ISO 19156:2020

ISO TC 211/WG 9

Secretariat: SIS

Geographic information — Observations and measurements

CD stage

Warning for WDs and CDs

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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

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

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 211, *Geographic information/Geomatics*, Working Group 9, *Information management*, in collaboration with the Open Geospatial Consortium, Inc. (OGC).

This second edition cancels and replaces the first edition (ISO 19156:2011), which has been technically revised. The UML model as well as the requirements/conformance class structure has been completely redesigned to address the contemporary modelling and observation data provision use cases. The fundamental Observation model has remained largely the same as in the previous version, with carefully designed improvements and clarifications for the intended use. Special care has been taken to ensure smooth migration from the ISO 19156:2011 to this version.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

This International Standard arises from work originally undertaken through the Open Geospatial Consortium's Sensor Web Enablement (SWE) activity. A set of interfaces and protocols was standardized, through which applications and services are able to access sensors of all types, and observations generated by them, over the Web.

A new generation of geospatial standards is now emerging, based on general Web standards, architecture, and current practice, as described in https://www.w3.org/TR/sdw-bp/. This includes several new standards for describing and publishing sensors and observations, such as OGC SensorThings API and W3C/OGC Semantic Sensor Network Ontology. This new version of the Observations and Measurements Standard is informed by these recent developments and is aimed at enabling the publication of observation data as part of the Web of data, while also supporting other means of data exchange.

The content presented here derives from the previous version published by Open Geospatial Consortium as OGC 10-004r3, OGC Abstract Specification Geographic information — Observations and measurements (ISO 19156:2011). A technical note describing the changes from the earlier version is available from the Open Geospatial Consortium (see http://www.opengeospatial.org/standards/om).

Geographic information — Observations and measurements

1 Scope

This International Standard defines a conceptual schema for observations, for features involved in the observation process, and for features involved in sampling when making observations. These provide models for the exchange of information describing observation acts and their results, both within and between different scientific and technical communities.

Observations commonly involve sampling of an ultimate feature-of-interest. This International Standard defines a common set of sample types according to their spatial, material (for ex-situ observations) or statistical nature. The schema includes relationships between sample features (sub-sampling, derived samples).

This International Standard concerns only externally visible interfaces and places no restriction on the underlying implementations other than what is needed to satisfy the interface specifications in the actual situation.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 19101-1:2014, *Geographic information — Reference model — Part 1: Fundamentals*

ISO/TS 19101-2:2008, *Geographic information — Reference model — Part 2: Imagery*

ISO 19103:2015, *Geographic information — Conceptual schema language*

ISO 19107:2019, *Geographic information — Spatial schema*

ISO 19108:2002, *Geographic information — Temporal schema*

ISO 19109:2015, *Geographic information — Rules for application schema*

ISO 19111:2019, *Geographic information — Referencing by coordinates*

ISO 19115-1:2014, *Geographic information — Metadata — Part 1: Fundamentals*

ISO 19123-1:20xx, *Geographic information — Schema for coverage geometry and functions — Part 1: Fundamentals*[1]

ISO 19123-2:2018, *Geographic information — Schema for coverage geometry and functions — Part 2: Coverage implementation schema*

ISO 19136-1:2020, *Geographic information — Geography Markup Language (GML) — Part 1: Fundamentals*

[1] To be published.

ISO 19136-2:2015, *Geographic information — Geography Markup Language (GML) — Part 2: Extended schemas and encoding rules*

ISO 19143:2010 *Geographic information — Filter encoding*

ISO 19157:2013, *Geographic information — Data quality*

ISO/IEC 19501:2005, *Information technology — Open Distributed Processing — Unified Modeling Language (UML)* Version 1.4.2

ISO Directives Part 2; available at ISO/IEC Directives, Part 2: Rules for the structure and drafting of International Standards

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
application schema
conceptual schema for data required by one or more applications

[ISO 19101-1:2014, definition 4.1.2]

3.2
coverage
feature that acts as a function to return values from its range for any direct position within its domain

[ISO 19123-1:20XX, definition 4.1.9]

3.3
data type
specification of a value domain with operations allowed on values in this domain

EXAMPLE Integer, Real, Boolean, String, Date.

Note 1 to entry: Data types include primitive predefined types and user-definable types.

[ISO 19103:2015, definition 4.14]

3.4
domain
well-defined set

Note 1 to entry: All elements within a domain (set) are of a given type.

[ISO 19109:2015]

3.5
domain feature
feature of a type defined within a particular application domain

Note 1 to entry: This may be contrasted with observation and sample features, which are features of types defined for cross-domain purposes.

3.6
ex-situ
referring to the study, maintenance or conservation of a specimen or population away from its natural surroundings ("off-site")

Note 1 to entry: Opposite of in-situ.

3.7
feature
abstraction of real-world phenomena

Note 1 to entry: A feature may occur as a type or an instance. In this International Standard, feature instance is meant unless otherwise specified.

[ISO 19101-1:2014, definition 4.1.11]

3.8
feature type
class of features having common characteristics

3.9
measure
value described using a numeric amount with a scale or using a scalar reference system

[ISO 19136:2020, definition 3.1.41]

3.10
measurement
set of operations having the object of determining the value of a quantity

[ISO/TS 19101-2:2018, definition 3.21]

3.11
measurand
quantity intended to be measured

[VIM3: International vocabulary of metrology – Basic and general concepts and associated terms : BIPM/ISO 2012, definition 2.3]

3.12
property
facet or attribute of an object referenced by a name

EXAMPLE Abby's car has the colour red, where "colour red" is a property of the car.

[ISO 19143:2010, definition 4.21]

3.13
property type
characteristic of a feature type

Note 1 to entry: The value for an instance of an observable property type can be estimated through an act of observation.

Note 2 to entry: In chemistry-related applications, the term "determinand" or "analyte" is often used.

Note 3 to entry: Adapted from ISO 19109:2005.

EXAMPLE Cars (a feature type) all have a characteristic colour, where "colour" is a property type.

3.14
range
set of feature attribute values associated by a function, the coverage, with the elements of the domain of a coverage

Note 1 to entry: This is consistent with the more generic definition of range in 19107 and 19136.

[ISO 19123-1:20XX, definition 4.1.44]

3.15
value
element of a type domain

Note 1 to entry: A value considers a possible state of an object within a class or type (domain).

Note 2 to entry: A data value is an instance of a datatype, a value without identity.

Note 3 to entry: A value can use one of a variety of scales including nominal, ordinal, ratio and interval, spatial and temporal. Primitive datatypes can be combined to form aggregate datatypes with aggregate values, including vectors, tensors and images.

[ISO/IEC 19501:2005]

3.16
requirement
expression, in the content of a document that conveys objectively verifiable criteria to be fulfilled and from which no deviation is permitted if conformance with the document is to be claimed

Note 1 to entry: Requirements are expressed using the verbal forms SHALL or SHALL NO. Equivalent phrases or expressions for use in certain cases are proposed by ISO.

[ISO/IEC Directives, Part 2 “Rules for the structure and drafting of International Standards”: 2018, definition 3.3.3]

3.17
recommendation
expression, in the content of a document, that conveys a suggested possible choice or course of action deemed to be particularly suitable without necessarily mentioning or excluding others

Note 1 to entry: Recommendations are expressed using the verbal forms SHOULD or SHOULD NOT. Equivalent phrases or expressions for use in certain cases are proposed by ISO.

Note 2 to entry: In the negative form, a recommendation is the expression that a suggested possible choice or course of action is not preferred but it is not prohibited.

[ISO/IEC Directives, Part 2 “Rules for the structure and drafting of International Standards”: 2018, definition 3.3.4]

4 Conformance

4.1 Overview

Clauses 7 to 13 of this International Standard use the Unified Modeling Language (UML) to present conceptual schemas for describing Observations. These schemas define conceptual classes that

a) may be considered to comprise a cross-domain application schema, or

b) may be used in application schemas, profiles and implementation specifications.

This flexibility is controlled by a set of UML types that can be implemented in a variety of manners. Use of alternative names that are more familiar in a particular application is acceptable, provided that there is a one-to-one mapping to classes and properties in this International Standard.

The UML model in this International Standard defines conceptual classes; various software systems define implementation classes or data structures. All of these reference the same information content. The same name may be used in implementations as in the model, so that types defined in the UML model may be used directly in application schemas.

Annex A defines a set of conformance tests that will support applications whose requirements range from the minimum necessary to define data structures to full object implementation.

4.2 Conformance classes related to models including Observations and Measurements

The conformance rules for Models in general are described in ISO 19109:2015. Application Schemas also claiming conformance to this International Standard shall also conform to the rules specified in Clauses 7 to 13 and pass all relevant test cases of the Abstract Test Suite in Annex A.

Depending on the characteristics of the implementing model application, schema or profile, one or more of the declared conformance classes can be chosen for fine-grained Observations and measurements conformance. Table 1, Table 2, Table 3, Table 4, Table 5 and Table 6 list all of these the classes by package, their relative identifiers and the corresponding subclauses of the Abstract Test Suite. The full URIs of the conformance classes is formed by prefixing the relative URI path as described in Clause 5.7.

Table 1 — Conceptual Observation schema conformance classes

Conformance class	Identifier	Annex A clause
Conceptual Observation schema package	/conf/cpt-obs	A.1.1
Conceptual Observation - Deployment	/conf/ cpt-obs /Deployment	A.1.2
Conceptual Observation - Host	/conf/obs-cpt/Host	A.1.3
Conceptual Observation - ObservableProperty	/conf/obs-cpt/ObservableProperty	A.1.4
Conceptual Observation - Observation	/conf/obs-cpt/Observation	A.1.5
Conceptual Observation - Observer	/conf/obs-cpt/Observer	A.1.6

Conceptual Observation - ObservingProcedure	/conf/obs-cpt/ObservingProcedure	A.1.7
Conceptual Observation - Procedure	/conf/obs-cpt/Procedure	A.1.8

Table 2 — Abstract Observation core conformance classes

Conformance class	Identifier	Annex A clause
Abstract Observation core package	/conf/obs-core	A.2.1
Abstract Observation core - AbstractDeployment	/conf/obs-core/AbstractDeployment	A.2.2
Abstract Observation core - AbstractHost	/conf/obs-core/AbstractHost	A.2.3
Abstract Observation core - AbstractObservableProperty	/conf/obs-core/AbstractObservableProperty	A.2.4
Abstract Observation core - AbstractObservationObservableProperty	/conf/obs-core/AbstractObservation	A.2.5
Abstract Observation core - AbstractObservationCharacteristics	/conf/obs-core/AbstractObservationCharacteristics	A.2.6
Abstract Observation core - AbstractObserver	/conf/obs-core/AbstractObserver	A.2.7
Abstract Observation core - AbstractObservingProcedure	/conf/obs-core/AbstractObservingProcedure	A.2.8
Abstract Observation core - NamedValue	/conf/obs-core/NamedValue	A.2.9

