	For records approaching maturity, which may still change, but should be available via the discovery services.	
working	working	For new/in progress records.	

A.2 MD_MaintenanceFrequencyCode

Code List : MD_MaintenanceFrequencyCode Frequency with which modifications and deletions are made to the data after it is first produced. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.18	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	Definition
continual	data is repeatedly and frequently updated
daily	data is updated each day
weekly	data is updated on a weekly basis
fortnightly	data is updated every two weeks
monthly	data is updated each month
quarterly	data is updated every three months
biannually	data is updated twice each year
annually	data is updated every year
asNeeded	data is updated as deemed necessary
irregular	data is updated in intervals that are uneven in duration
notPlanned	there are no plans to update the data
unknown	frequency of maintenance for the data is not known

A.3 MD_ProgressCode

Code List : MD_ProgressCode Status of the dataset or progress of a review. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.23	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	Definition
completed	production of the data has been completed
historicalArchive	data has been stored in an offline storage facility
obsolete	data is no longer relevant
onGoing	data is continually being updated
planned	fixed date has been established upon or by which the data will be created or updated
required	data needs to be generated or updated
underDevelopment	data is currently in the process of being created

A.4 CI_OnLineFunctionCode

Code List : CI_OnLineFunctionCode Function performed by the resource. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.3	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	Definition
download	online instructions for transferring data from one storage device or system to another
information	online information about the resource
offlineAccess	online instructions for requesting the resource from the provider
order	online order process for obtaining the resource
search	online search interface for seeking out information about the resource

A.5 MD_KeywordTypeCode

Code List : MD_KeywordTypeCode Methods used to group similar keywords. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.17	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	Definition
discipline	identifies a branch of instruction or specialized learning
place	identifies a location
stratum	identifies the layer(s) of any deposited substance
temporal	identifies a time period related to the dataset
theme	identifies a particular subject or topic

A.6 CI_PresenationFormCode

Code List : CI_PresentationFormCode Mode in which the data is represented. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.4	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	Definition
documentDigital	digital representation of a primarily textual item (can contain illustrations also)
documentHardcopy	representation of a primarily textual item (can contain illustrations also) on paper, photographic material, or other media
imageDigital	likeness of natural or man-made features, objects, and activities acquired through the sensing of visual or any other segment of the electromagnetic spectrum by sensors, such as thermal infrared, and high resolution radar and stored in digital format
imageHardcopy	likeness of natural or man-made features, objects, and activities acquired through the sensing of visual or any other segment of the electromagnetic spectrum by sensors, such as thermal infrared, and high resolution radar and reproduced on paper, photographic material, or other media for use directly by the human user
mapDigital	map represented in raster or vector form
mapHardcopy	map printed on paper, photographic material, or other media for use directly by the human user
modelDigital	multi-dimensional digital representation of a feature, process, etc.
modelHardcopy	3-dimensional, physical model
profileDigital	vertical cross-section in digital form
profileHardcopy	vertical cross-section printed on paper, etc
tableDigital	digital representation of facts or figures systematically displayed, especially in columns
tableHardcopy	representation of facts or figures systematically displayed, especially in columns, printed on paper, photographic material, or other media
videoDigital	digital video recording
videoHardcopy	video recording on film

A.7 CI_RoleCode

Code List : CI_RoleCode Function performed by the responsible party. **Identifier**			
governance	**Availability**	**Version**	**Format**
ISO 19115:2005	ISO 19115:2005 section B.5.5	ISO 19115:2005	Word Document

The values of this code list as quoted from ISO 19115 are:

Value	**Definition**
resourceProvider	party that supplies the resource
custodian	party that accepts accountability and responsibility for the data and ensures appropriate care and maintenance of the resource
owner	party that owns the resource
user	party who uses the resource
distributor	party who distributes the resource
originator	party who created the resource
pointOfContact	party who can be contacted for acquiring knowledge about or acquisition of the resource
principalInvestigator	key party responsible for gathering information and conducting research
processor	party who has processed the data in a manner such that the resource has been modified
publisher	party who published the resource
author	party who authored the resource

Annex B

Feature and UML notation

B.1 Concept of Feature

A fundamental unit in the development of an application schema is the feature concept which is defined formally in ISO 19101 and ISO 19109. Feature represents an abstraction of real-world phenomena; consequently a portion of the real world is modelled in terms of features together with their characteristics and relationships.

ISO 19109 distinguishes four aspects of defining features:

- the definitions or description used to group them into types;
- the attributes, i.e. characteristics of a feature, associated with each type;
- the relationships among the feature types and instances;
- the behaviour of the features, i.e. operation that every instance of a feature type may perform.