Table 3 — Basic Observations conformance classes

Conformance class	Identifier	Annex A clause
Basic Observations package	/conf/obs-basic	A.3.1
Basic Observations - Deployment	/conf/obs-basic/Deployment	A.3.2
Basic Observations - GenericDomainFeature	/conf/obs-basic/GenericDomainFeature	A.3.3
Basic Observations - Host	/conf/obs-basic/Host	A.3.4
Basic Observations - ObservableProperty	/conf/obs-basic/ObservableProperty	A.3.5
Basic Observations - Observation	/conf/obs-basic/Observation	A.3.6
Basic Observations - ObservationCharacteristics	/conf/obs-basic/ObservationCharacteristics	A.3.7
Basic Observations - ObservationCollection	/conf/obs-basic/ObservationCollection	A.3.8
Basic Observations - Observer	/conf/obs-basic/Observer	A.3.9
Basic Observations - ObservingCapability	/conf/obs-basic/ObservingCapability	A.3.10
Basic Observations - ObservingProcedure	/conf/obs-basic/ObservingProcedure	A.3.11

Table 4 — Conceptual Sample schema conformance classes

Conformance class	Identifier	Annex A clause
Conceptual Sample schema package	/conf/sam-cpt	A.4.1

Conceptual Sample - PreparationProcedure	/conf/sam-cpt/PreparationProcedure	A.4.2
Conceptual Sample - PreparationStep	/conf/sam-cpt/PreparationStep	A.4.3
Conceptual Sample - Sample	/conf/sam-cpt/Sample	A.4.4
Conceptual Sample - Sampler	/conf/sam-cpt/Sampler	A.4.5
Conceptual Sample - Sampling	/conf/sam-cpt/Sampling	A.4.6
Conceptual Sample - SamplingProcedure	/conf/sam-cpt/SamplingProcedure	A.4.7

Table 5 — Abstract Sample core conformance classes

Conformance class	Identifier	Annex A clause
Abstract Sample core package	/conf/sam-core	A.5.1
Abstract Sample core - AbstractPreparationProcedure	/conf/sam-core/AbstractPreparationProcedure	A.5.2
Abstract Sample core - AbstractPreparationStepSample - PreparationStep	/conf/sam-core/AbstractPreparationStep	A.5.3
Abstract Sample core - AbstractSample	/conf/sam-core/AbstractSample	A.5.4
Abstract Sample core - AbstractSampler	/conf/sam-core/AbstractSampler	A.5.5
Abstract Sample core - AbstractSampling	/conf/sam-core/AbstractSampling	A.5.6
Abstract Sample core - AbstractSamplingProcedure	/conf/sam-core/AbstractSamplingProcedure	A.5.7

Table 6 — Basic Samples conformance classes

Conformance class	Identifier	Annex A clause
Basic Samples package	/conf/sam-basic	A.6.1
Basic Samples - MaterialSample	/conf/sam-basic/MaterialSample	A.6.2
Basic Samples - NamedLocation	/conf/sam-basic/NamedLocation	A.6.3
Basic Samples - PhysicalDimension	/conf/sam-basic/PhysicalDimension	A.6.4
Basic Samples - Sample	/conf/sam-basic/Sample	A.6.5
Basic Samples - SampleCollection	/conf/sam-basic/SampleCollection	A.6.6
Basic Samples - Sampler	/conf/sam-basic/Sampler	A.6.7
Basic Samples - Sampling	/conf/sam-basic/Sampling	A.6.8
Basic Samples - SpatialSample	/conf/sam-basic/SpatialSample	A.6.9
Basic Samples - StatisticalClassification	/conf/sam-basic/StatisticalClassification	A.6.10
Basic Samples - StatisticalSample	/conf/sam-basic/StatisticalSample	A.6.11

5 Document conventions

5.1 Abbreviated terms and acronyms

GFM	General Feature Model
GML	Geography Markup Language
INSPIRE	INfrastructure for SPatial InfoRmation in Europe
O&M	Observations and Measurements
OGC	Open Geospatial Consortium
SensorML	OGC Sensor Model Language
SOS	OGC Sensor Observation Service
STA	OGC SensorThings API
SWE	OGC Sensor Web Enablement
UML	Unified Modeling Language
XML	Extensible Markup Language
2-D	Two Dimensional
3-D	Three Dimensional

5.2 Schema language

The conceptual schema specified in this International Standard is in accordance with the Unified Modelling Language (UML) ISO/IEC 19501, following the guidance of ISO 19103:2015.

The UML in Abstract Core and Basic packages is conformant with the profile described in ISO 19136:2007, Annex E. Use of this restricted idiom supports direct transformation into a GML Application Schema. ISO 19136 introduces some additional stereotypes. In particular «FeatureType» implies that a class is an instance of the «metaclass» GF_FeatureType (ISO 19109), and therefore represents a feature type.

The prose explanation of the model uses the term “property” to refer to both class attributes and association roles. This is consistent with the General Feature Model described in ISO 19109. In the context of properties, the term “value” refers to either a literal (for attributes whose type is simple), or to an instance of the class providing the type of the attribute or target of the association. Within the explanation, the property names (property types) are sometimes used as natural language words where this assists in constructing a readable text.

5.3 Model element names

This International Standard specifies a model for observations using terminology that is based on current practice in a variety of scientific and technical disciplines. It is designed to apply across disciplines, so the best or "most neutral" term has been used in naming the classes, attributes and associations provided. The terminology does not, however, correspond precisely with any single discipline. As an aid to implementers, a mapping from the element names specified in this International Standard to common terminology in some application domains is provided in Annex B.

5.4 Requirements and recommendations

All requirements are **normative**, and each is presented with the following template:

Requirement /req/{pkg}/{classM}/{reqN}	[Normative statement]

where **/req/{pkg}/{classM}/{reqN}** identifies the requirement. The use of this layout convention allows the normative provisions of this standard to be easily located by implementers. In the description of the convention right after the **/req/{pkg}/** part has been removed to ease reading.

All defined classes, attributes and associations mentioned within requirements or recommendations are shown in **bold** correspond to references to the definition of the referenced element.

The following base has been used per package:

a) **/req/obs-cpt**: Conceptual Observation schema,

b) **/req/obs-core**: Abstract Observation Core,

c) **/req/obs-basic**: Basic Observations,

d) **/req/sam-cpt**: Conceptual Sample schema,

e) **/req/sam-core**: Abstract Sample core,

f) **/req/sam-basic**: Basic Samples

For naming of individual requirements pertaining to classes, the following syntax is used:

— **{Class Name}-sem**: The semantic definition of the concept, together with the naming of the Class.

For naming of individual requirements pertaining to attribute or associations, the following syntax is used:

i) **{Attribute/Association Name}-sem**: The semantic definition of the concept, together with the naming of the attribute or association role. Except for cases where concepts are mandatory within all packages, these statements are phrased to be cardinality neutral, e.g. they also apply to cardinality 0..*;

ii) **{Attribute/Association Name}-type**: Type information pertaining to the attribute or association when the type is constrained within one model package;

iii) **{Attribute/Association Name}-card**: Cardinality information pertaining to the attribute or association;

iiii) **{Attribute/Association Name}-con**: Additional constraints. As these sometimes pertain to multiple attributes or associations, this part of the name may become more complex.

All recommendations are **informative**, and each is presented with the following template:

Recommendation /rec/{pkg}/{classM}/{reqN}	[Informative statement]

where **/rec/{pkg}/{classM}/{reqN}** identifies the recommendation. The use of this layout convention allows the informative provisions of this standard to be easily located by implementers.

5.5 Requirements classes

Each statement (requirement or recommendation) in this standard is a member of a requirements class.

All requirement classes are normative.

Each requirements class is described in a discrete clause, and summarized using the following template:

Requirements class	**/req/{pkg}/{classM}**
Target type	[artefact or technology type]
Name	Name of the requirements class
Imports	/req/{pkg}/{classZ}
Requirement	/req/{pkg}/{classM}/{reqN}
Recommendation	/rec/{pkg}/{classM}/{recO}
Requirement	/req/{pkg}/{classM}/{reqP}
Requirement /Recommendation	[repeat as necessary]

All requirements in a class must be satisfied. Hence, the requirements class is the unit of re-use and dependency.

Dependency to another requirement class (and the requirements and recommendations defined in it) is done using the “Imports” keyword. All requirements in a dependency SHALL also be satisfied by a conforming implementation.

A requirements class may consist only of dependencies and introduce no new requirements.

5.6 Conformance classes

Conformance to this standard is possible at a number of levels, specified by conformance classes (Annex A). Each conformance class is summarized using the following template:

Conformance Class	**/conf/{pkg}/{classM}**
Requirements	[identifier for the requirements class]
Test purpose	[Reason for test]
Test method	[Method to determine if test fulfilled]
Test Type	[Type of test]

All tests in a class must be passed. Each conformance class tests conformance to a set of requirements packaged in a requirements class.

5.7 Identifiers

Each requirements class, requirement and recommendation are identified by a unique identifier. This allows cross-referencing of class membership, dependencies, and links from each conformance test to the requirements tested. Appended to a base URI that identifies the specification as a whole, it enables the construction of a complete URI for identification in an external context.

The entire Requirements and Conformance Structure, consisting of the individual requirements and definitions together with the information on how these are linked together for the creation of Requirements and Conformance classes, will be exposed in a machine actionable format (such as the one provided by the OGC Definitions Server).

The URI for each requirements class has the form:

[base URI][2]**/req/{pkg}/{classM}**

The URI for each requirement has the form:

[base URI]**/req/{pkg}/{classM}/{reqN}**

The URI for each recommendation has the form:

[base URI]**/rec/{pkg}/{classM}/{reqN}**

The URI for each conformance class has the form:

[base URI]**/conf/{pkg}/{classM}**

[2] [baseURI] will be replaced by the actual baseURI of the specification in the DIS version.

6 Packaging, requirements and dependencies

6.1 Requirement and conformance class structure

As O&M implementations often seamlessly integrate with existing data ecosystems, a very flexible requirements and conformance structure is defined. It enables users to selectively mix and match elements as required for their purposes from the O&M data model without the necessity of achieving compliance with the entire data model.

Such flexibility is becoming increasingly relevant with the shift to Linked Data practices, where different organizations maintain and expose only certain aspects of a larger distributed dataset.

EXAMPLE A provider may only serve information on Observable Properties or Monitoring Facilities, while relying on other partners to provide information on measurement procedures. These could claim compliance to those parts falling under their responsibility, while letting other data providers link to these resources.

For this purpose, a fine grained structure for requirements and recommendations, requirements classes and conformance classes has been defined. As far as possible, patterns from the OGC Modular Specification [24] have been taken into account. However, pertaining to the alignment between UML Packages and Conformance Classes, a relaxation of the requirement on one-to-one alignment between UML Package and Conformance Class has been proposed as follows:

— For each UML Package, both a Requirements Class as well as a Conformance Class have been defined;

— Additional Requirements Classes have been created for each Class appearing in the data model, Conformance Classes are added accordingly to enable grouping of the formers and support references to either a group or an individual Requirement Class depending on the need;

— Thematic Domains may create additional Requirements and Conformance Classes reflecting their domain profiles by reference to existing Requirements and Requirements Classes.

As mentioned, as data provision paradigms increasingly shift towards distributed and linked approaches, it becomes increasingly difficult to stipulate that all aspects of an information system conform explicitly to the same underlying standards. Simultaneously, as distributed data provision becomes increasingly ubiquitous, ever more communities are emerging dedicated to individual aspects of the wider data provision landscape.

One example of such external definition and hosting pertains to the provision of observable properties. In previous versions of the O&M Model, this concept was only included as a metaclass, with the assumption that a reference to an existing code list will be provided. Within the current O&M Model, the observable property has been upgraded to a featureType, as emerging requirements show the need for a more detailed model for this concept. Simultaneously, other communities such as the Research Data Alliance (RDA) are also working on observable property models. The same rationale can be applied to most concepts from the O&M Models.

In order to expose this flexibility beyond the package structure described above, a fine grained hierarchical requirements class structure has been created. A modular requirements class is provided for each concept at all three levels of the model. In addition, a further requirements class that imports all the modular classes provided for the individual concepts has been provided for each package.

6.1.1 Requirements class dependency graphs

The graphs in Figure 1 and Figure 2 have been auto-generated by parsing the requirements class tables contained in this document, and are provided here for the reader convenience.

Double-bordered boxes are requirements classes, simple boxes are individual requirements or recommendations. Arrows with solid lines show inclusion of requirements, dotted lines are dependencies showing where other requirement classes have been imported.

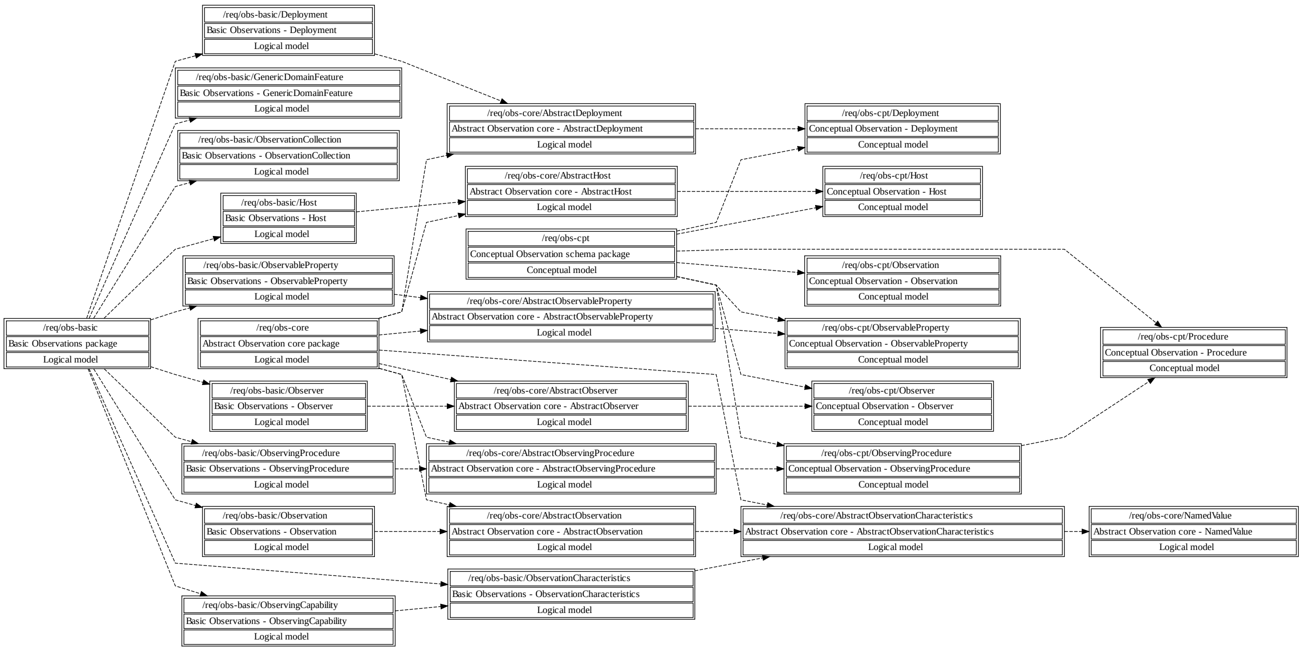

Figure 1 — Requirements class dependencies of the Observation packages.

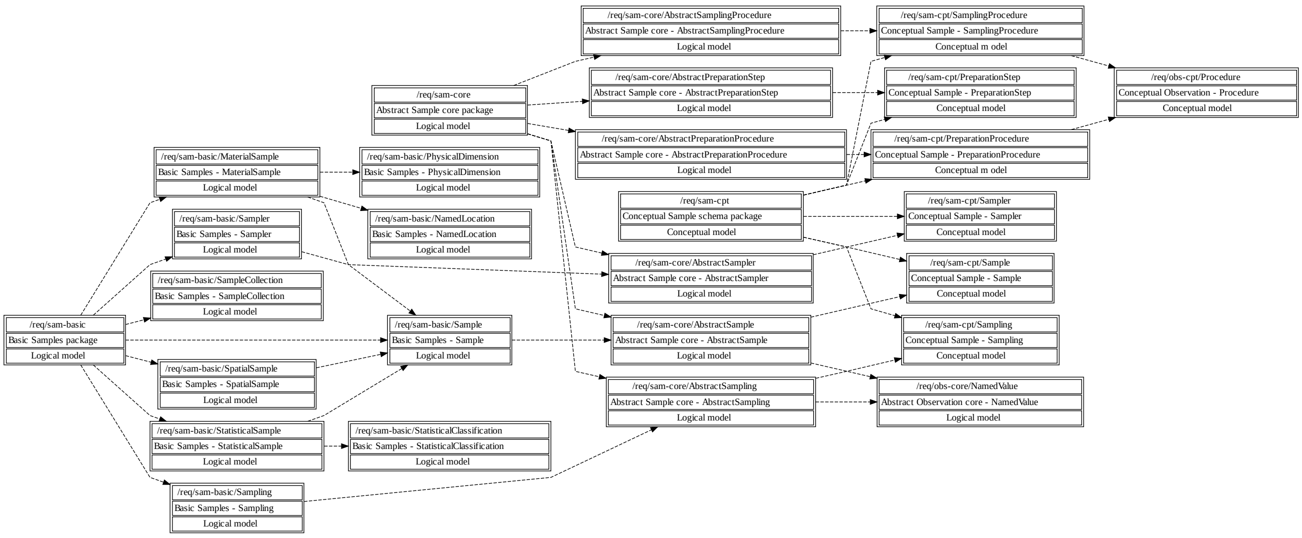

Figure 2 — Requirement class dependencies of the Sample packages.

6.2 UML package structure

O&M provides the relevant concepts for the structured description of observations, including the sampling structure often essential for true understanding of the nature of the observations being provided. As data provision mechanisms are transitioning towards highly distributed linked approaches, the model structure and packaging has been significantly abstracted. This approach allows implementers to explicitly select the concepts to be supported based on their requirements, while clearly stating to which requirements and Conformance Classes their implementation complies. Both the Observation and Sample sections of this model have been structured using the following layering of packages:

— **Conceptual**: Within the Conceptual Model Packages, only Interfaces have been provided. These models provide a very abstract view on the individual concepts they contain without reference to specific implementations. This approach allows for the inclusion of semantically aligned objects from external sources, that while not having been created under the Observations & Measurements Model do provide concepts sharing the same semantic meaning as the concepts from the Conceptual Models;

— **Abstract Core**: Within the Abstract Core Model Packages, only abstract featureTypes have been provided following the semantic structure of the Conceptual model (i.e: realizing the interfaces provided by the Conceptual). A consistent approach to metadata provision introduced. All associations from the abstract featureTypes reference the conceptual Interfaces for greater flexibility of implementation. The Abstract Core Model Packages are foreseen for the creation of domain models providing an Abstract Core ready for Extension;

— **Basic**: Within the Basic Packages, simple concrete featureTypes (specializing the abstract ones from the Abstract Core model) have been defined with some basic utility attributes added for rapid out-of-the-box deployment. A few additional concepts pertaining to collections and potential observations are introduced at this level.

6.2.1 UML package dependencies

Some model elements used in the schema are defined in other International Standards. The Table 7 lists the dependencies between the UML packages defined in this International Standard and other International Standards, and the Figure 3 show the dependencies of the entire Observations and measurements UML model package to the other International Standards in a graphical form.

Table 7 — UML package level dependencies

Dependency from package	to package	in an International Standard	Notes
Conceptual Observation schema	Any type	ISO 19103:2015 (Edition 1)	Any
Conceptual Observation schema	Temporal Objects	ISO 19108:2002 (Edition 1)	TM_Object
Conceptual Observation schema	Name types	ISO 19103:2015 (Edition 1)	GenericName
Abstract Observation core	Conceptual Observation schema	This International Standard	TM_Instant, TM_Period via the Temporal

Dependency from package	to package	in an International Standard	Notes
			Objects dependency
Abstract Observation core	General Feature Model	ISO 19109:2015 (Edition 2)	Feature concepts
Abstract Observation core	Text	ISO 19103:2015 (Edition 1)	CharacterString
Basic Observations	Abstract Observation core	This International Standard	
Basic Observations	Web environment	ISO 19103:2015 (Edition 1)	URI
Basic Observations	Geometry	ISO 19107:2019 (Edition 2)	Geometry
Conceptual Sample schema	Any type	ISO 19103:2015 (Edition 1)	Any
Conceptual Sample schema	Temporal Objects	ISO 19108:2002 (Edition 1)	TM_Object
Conceptual Sample schema	Name types	ISO 19103:2015 (Edition 1)	GenericName
Conceptual Sample schema	Conceptual Observation schema	This International Standard	Observation, Procedure
Abstract Sample core	Conceptual Sample schema	This International Standard	
Abstract Sample core	General Feature Model	ISO 19109:2015 (Edition 2)	Feature concepts
Abstract Sample core	Geometry	ISO 19107:2019 (Edition 2)	Geometry
Abstract Sample core	Text	ISO 19103:2015 (Edition 1)	CharacterString
Abstract Sample core	Abstract Observation core	This International Standard	NamedValue
Basic Samples	Abstract Sample core	This International Standard	
Basic Samples	Web environment	ISO 19103:2015 (Edition 1)	URI
Basic Samples	Measure types	ISO 19103:2015 (Edition 1)	Measure