A feature may occur as a type or an instance. A feature type is a class of features having common characteristics whereas a feature instance is an individual of a given feature type having specified feature attribute values. Feature types are equivalent to classes and feature instances are equivalent to objects, in object-oriented modelling.

Figure B.1 describes the most abstract level of defining and structuring geographic data. The classification of real-world phenomena as features depends on their significance to a particular universe of discourse [*ISO 19109*].

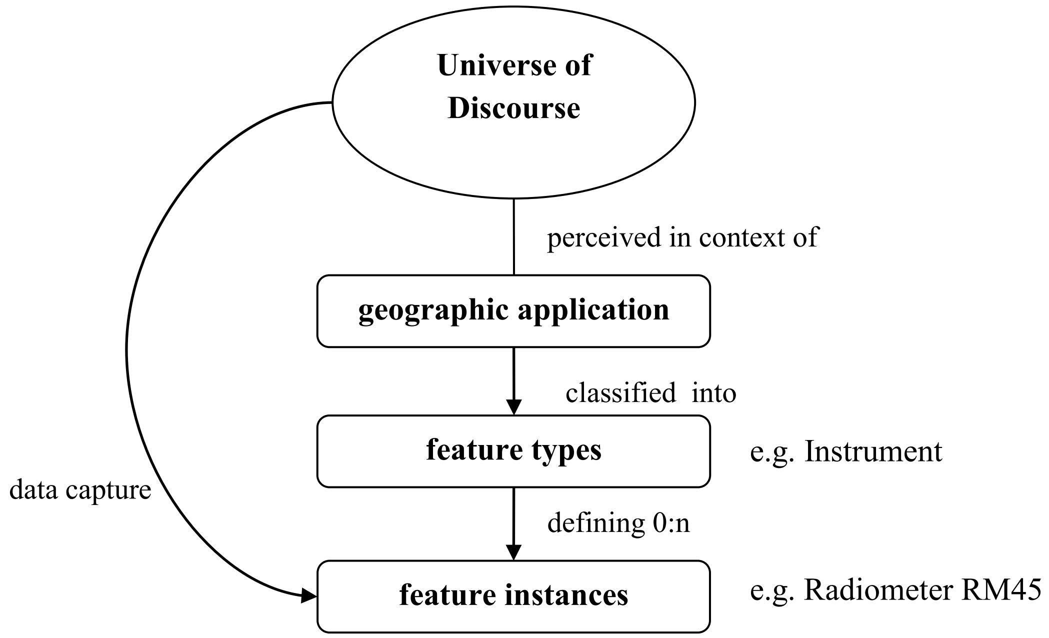

Figure B.1 - The process from universe of discourse to data

For example, within MOLES 3.4 it is defined a feature type representing the concept of the "acquisition". As every feature type, it has:

- a unique name within the application schema: *MO_Acquisition* and
- a definition describing it: *A process component which interacts with the feature of interest (e.g. an atmospheric column, a specimen etc) to provide a result.*

Figure B.2 depicts the UML class of this feature – the class has the stereotype "featureType" (see section B.5).

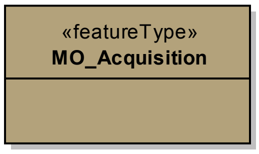

Figure B.2 – Formalization of the concept of acquisition in UML

A class representing a feature type could be abstract. This means that the class is so generic that it cannot be directly instantiated. It serves as a super class, in the inheritance hierarchy, for more specific classes. In UML notation an abstract class has its name in italics.

B.2 Feature Attributes

In addition to its name and definition a feature type may have attributes. A feature attribute has a name, a description, a data type of the attribute value, a value domain and cardinality associated with it.

The cardinality represents the number of instances of the attribute that may be associated with a single instance of a feature type. The range of the allowable cardinality is indicated by a multiplicity which can be one of: exactly-one (1), zero-or-one [0..1], one-or-more [1..*], zero-or-more [0..*] or an interval [n..m]. If no explicit multiplicity exists, the multiplicity is assumed to be 1.

Figure B.3 shows the attributes of the class *MO_Acquisition*. For example, the attribute with the name "*description*" (which is unique within the class) has the definition: *Textual description of the acquisition component*. The "*CharacterString*", a primitive type from ISO 19103, defines the type of the attribute, whereas the range of the allowable cardinality is specified by the multiplicity [0..1]. This means that an instance of *MO_Acquisition* may have zero or one instance of the attribute *description*.