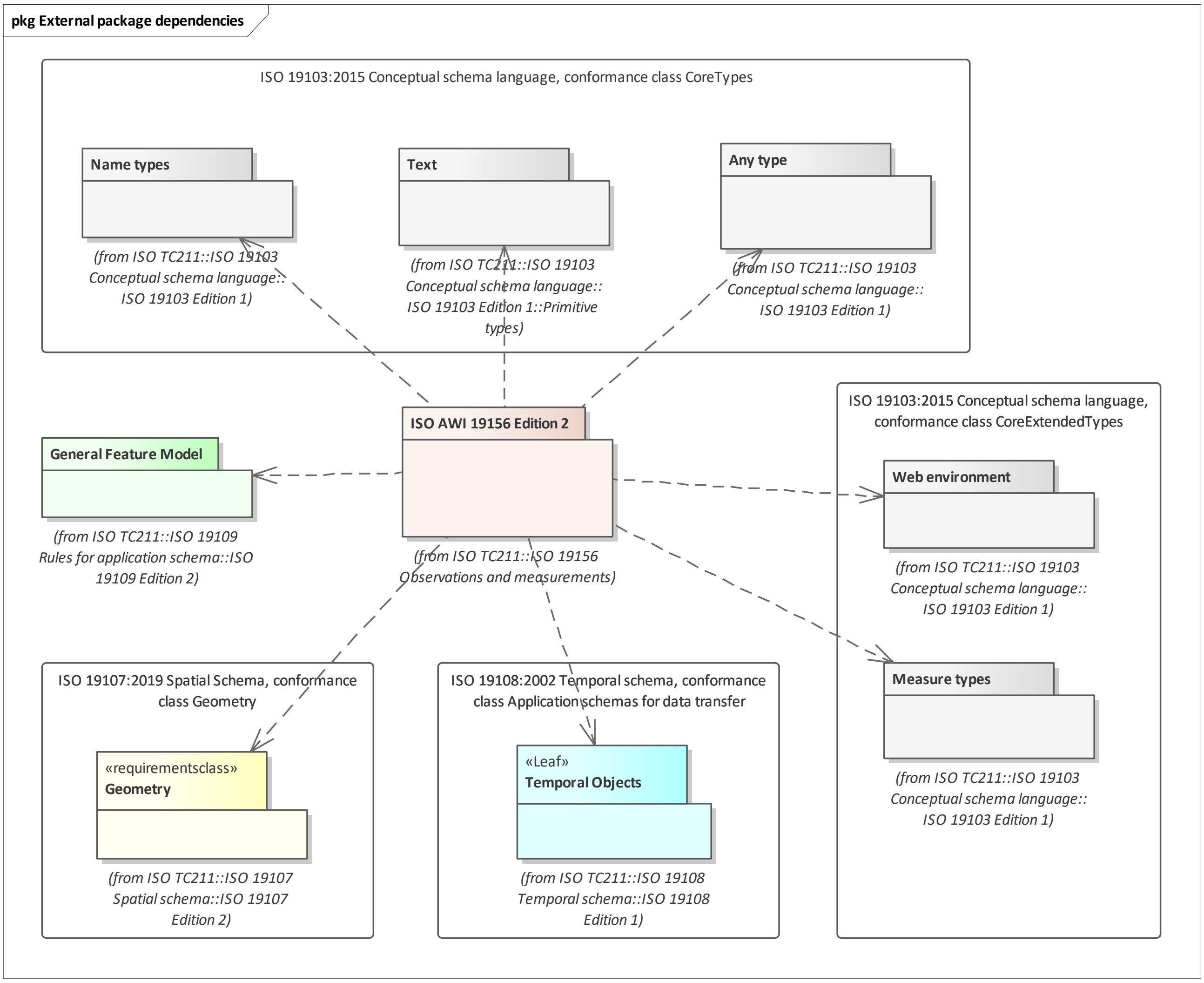

Figure 3 — External UML package dependencies

6.3 Note on the use of Any

The UML models defined in this International Standard make extensive use of the Any interface defined in ISO 19103:2015 Any type package. The realized Any values of the associations with role names **proximateFeatureOfInterest**, **ultimateFeatureOfInterest**, **result**, **metadata**, **featureOfInterest** and **sampledFeature** MAY be of any type or a reference to an digital representation of an appropriate concept. In the case they are of feature type, the values are not owned by the instances of referring classes, they may have an independent life span from the referring classes, and they may be associated with more than one instance of referring classes.

NOTE any type can be owl:Thing, featureType, dataType

EXAMPLES:

— Reference to SWEET Ontology: http://sweetontology.net/realmAtmoBoundaryLayer#planetaryboundarylayer

— Reference to SensorThings deployment: https://lubw-frost.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Locations(269)

— Reference to 19115 Metadata: https://inspire-geoportal.ec.europa.eu/resources/INSPIRE-61494ff5-6fad-11e8-b649-52540023a883_20200903-065202/services/1/PullResults/601-650/31.iso19139.xml

— Reference to an instance of Borehole : https://data.geoscience.fr/id/borehole/BSS001REWW

— Reference to an hydro station : https://iddata.eaufrance.fr/id/HydroStation/Y251002001

— Reference to a river segment : https://iddata.eaufrance.fr/id/WatercourseLinkSequence/A0080300

— An (embedded) Boolean value as Result

— An (embedded) SWE DataRecord

— Elevation Coverage from an external WCS as an observation Result: https://inspire.rasdaman.org/rasdaman/ows?service=WCS&version=2.0.1&request=GetCoverage&coverageId=INSPIRE_EL&subset=E(494500,496000)&subset=N(4654300,4655000)&format=image/jpeg

— O&M MaterialSample -> Reference to a rock sample : https://www.geodata.rocks/Samples/SD-5054_1_A_564_7WR_20-40

7 Fundamental characteristics of observations and samples (informative)

7.1 Observation schema

7.1.1 Property evaluation

Properties of a feature fall into two basic categories:

a) Value (e.g. name, price, legal boundary) assigned by some authority. These are exact.

b) Value (e.g. height, classification, colour) determined by application of an observation procedure.

These are estimates, with a finite error associated with the value.

The observation error typically has a systematic component, which is similar for all estimates made using the same procedure, and a random component, associated with the particular application instance of the observation procedure. If potential errors in a property value are important in the context of a data analysis or processing application, then the details of the act of observation which provided the estimate of the value are required.

7.1.2 Observation

An observation is an act associated with a discrete time instant or period through which a number, term or other symbol is assigned to a phenomenon [2]. It involves application of a specified procedure, such as a sensor, instrument, algorithm or process chain. The procedure may be applied in-situ, remotely, or ex-situ with respect to the sampling location. The result of an observation is an estimate of the value of a property of some feature. Use of a common model allows observation data using different procedures to be combined unambiguously.

In conventional measurement theory (e.g. [1][5][10][11][20]) the term “measurement” is used. However, a distinction between measurement and category-observation has been adopted in more recent work

[2][12][21] so the term "observation" is used here for the general concept. "Measurement" may be reserved for cases where the result is a numerical quantity.

The observation itself is also a feature, since it has properties and identity.

Observation details are important for data discovery and for data quality estimation.

The observation could be considered to carry "property-level" instance metadata, which complements the dataset-level and feature-level metadata that have been conventionally considered (e.g. ISO 19115 or other community agreed one).

7.1.3 Properties of an Observation

An observation results in a value being assigned to a phenomenon. The phenomenon is a property of a feature, the latter being the feature-of-interest of the observation. The observation uses a specified procedure performed by an observer, which is often an instrument or sensor [1][2], but may be a process chain, human observer, an algorithm, a computation or simulator. The key idea is that the observation result is an estimate of the value of some property of the feature-of-interest, and the other properties of the observation properties of the observation provide context or metadata to support evaluation, interpretation and use of the result.

The relationship between the properties of an observation and those of its feature-of-interest is key to the semantics of the model. This is further elaborated in Annex D.3.

7.1.4 Observation location

The principal location of interest is usually associated with the ultimate feature-of-interest.

However, the location of the feature-of-interest may not be trivially available. For example: in remote sensing applications, a complex processing chain is required to geolocate the scene or swath; in feature-detection applications the initial observation may be made on a scene, but the entity to be detected, which is the ultimate feature-of-interest, occupies some location within it. The distinction between the proximate and ultimate feature-of-interest is a key consideration in these cases.

Other locations appear in various scenarios. Sub-sampling at locations within the feature-of-interest may occur. The procedure may involve a sensor located remotely from the ultimate feature-of-interest like in remote sensing, or where specimens are removed from their sampling location and observations made ex-situ (the sampling schema description below elaborates on this). Furthermore, the location of the feature-of-interest may be time-dependent.

The model is generic. The geospatial location of the feature-of-interest may be of little or no interest for some observations (e.g. live specimens, observations made on non-located things like chemical species).

For these reasons, a generic Observation class does not have an inherent location property. Relevant location information should be provided by the feature-of-interest, by the sampling procedure, or by the observation procedure, according to the specific scenario.

NOTE In contrast to spatial properties, some temporal properties are associated directly with an observation (Clauses 8.2.3; 8.2.4). This is a consequence of the fact that an observation is a kind of 'event' so its temporal characteristics are fundamental, rather than incidental.

7.1.5 Result types

Observation results may have many datatypes, including primitive types like category or measure, but also more complex types such as time, location and geometry. Complex results are obtained when the observed property requires multiple components for its encoding. Furthermore, if the property varies on the feature-of-interest, then the result is a coverage, whose domain extent is the extent of the feature. In a physical realization, the result will typically be sampled discreetly on the domain, and may be represented as a discrete coverage.

Building on this, Specialized observation types as defined by communities help describe the type of result provided.

7.1.6 Use of the observation model

The Observation model takes a data-user-centric viewpoint, emphasizing the semantics of the feature-of-interest and its properties. This contrasts with Sensor-oriented models, which take a process- and thus provider-centric viewpoint.

An observation is a property-value-provider for a feature-of-interest. Aside from the result, the details of the observation event are primarily of interest in applications where an evaluation of errors in the estimate of the value of a property is of concern. The Observation could be considered to carry “property-level” instance metadata, complementing the dataset-level and feature-level metadata that have been conventionally considered (e.g. ISO 19115).

Additional discussion of the application of the observation and sample models, and nuances within these, is provided in Annex D.

7.2 Sample schema

7.2.1 Role of sample features

A Sample may act as a proxy for the ultimate feature-of-interest of an Observation, and be associated with this Observation by the role featureOfInterest as a specialization of Any. In this case the sampledFeature association of Sample would point upwards in the chain of sampled features leading to ultimate feature-of-interest of the Observation. The Sample may associate itself with the Observation in question by the role relatedObservation.