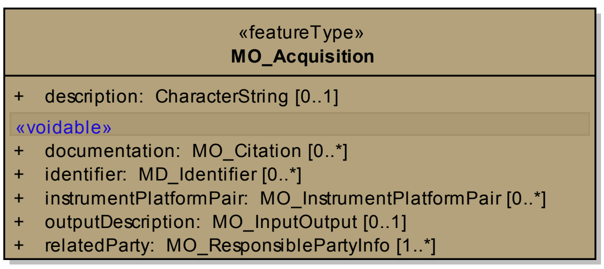

Figure B.3 – Formalization of the concept of acquisition and its attributes in UML

The rest of the attributes of *MO_Acqusition* class, regardless their multiplicities, are optional as they have the stereotype <<voidable>> (see section B.5).

A feature attribute for a feature instance has an attribute value taken from the value domain.

The attribute provides the interface to other ISO 19100 International Standards because it uses their schemas (e.g. spatial schema) to define its type.

B.3 Relationships

The relationships among features are classified as follows:

- generalization/specialization of feature types; and
- associations between features.

B.3.1 Generalization/specialization

A specialisation (generalisation) specifies subtypes (supertypes) of feature types. Each instance of the specific feature type (i.e. subtype) is also an indirect instance of the general feature type (supertype). A typical and powerful property of this kind of relationship is that a subtype inherits all the properties of its supertype. The generalization/specialization relationship occurs only between feature types.

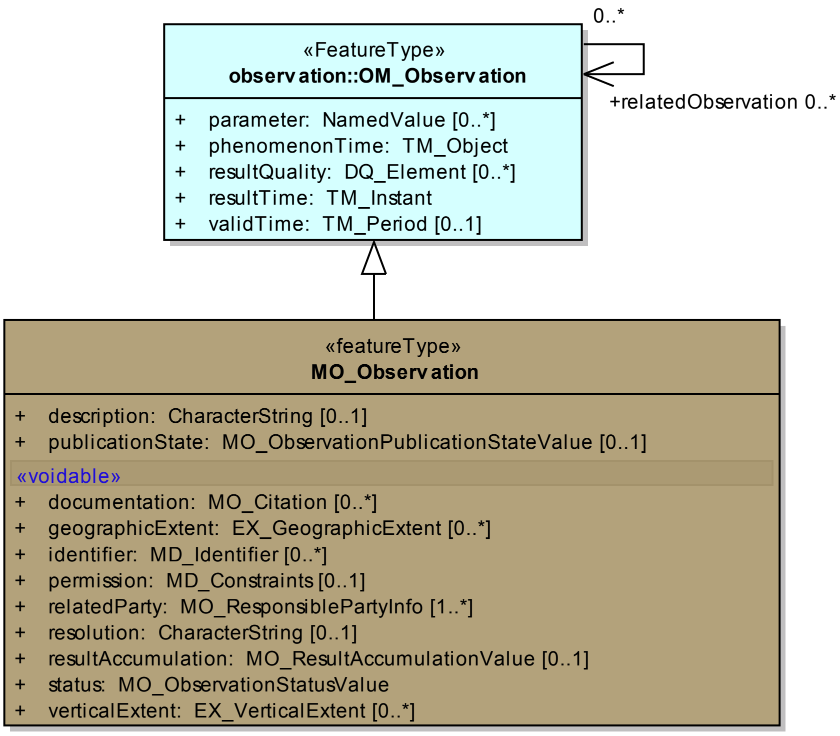

Figure B.4 – Example of specialisation relationship and its UML notation

As shown in Figure B.4 the MOLES3.4 class *MO_Observation* is a specialisation of the ISO 19156 *OM_Observation* class. All properties of *OM_Observation* are inherited by *MO_Observation*.

B.3.2 Associations and Role Names

An association is a semantic relationship that can occur between typed instances [*ISO 19103*]. All associations have cardinalities defined for both association ends which are specified by multiplicity statements. The cardinality of one end of the association represents the number of instances at this end which are connected with a single instance at the other end of the association.

An association is navigable in at least one direction having a role name at this end that is appropriate for the role of the target object in relation to the source object. Thus in a two-way navigable association two role names will be supplied.

Figure B.5 illustrates the one way navigable association between the classes *MO_Project* and *MO_ObservationCollection*, both defined within MOLES3.4. The role name *observationCollection* identifies the nature of the relation *MO_ObservationCollection* class has to class *MO_Project* through the association. The multiplicity [0..*] indicates that an instance of *MO_Project* class may be connected with zero or many instances of *MO_ObservationCollection* class.

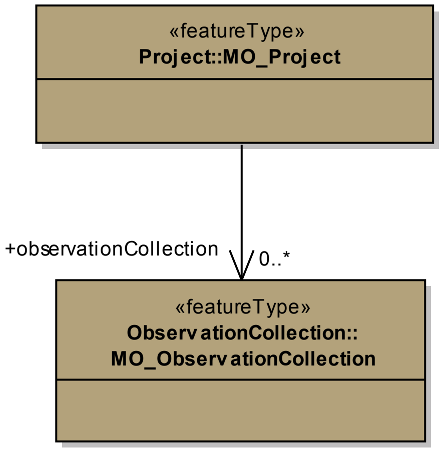

Figure B.5 – Example of association relationship and its UML notation

An association, according to General Feature Model [*ISO 19109*], is a feature type and therefore it may be having its own attributes, for example the association class *PreparationStep* in section 3.2.2.2.1. However, association names are optional and principally for documentation purposes.

B.3.4 Aggregation/Composition

Aggregation is a special form of association that models *"has-a"* relationships representing ownership relationships between typed instances. An aggregated instance may be owned by several other aggregating instances. For example, as shown in figure B.6, *Used Instrument* is an aggregation relationship between the *MO_Acquisition* and *MO_Instrument*. An instance of *MO_Acquisition* has 1 to many associated instances of *MO_Instruments*. The same instance of *MO_Instrument* may be owned by other aggregating *MO_Acquisition* instances.

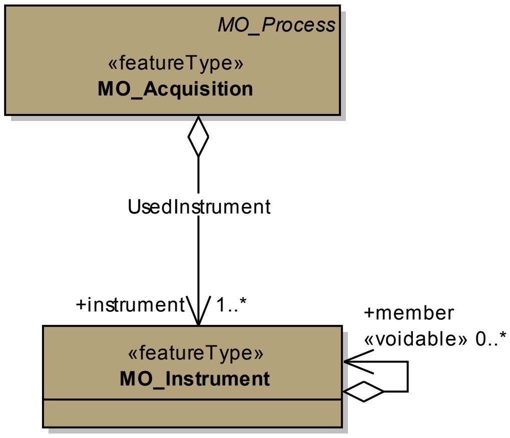

Figure B.6 – Example of aggregation relationship and its UML notation

If a typed instance is exclusively owned by an aggregating instance the relationship between them is referred as a composition. As shown in figure B.7 the result of an observation cannot be owned by other observations therefore their relationship is a composition.

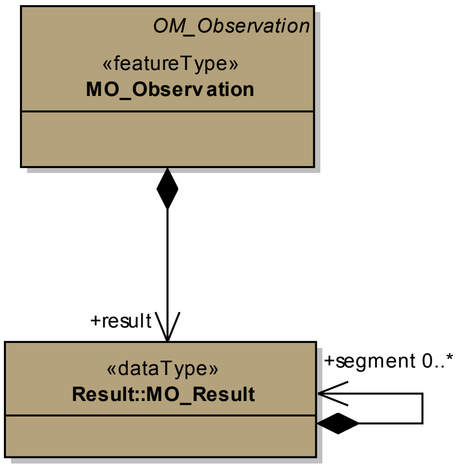

Figure B.7 – Example of composition relationship and its UML notation

B.4 Feature Properties

An association with role names can be viewed as similar to defining attributes for the two classes involved. For example, the role name *observationCollection* (see Figure B.5) serves as pseudo-attribute for the class *MO_Project* of type *MO_ObservationCollection*.

Therefore, both attributes and role names define the properties of feature types.

B.5 Stereotypes

A stereotype is a model element that is used to classify (or mark) other UML elements (e.g. classes, attributes) so that they in some respect behave as if they were instances of new virtual or pseudo metamodel (i.e. model that defines the language for expressing other models) classes whose form is based on existing base metamodel classes [*ISO 19103*].

In MOLES3.4 application schema the following stereotypes are used which are defined within ISO 19103.

- <<applicationSchema>> is a package representing an application schema as defined in ISO 19109.
- <<leaf>> is a package that contains definitions without any sub packages. A leaf package is not an application schema.

- <<featureType>> is a feature type as defined by the General Feature Model in ISO 19109.
- <<dataType>> is a set of properties that lack identity. A data type is a class with no operations whose primary purpose is to hold information.
- <<type>> is a set of abstract attributes and associations. Abstract means that their specification does not imply that they have to be concretely implemented as instance variables.
- <<Union>> is a type consisting of one and only one of several alternatives (listed as member attributes).