7.2.2 Proximate vs. ultimate feature-of-interest

7.2.2.1 Introduction

The observation model maps the result of the application of a procedure to a subject, which plays the role of feature-of-interest of the observation. However, the proximate feature-of-interest of an observation may not be the ultimate domain-specific feature whose properties are of interest in the investigation of which the observation is a part. There are three circumstances that can lead to this:

a) the observation does not obtain values for the whole of a domain feature;

b) the observation is performed on a proxy that is not part of the domain feature;

c) the observation procedure obtains values for properties that are not characteristic of the type of the ultimate feature.

Furthermore, in some practical situations, multiple differences apply.

7.2.2.2 Proximate feature-of-interest embodies a sample design

For various reasons, the domain feature may not be fully accessible. In such circumstances, the procedure for estimating the value of a property of the domain feature involves sampling in representative locations. Then the procedure for transforming a property value observed on the sample to an estimate of the property on the ultimate feature-of-interest depends on the sample design.

EXAMPLE 1 The chemistry of water in an underground aquifer is sampled at one or more positions in a well or bore.

EXAMPLE 2 The magnetic field of the earth is sampled at positions along a flight-line.

EXAMPLE 3 The structure of a rock mass is observed on a cross-section exposed in a river bank.

EXAMPLE 4 The bubble of air around the intake of an air quality monitoring station is taken as representative for the wider air around the station.

In other cases, where direct observation of the domain feature is not possible, the observation may be performed on a proxy.

EXAMPLE 5 In order to measure the intensity of the sun’s light, the reflectance on a white sheet of paper may be utilized as a proxy for the sun’s intensity.

In some cases, the the observation procedure obtains values for properties that are not characteristic of the type of the ultimate feature

EXAMPLE 6 The salinity of water in a Well is measured, the featureOfInterest of this Well is an Aquifer. However, the final target of the observation is the FluidBody contained within the Aquifer (see Figure 8).

7.2.2.3 Observed property is a proxy

The procedure for obtaining values of the property of interest may be indirect, relying on direct observation of a more convenient parameter which is a proxy for the property of interest. Application of an algorithm or processing chain obtains an estimate of the ultimate property of interest.

The observation model requires that the feature-of-interest of the initial observation be of a type that carries the observed property within its properties. Thus, if the proxy property is not a member of the ultimate feature-of-interest, a proxy feature with a suitable model shall be involved.

EXAMPLE 1 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.

EXAMPLE 2 The direct value coming from a sensor may be quantified as a voltage, whereas the observed property represented by this voltage is the physiochemical value being observed by the sensor (ex : pH).

7.2.2.4 Combination

These variations may be combined if exhaustive observation of the domain feature is impractical, and direct measurement is of a proxy property.

EXAMPLE For certain styles of mineralization, the gold concentration of rocks in a region might be estimated through measurement of a related element (e.g. copper), in a specimen of gravel collected from a stream that drains part of the region. The gravel samples the rocks in the catchment of the stream, i.e. in the stream bed and upslope.

7.2.3 Role of Sample

Samples are artefacts of an observational strategy, and have no significant function outside of their role in the observation process. The physical characteristics of the samples themselves are of little interest, except perhaps to the manager of a sampling campaign.

EXAMPLE 1 In various countries/domains, terms like “site”, station” are encountered. These usually correspond to an identifiable locality where a monitoring facility (Host, Platform,...) has been established, sensors or other measurement devices (Observer) have been deployed, to acquire observations on a given observable property applying a specific procedure. In the context of the observation model, the Spatial Sample (both proximate and ultimate) connotes the “world in the vicinity of the Observer/Sampler”, so the observed properties relate to the physical medium at the Observer/Sampler described by the sample, and not to any physical artefact such as a mooring, buoy, benchmark, monument, well, etc, that may be described by Host or derived types.

EXAMPLE 2 In some domains, elements are taken from their natural environment (ex-situ) curated and preserved for the purpose of keeping a trace of their existence (ex : biodiversity studies, crop seed preservation, …). In those cases the Material Samples considered are called Specimen. That’s why the SF_Specimen named class in the previous version of the standard is renamed into MaterialSample in this updated version.

EXAMPLE 3 Statistical Samples usually apply to populations or other sets, of which certain subset may be of specific interest.

NOTE A transient Spatial Sample, such as a ships-track or flight-line, might be identified and described, but is unlikely to be revisited exactly.

A Sample is intended to sample some object in an application domain. However, in some cases the identity, and even the exact type, of the sampled object may not be known when observations are made using the sample.

7.2.4 Sampling process

Understanding the process by which samples were obtained is often essential to understanding the context of subsequent measurements on this object; different sampling strategies can provide vastly different samples, in turn leading to different result values in observations pertaining to these samples.

A Sample is created through the act of Sampling, whereby a Sampler follows a defined Procedure in order to identify and/or extract representative Samples from the ultimate feature-of-interest.

The nature of the Sampler varies by sampling strategy; at one end of the spectrum the Sampler can be a sensor or other automated measurement device; at the other end of the spectrum the Sampler can be a human being providing observations or taking part in a biodiversity survey campaign.

In dependence on the sampling strategy, a sampling procedure appropriate to the Sampling to be performed must be selected and defined. For the provision of fine grained information pertaining to the sampling process, multiple sampling procedures can be applied to one Sampling. Multiple sampling procedures may also be required for the case where one sampling process classifies samples in accordance with multiple criteria.

EXAMPLE When performing observations on populations, these may first be sampled by gender and age. Sampling Procedures describing the criteria utilized for gender and age classification can be provided individually.

A sampling event may involve very different Samples, whereby some of these samples may serve purely to provide contextual information pertaining to the Sampling event.

EXAMPLE When sampling water from a river, information on the meteorology at the time of sampling may be relevant for the interpretation of measurements obtained on the water sample.

7.2.5 Classification of samples

A small number of sampling patterns are common across disciplines in observational science. These provide a basis for processing and portrayal tools which are similar across domains, and depend particularly on the geometry of the sample design. Common names for sampling features include specimen, sample, site, profile, transect, path, swath and scene.

Spatial sampling is classified primarily by the topological dimension. Material samples may provide information on their original source location, but are more often characterized by their size and storage location.

In addition, various preparation steps may be performed on samples both before and after observations are performed on the sample.

Additional information on provenance, curation and archivation has been delegated to external standards, that may be referenced via the ‘metadata’ association that can be provided for all types contained within the Sampling model.

7.3 Alignment between Observation, Sample and domain models

7.3.1 Model consistency

The type of the feature-of-interest is defined in an application schema (ISO 19109). This may be part of a domain model, or may be from a cross-domain model, such as Sample (Clause 11). The feature type defines its set of characteristics as properties. For consistency, the feature-of-interest shall carry the observed property as part of the definition of its type (e.g. Figure 5).

EXAMPLE

A pallet with the characteristic mass is to be described via a feature model. In the simplest form, an interface “Pallet” may be defined as having the attribute “mass” of type “Measure” describing the mass characteristic of the pallet being described (Figure 4). However, when using this direct approach, no further measurement metadata is available, only the numeric mass is provided together with the unit of measurement.

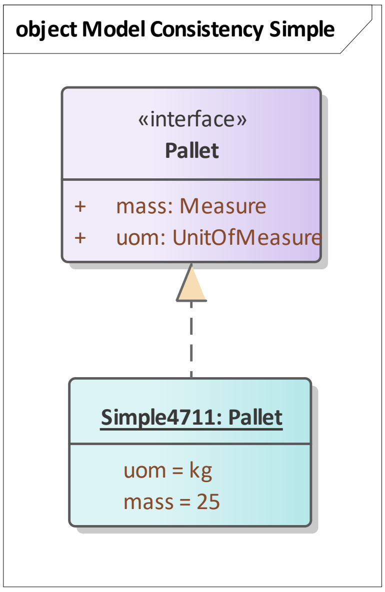

Figure 4 — (Example) Pallet interface, simple example for model consistency.

Alternatively through utilization of the Observations & Measurements model, an observation providing the value of this property for the feature being investigated may be utilized to fulfil the data requirements ensuing from the Pallet Interface. This approach makes it possible for the information system to 'describe' how the result (here mass value) was obtained together with the relevant value.

For this purpose, the observation shall have observedProperty "mass", the result shall be of the type "Measure" and the scale (unit of measure) shall be suitable for mass measurements. Thus, the requirements ensuing from the Pallet Interface are fulfilled, while additional relevant measurement meta-information is also provided; model consistency has been ensured. This approach is illustrated in Figure 5.

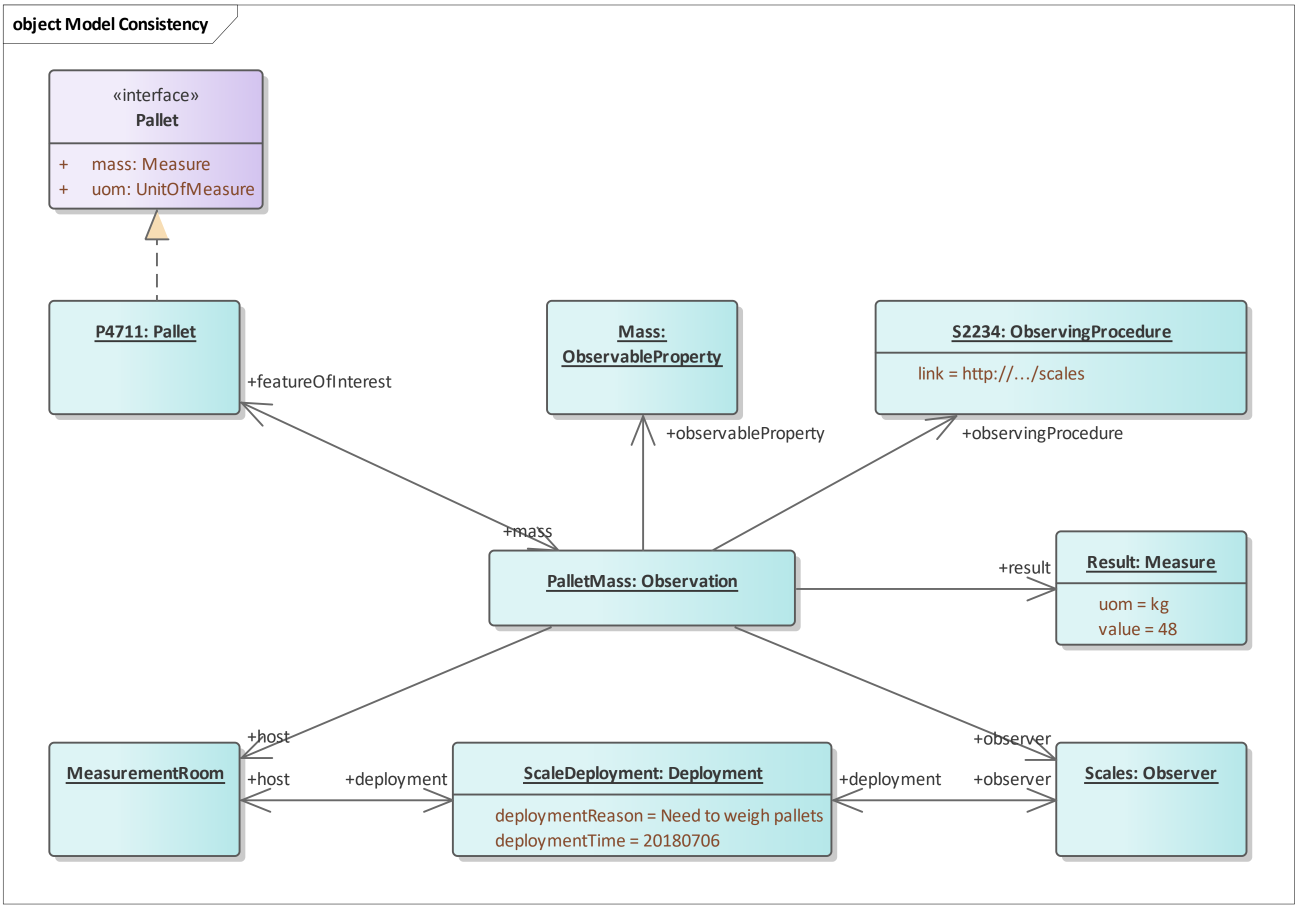

Figure 5 — (Example) An observation with consistent properties: the observed property (mass) is a phenomenon associated with the type of the feature-of-interest (Pallet) and the procedure and result type are also suitable.