- <<enumeration>> is a fixed list of valid identifiers of named literal values. Attributes of an enumerated type may only take values from this list.
- <<CodeList>> is a flexible enumeration that uses string values through a binding of the Dictionary type key and return values as string types; e.g. Dictionary (String, String). If the elements of a list are completely known, an enumeration shall be used; if only the likely values of the elements are known, a code list shall be used [*ISO 19103*].

- <<voidable>> identifies an attribute or association role as optional, i.e. a value of 'void' is a valid value of the property.

The <<voidable>> stereotype does not give any information on whether or not a feature property exists in the real world.

> EXAMPLE The feature type *MO_Acquisition* (see Figure B.3) carries the attributes *documentation* and *relatedParty*. According to their multiplicities, [0..*] and [1..*] respectively, not all instances of *MO_Acquisition* have a *documentation*, in contrary to the attribute *relatedParty* which is always present. However, since both attributes have the stereotype <<voidable>> their values may not be captured, populated, and published but could be void.

Annex C

Types of feature attributes from the ISO 19100 series standards

This section provides a brief description of the types, obtained from the ISO 19100 series standards, for the feature attributes used in the MOLES3.4 application schema.

C.1 TM_Object, TM_Instant, TM_Period (ISO 19108)

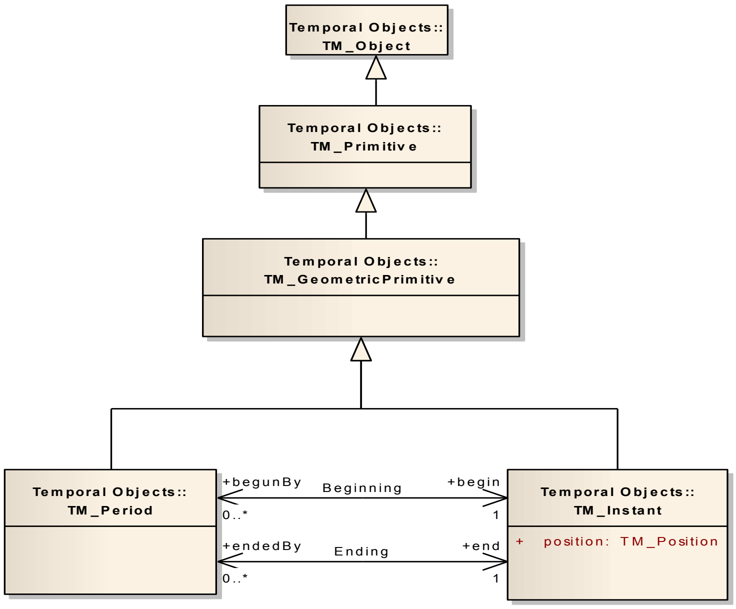

Figure C.1 – TM_Object, TM_Instant and TM_Period

C.2 MD_Identifier (ISO 19115)

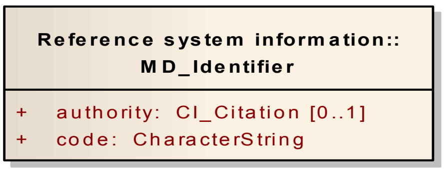

Figure C.2 – MD_Identifier

C.3 EX_GeographicExtent and EX_VerticalEttent (ISO 19115)

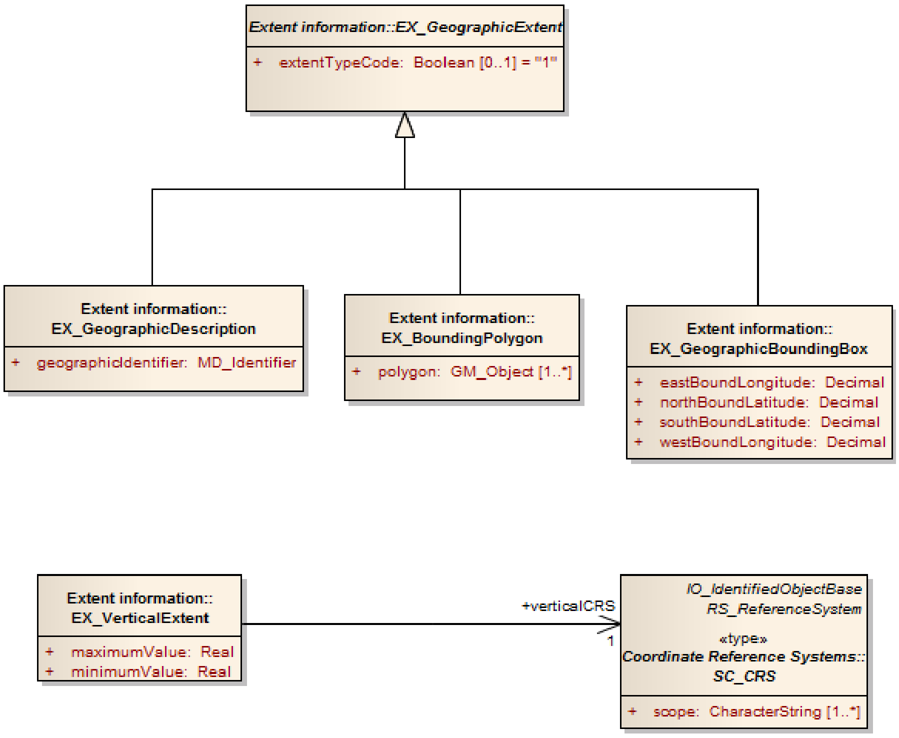

Figure C.3 – EX_GeographicExtent and EX_VerticalExtent

C.4 MD_Constraints (ISO 19115)

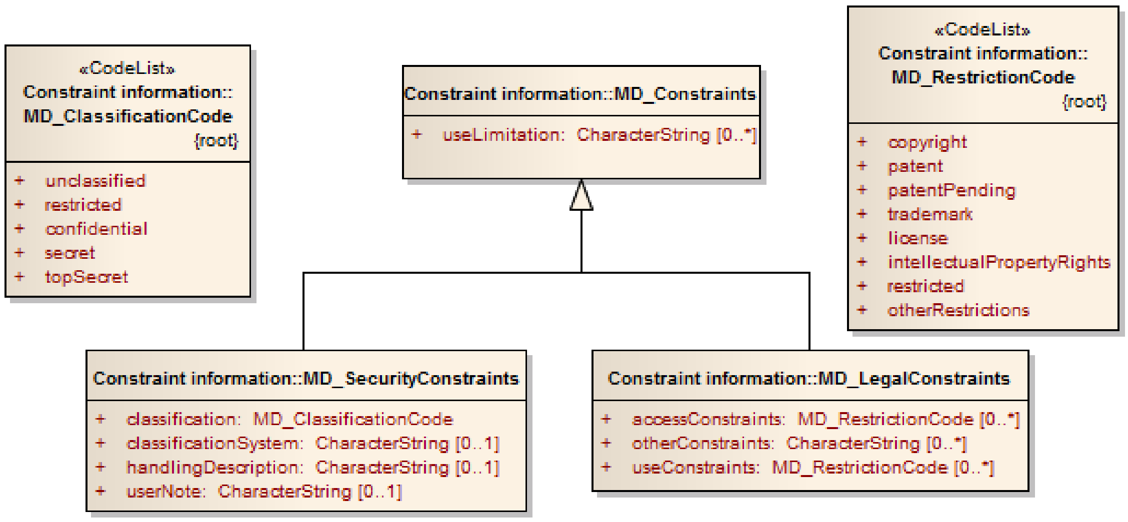

Figure C.4 – MD_Constraints

C.5 GM_Object (ISO 19107)

The abstract class *GM_Object* with some of its specialisations.