An attribute from within the conceptual model can be instantiated as an Observation in the concrete realization. The attributes that have been defined for the domain feature within the interface, in the example “mass” and “uom”, can be realized through the association of an observation carrying this information. Formally, these two representations both realize the defined interface.

It is a modelling choice to decide, based on the use case, whether the solely providing information of type ‘Measure’ with uom is sufficient for the domain considered or whether the full Observations & Measurements model is required to actually discover, exchange and reuse data properly. For example a single attribute ‘lake surface’ will be sufficient for most mapping agency needs whereas a more thorough Observation description of how that surface was measured and when (e.g : dam empty/full, rainfall observation...) is important for water management needs.

7.3.2 Relationship between Sample and domain features

A Sample feature is established in order to make observations concerning some domain feature. The association with the role sampledFeature shall link the Sample to the feature which the sampling feature was designed to sample. The target of this association has the role sampledFeature with respect to the sampling feature, and shall not be a sampling feature or observation. It is usually a real-world feature from an application domain (see Figure 6).

EXAMPLE A profile typically samples a water- or atmospheric-column; a well samples the water in an aquifer; a tissue specimen samples a part of an organism.

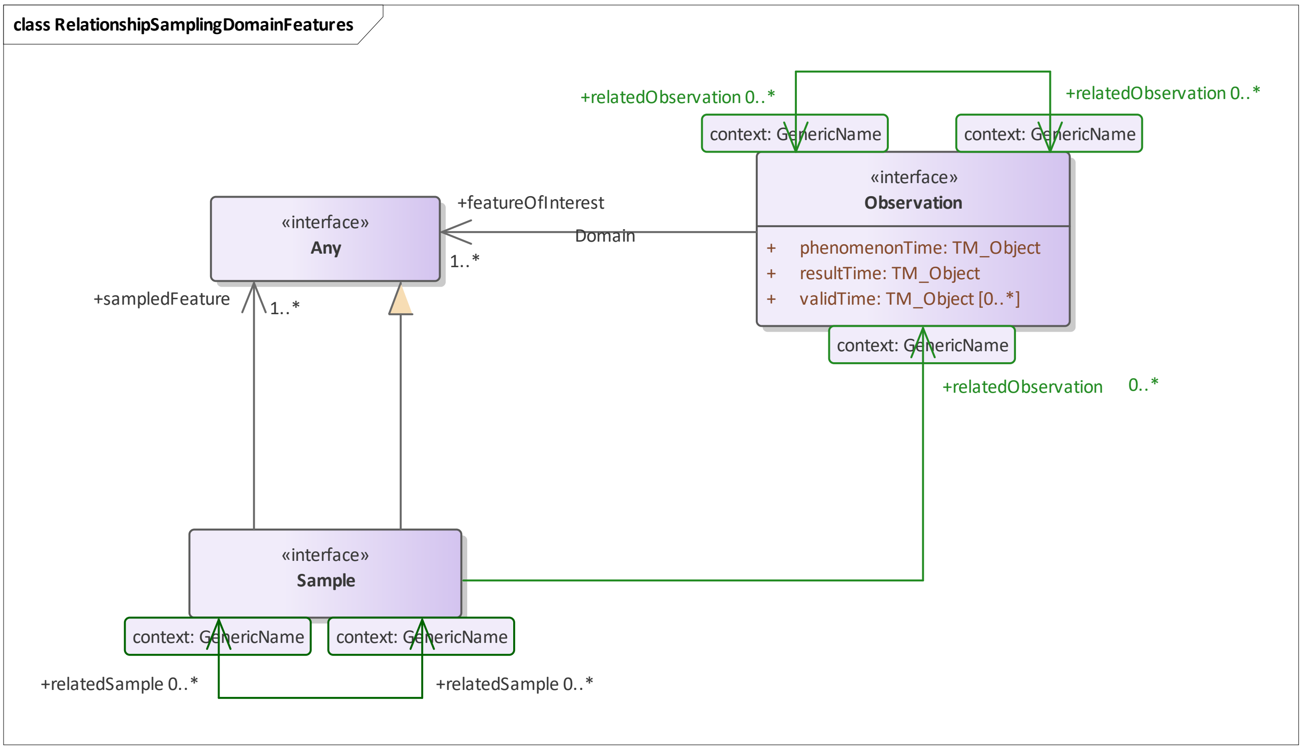

Figure 6 — (Informative) Relationship between Sample and domain features.

Both the Sample and the Domain feature may appear as the feature-of-interest. If a Sample feature is involved, it samples a feature of a type defined in a domain model.

Any Domain object can be a featureOfInterest of an Observation.

The more refined example described in Figure 7 further explains how both Sample and Observation from the Observations & Measurements model can interact with a domain model.

In this example, Well, Aquifer and FluidBody are modeled outside the Observations & Measurements model but

- The Well also conforms to the Sample requirements
- Instances from the domain model are the proximate and ultimate features of interest of the WaterSalinity Observation.

The Well that samples the Aquifer acts as a proxy to the Aquifer in the observation act. It is thus considered as the proximateFeatureOfInterest of the Observation. The sampledFeature (the Aquifer) being the ultimateFeatureOfInterest.

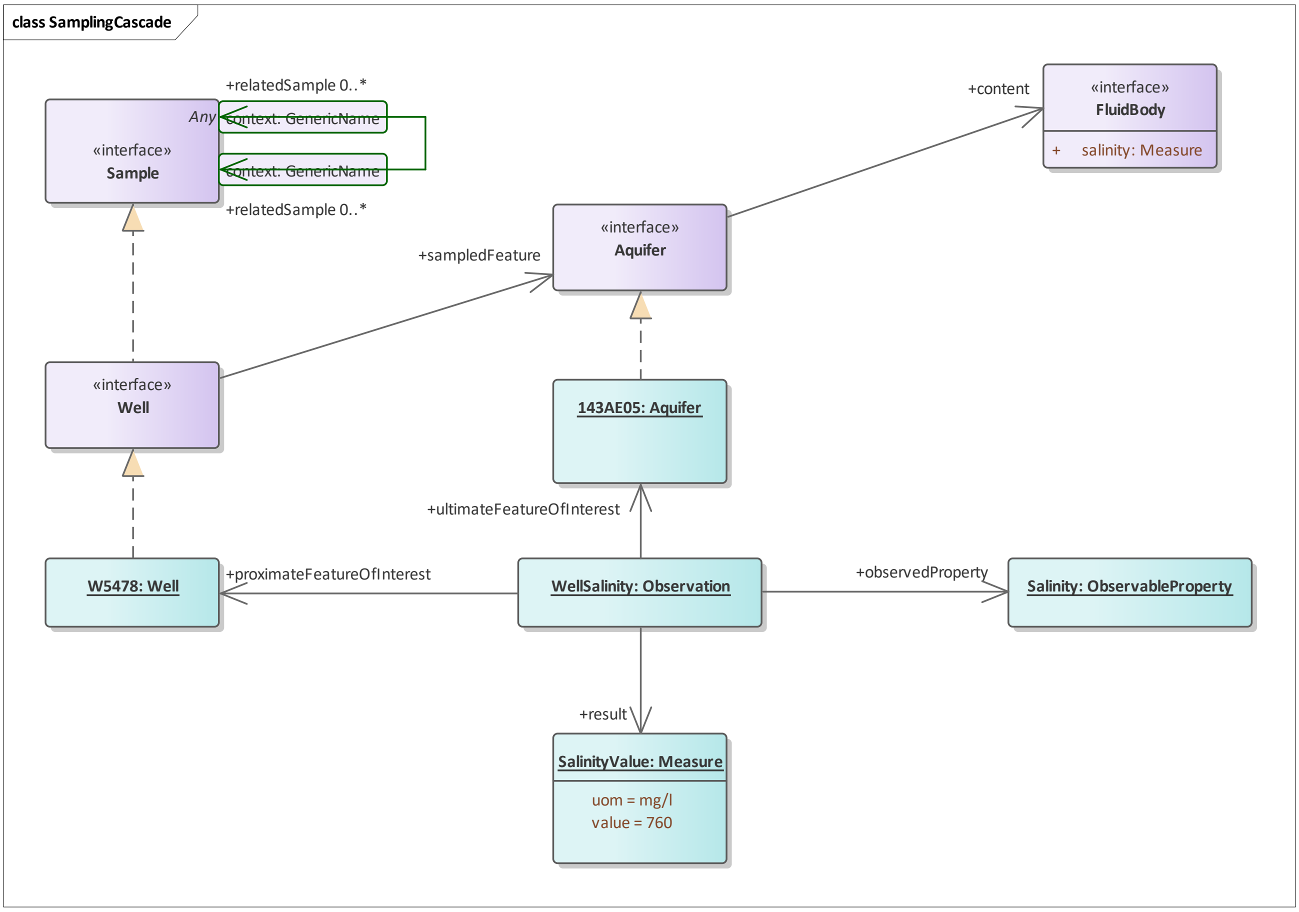

Figure 7 — (Example) Sampling Cascade example including domain features.

Depending on the use case, one might want to push the modelling choice a step further and instantiate a FluidBody in the system according to the semantic of the domain model (Well, Aquifer, FluidBody). That example is further refined in Figure 8. Then depending on the viewpoint considered, either the instance of the Aquifer and/or the instance of the FluidBody can be considered as the ultimateFeatureOfInterest of the Observation. The Well remains the proximateFeatureOfInterest.