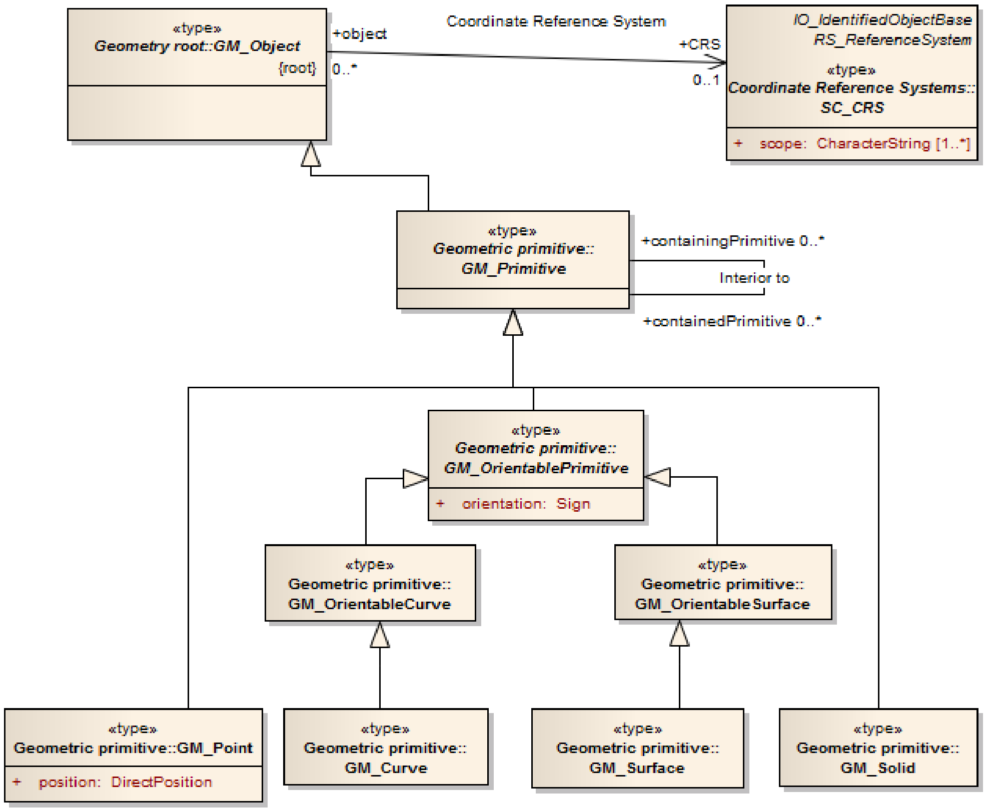

Figure C.5 – GM_Object

C.6 MD_Keywords (ISO 19115)

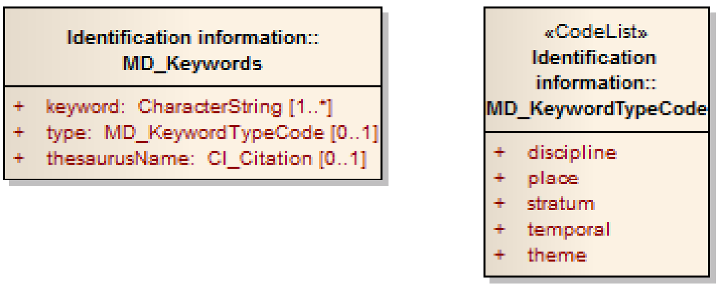

Figure C.6 – MD_Keywords

C.7 CI_Contact (ISO 19115)

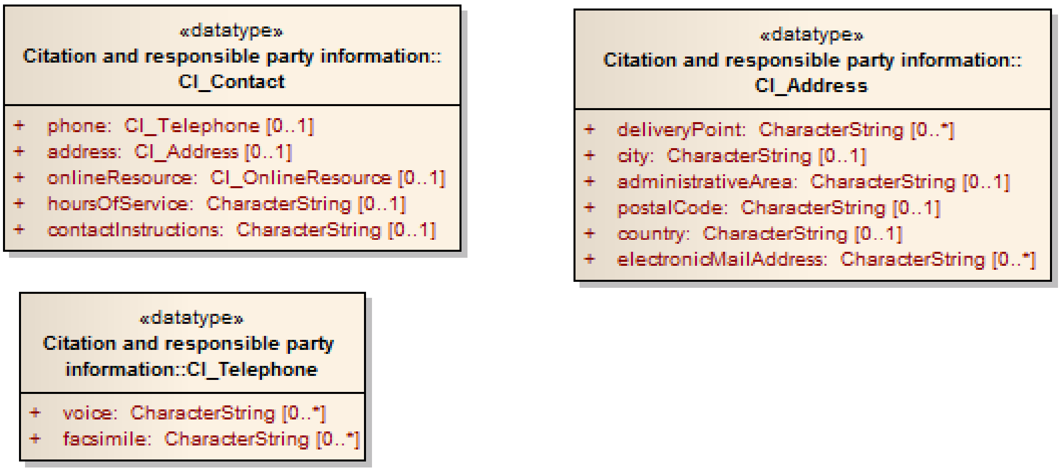

Figure C.7 – CI_Contact

Annex D

MOLES result and Actual values

The numerical artefact that the act of the observation produces, i.e. a number, term or other symbol is stored in a real world object. In this discussion, without loss of generality, we regard only files as real world objects that can be used to store numerical artefacts.

Even though the relationship between the observation and the observation result is by definition 1:1, the result can be stored in one or more files. Furthermore, the results of two or more observations are stored in the same file.

On the other hand *MO_Result* represents the resources for online access to the observation result. In other words, *MO_Result* refers to or points to the actual observation result. Therefore, see Figures E.1 and E.2, MOLES3.4 is insensitive to where and how the numerical artefacts are stored.

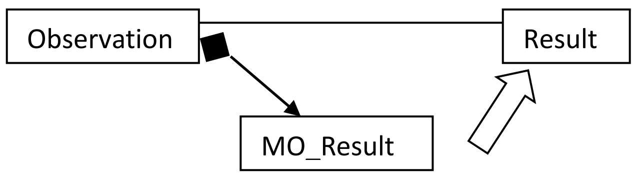

Figure E.1 – The observation result is stored in one or more files

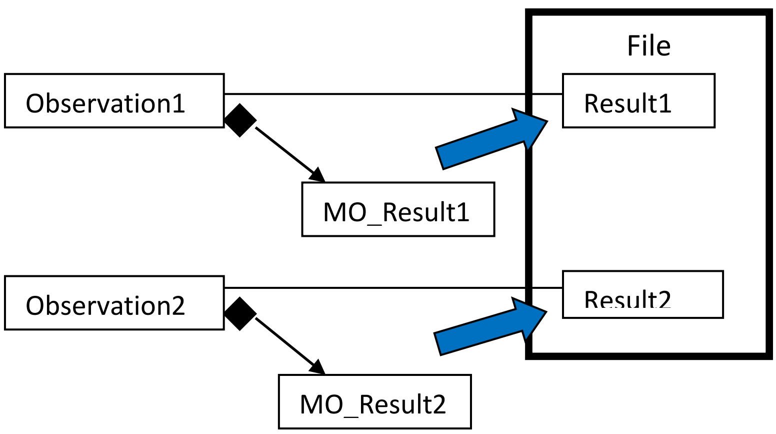

Figure E.2 – the results of two or more observations are stored in the same file.

References

[1] Fowler M., (1998) Analysis Patterns: reusable object models, Addison Wesley Longman, Menlo Park, CA.

[2] ISO 19101:2002, Geographic information – Reference Model

[3] ISO 19103:2005, Geographic information – Conceptual schema language

[4] ISO 19106:2006, Geographic information – Profiles

[5] ISO 19107:2003, Geographic information – Spatial Schema

[6] ISO 19108:2002, Geographic information – Temporal Schema

[7] ISO 19109:2005, Geographic information – Rules for application schema

[8] ISO 19115:2005, Geographic information – Metadata

[9] ISO 19115:2006, Geographic information – Metadata – Technical Corrigendum 1

[10] ISO 19915-2:2009, Geographic information – Metadata – Part2: Extensions for imagery and gridded data

[11] ISO DIS 19115-1:2012, Geographic information – Metadata – Part1: Fundamentals

[12] ISO 19119:2005, Geographic information – Services

[13] ISO 19136:2007, Geographic information – Geography Markup Language (GML)

[14] ISO DIS 19157:2011, Geographic information — Data quality

[15] ISO/IEC 19501:2005, Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2

[16] Khurana R (2010) Introduction to DataBase Systems, REARSON, ISBN 978-81-317-3192-5

[17] Lawrence, B.N., Lowry, R., Miller, P., Snaith, H. and Woolf, A. (2009) 'Information in environmental data grids.' Philosophical Transactions of the Royal Society A, 367, 1003-1014. doi:10.1098/rsta.2008.0237.

[18] Lawrence, B.N., Balaji, V., Bentley. P., Callaghan, S., DeLuca, C., Denvil, S, Devine G, Elkington M, Ford RW, Guilyardi E, Lautenschlager M, Morgan M, Moine P,Murphy S,Pascoe C, Ramthun H, Slavin P, Steenman-Clark L, Toussaint F, Treshansky A, and S. Valcke, (2012) 'Describing Earth System Simulations with the Metafor CIM', Geoscientific Model Development Discussion, 5, 1669-1689, (www.geosci- model-dev-discuss.net/5/1669/2012/)

[19] Latham, S.E., Cramer, R., Grant, M., Kershaw, P., Lawrence, B., Lowry, R., Lowe, D., O'Neill, K., Miller, P., Pascoe, S., Pritchard, M., Snaith, H. and Woolf, A. (2009) The NERC DataGrid services. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 367 (1890). pp. 1015-1019. ISSN 1364-503X doi: 10.1098/rsta.2008.0238

[20] Ventouras, S., Lawrence, B.N. and Cox, S. (2009), 'MOLES-v3 Information Model.' Proceedings Of the Data Management Workshop, Cologne, Germany.

[21] Ventouras, S., Lawrence, B.N., Woolf, A. and Cox, S. (2010), 'MOLES Information Model.' Proceedings of Geophysical Research Abstracts, 12, EGU2010-5080

[22] Ventouras, S. and Lawrence, B.N. (2011), 'Streamlining the Metadata Objects for Linking Environmental Sciences', Proceedings of Geophysical Research Abstracts, EGU2011-3368