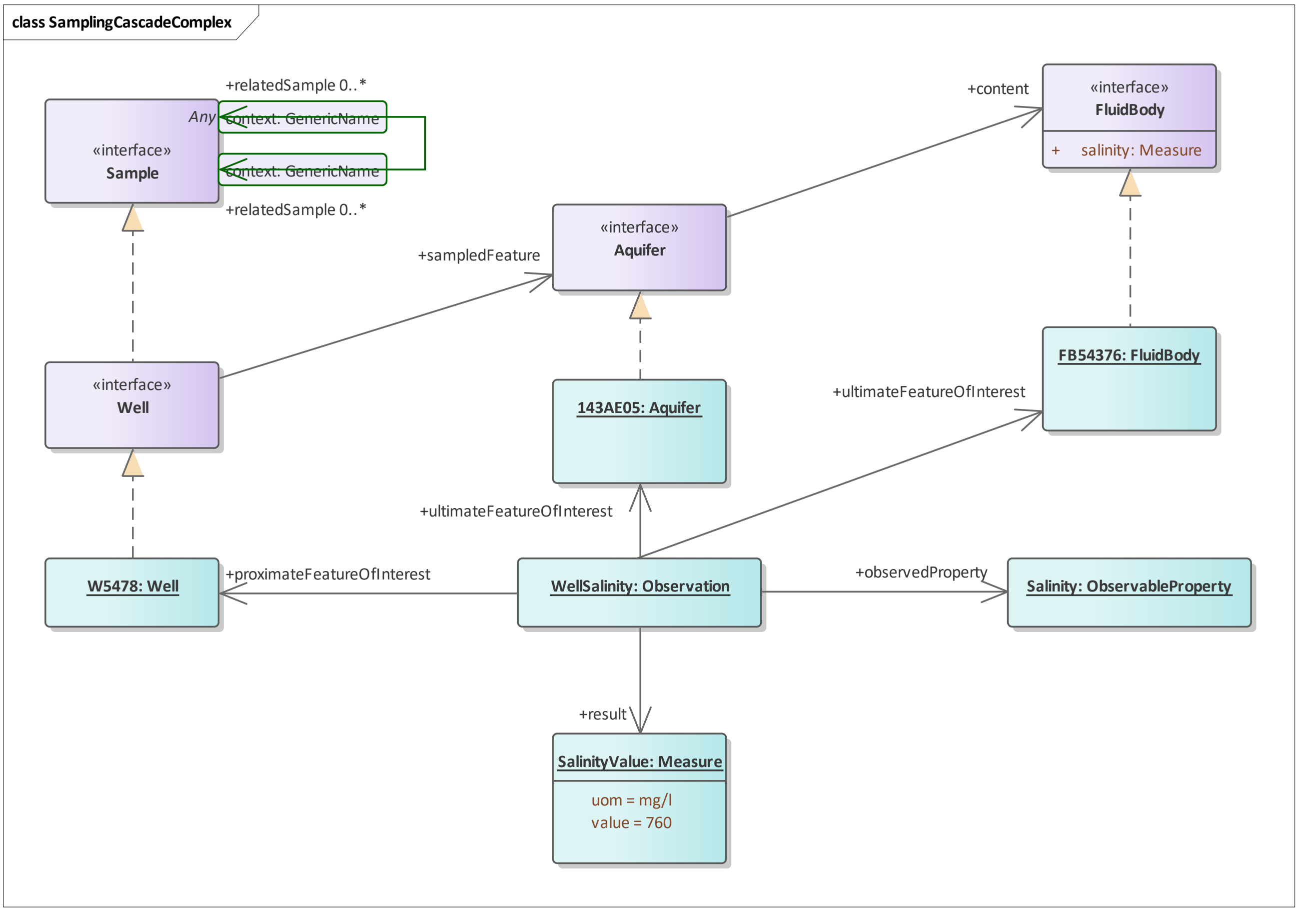

Figure 8 — (Example) Complex Sampling Cascade example referencing external domain feature.

8 Conceptual Observation schema

8.1 General

8.1.1 Conceptual Observation model

The Conceptual Observation schema described as a class diagram in Figure 9. It is fully described in the Clause 8.1.2.

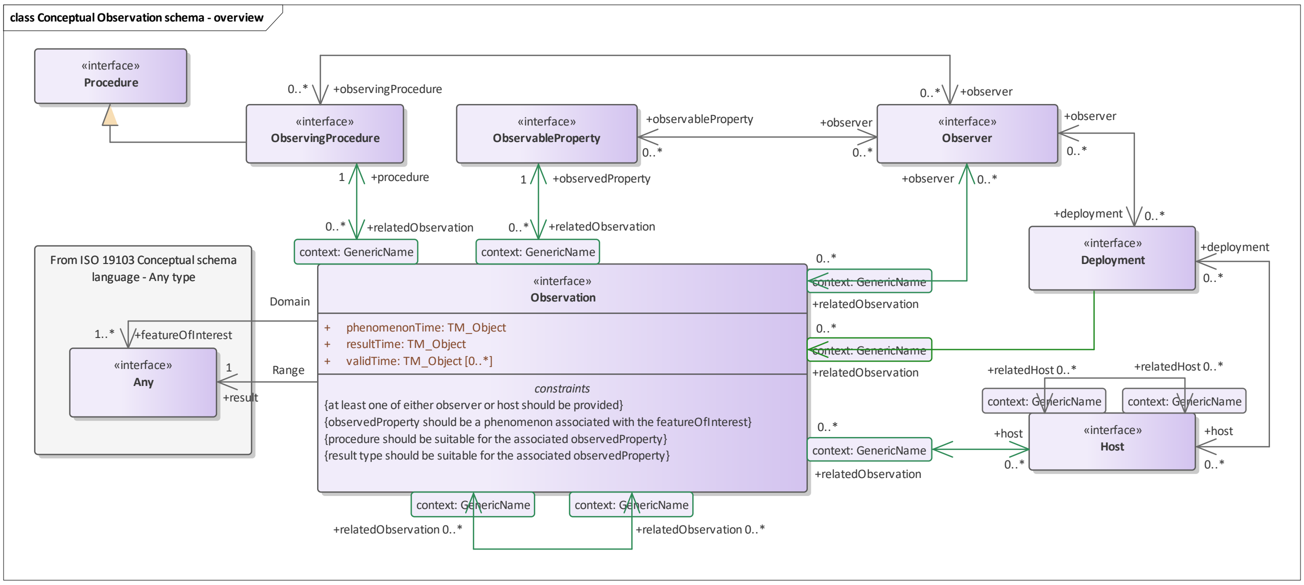

Figure 9 — Conceptual Observation schema overview.

8.1.2 Conceptual Observation schema package Requirements Class

Requirements Class	/req/obs-cpt
Target type	Conceptual model
Name	Conceptual Observation schema package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Observation
Imports	/req/obs-cpt/ObservableProperty
Imports	/req/obs-cpt/Procedure
Imports	/req/obs-cpt/ObservingProcedure
Imports	/req/obs-cpt/Observer
Imports	/req/obs-cpt/Host
Imports	/req/obs-cpt/Deployment

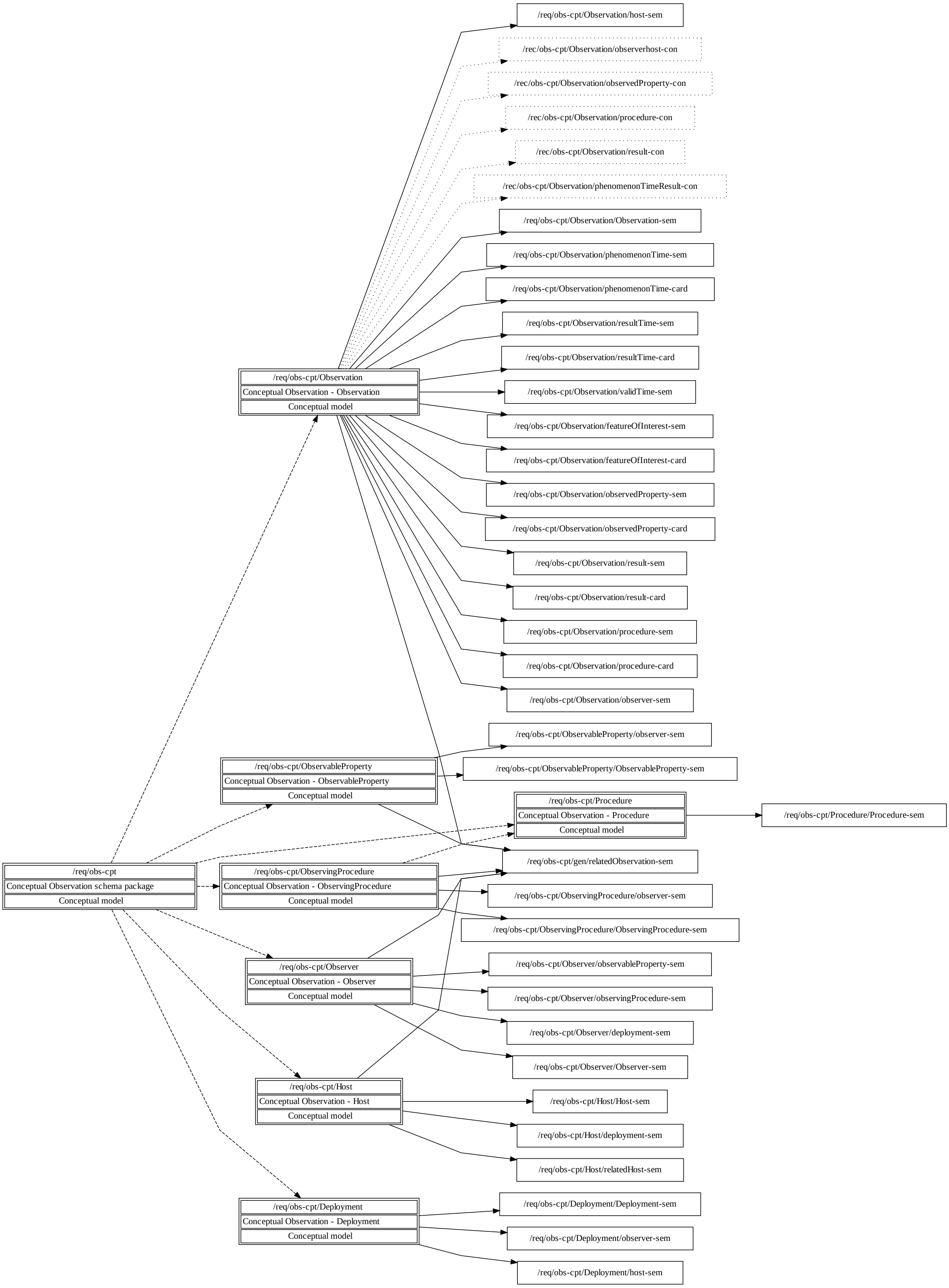

Figure 10 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Observation schema package requirements class.

8.1.3 Association relatedObservation

Requirement /req/obs-cpt/gen/relatedObservation-sem	An **Observation** the **Observation** is related to. If a reference to a related **Observation** is provided, the association with role **relatedObservation** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

8.2 Observation

8.2.1 Observation Requirements Class

Requirements Class	/req/obs-cpt/Observation
Target type	Conceptual model
Name	Conceptual Observation - Observation
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for data transfer conformance class
Requirement	/req/obs-cpt/Observation/Observation-sem
Requirement	/req/obs-cpt/Observation/phenomenonTime-sem
Requirement	/req/obs-cpt/Observation/phenomenonTime-card
Requirement	/req/obs-cpt/Observation/resultTime-sem
Requirement	/req/obs-cpt/Observation/resultTime-card
Requirement	/req/obs-cpt/Observation/validTime-sem
Requirement	/req/obs-cpt/Observation/featureOfInterest-sem
Requirement	/req/obs-cpt/Observation/featureOfInterest-card
Requirement	/req/obs-cpt/Observation/observedProperty-sem
Requirement	/req/obs-cpt/Observation/observedProperty-card

Requirement	/req/obs-cpt/Observation/result-sem
Requirement	/req/obs-cpt/Observation/result-card
Requirement	/req/obs-cpt/Observation/procedure-sem
Requirement	/req/obs-cpt/Observation/procedure-card
Requirement	/req/obs-cpt/Observation/observer-sem
Requirement	/req/obs-cpt/Observation/host-sem
Recommendation	/rec/obs-cpt/Observation/observerhost-con
Recommendation	/rec/obs-cpt/Observation/observedProperty-con
Recommendation	/rec/obs-cpt/Observation/procedure-con
Recommendation	/rec/obs-cpt/Observation/result-con
Recommendation	/rec/obs-cpt/Observation/phenomenonTimeResult-con
Requirement	/req/obs-cpt/gen/relatedObservation-sem

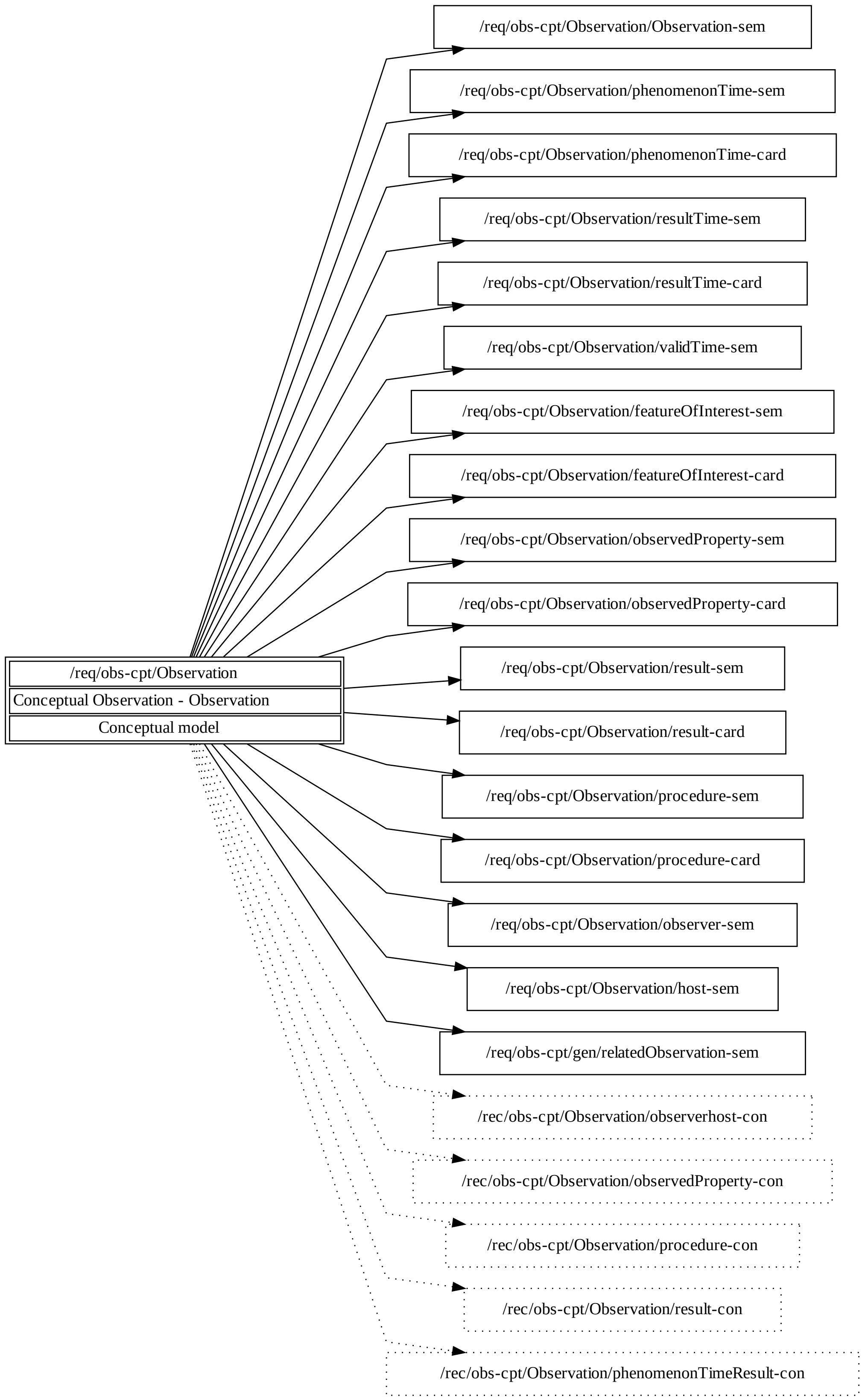

Figure 11 — (Informative) Included requirements and recommendations of the Conceptual Observation schema — Observation requirements class.

8.2.2 Interface Observation

Requirement /req/obs-cpt/Observation/Observation-sem	An observation is an act carried out by an **Observer** to determine the value of an **ObservableProperty** of an object (**featureOfInterest)** by using a **Procedure**; the value is provided as the **result.**

NOTE It is important to note that the terms 'observation', 'interpretation', 'forecast', 'simulation' do correspond to this definition. This aspect is further clarified in .

8.2.3 Attribute phenomenonTime

Requirement /req/obs-cpt/Observation/phenomenonTime-sem	The time that the **Result** applies to the **FeatureOfInterest.** If the **phenomenonTime** is described, this SHALL be provided by the attribute **phenomenonTime:TM_Object**

NOTE 1 The phenomenonTime is often the time the Sample has been taken.

NOTE 2 This is often the time of interaction by a sampling Procedure or observation Procedure with a real-world feature.

NOTE 3 If the result is the average of multiple samples taken at different times, then the phenomenonTime is the time interval over which these measurements were taken.

Requirement /req/obs-cpt/Observation/phenomenonTime-card	An **Observation** SHALL have exactly 1 **phenomenonTime.**

Recommendation /rec/obs-cpt/Observation/phenomenonTimeResult-con	If the **observedProperty** of an **Observation** is 'occurrence time' then the **result** SHOULD be the same as the **phenomenonTime.**

8.2.4 Attribute resultTime

Requirement /req/obs-cpt/Observation/resultTime-sem	The instant of time when the **result** of the **Observation** became available. If the **resultTime** is described, this SHALL be provided by the attribute **resultTime:TM_Object**

EXAMPLE 1 The resultTime typically corresponds to 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.

EXAMPLE 2 Where a measurement is made on a specimen in a laboratory, the phenomenonTime is the time the specimen was retrieved from its host, while the resultTime is the time the laboratory procedure was applied.

EXAMPLE 3 The resultTime also supports disambiguation of repeat measurements made of the same property of a feature using the same procedure.

EXAMPLE 4 Where sensor observation results are post-processed, the resultTime is the post-processing time, while the phenomenonTime is the time of initial interaction with the world.

EXAMPLE 5 Simulations may be used to 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.

Requirement /req/obs-cpt/Observation/resultTime-card	An **Observation** SHALL have exactly 1 **resultTime.**

8.2.5 Attribute validTime

Requirement /req/obs-cpt/Observation/validTime-sem	The time period during which the **result** is valid. If **validTime(s)** are described they SHALL be provided by the attribute **validTime:TM_Object**

NOTE This attribute is commonly required in forecasting applications.

8.2.6 Association featureOfInterest

Requirement /req/obs-cpt/Observation/featureOfInterest-sem	The subject of the **Observation.** The reference(s) to **featureOfInterest(s)** SHALL be provided using the association **Domain** with the role **featureOfInterest**.

NOTE 1 The featureOfInterest can be of Any type

EXAMPLE 1 An instance of a feature modelled in a specific domain model (Borehole according to OGC GeoSciML)

EXAMPLE 2 The bubble of air around the intake of an air quality monitoring station

EXAMPLE 3 An existing well being used for water quality measurements

NOTE 2 This object is the real-world object whose properties are under observation, or is an object created with the intention to sample the real-world object, as described in the Sampling part of section 7. An observation instance serves as a propertyValueProvider for its feature-of-interest.

Requirement /req/obs-cpt/Observation/featureOfInterest-card	An **Observation** SHALL have at least 1 **featureOfInterest** and MAY have more than 1 in cases objects are created with the intention to sample the real-world object The cardinality of the **featureOfInterest** association SHALL be 1 at minimum.

8.2.7 Association observedProperty

Requirement /req/obs-cpt/Observation/observedProperty-sem	The **ObservableProperty** that is the subject of the **Observation**. If a reference to an **ObservableProperty** is provided, the association with the role **observedProperty** SHALL be used.

Requirement /req/obs-cpt/Observation/observedProperty-card	An **Observation** SHALL have exactly 1 **observedProperty**.

8.2.8 Association result

Requirement /req/obs-cpt/Observation/result-sem	The **Result** of the **Observation**. If a reference to a **result** is provided, the association **Range** with the role **result** SHALL be used.

NOTE 1 The result can be of Any type as it may represent the value of any feature property.

NOTE 2 If the observed property is a spatial operation or function, the type of the result may be a coverage.

NOTE 3 In some contexts, particularly in earth and environmental sciences, the term “observation” is used to refer to the result itself.

Requirement /req/obs-cpt/Observation/result-card	An **Observation** SHALL have exactly 1 **result** .

8.2.9 Association procedure

Requirement /req/obs-cpt/Observation/procedure-sem	The **ObservingProcedure** used by the **Observation** to determine the value of the **ObservableProperty** provided by the **Result.** If a reference to an **ObservingProcedure** is provided, the association with the role **procedure** SHALL be used.

EXAMPLE Observed radiance wavelength is determined by the response characteristics of the sensor.

A description of the observation procedure provides or implies an indication of the reliability or quality of the observation result.

8.2.10 Association observer

Requirement /req/obs-cpt/Observation/observer-sem	An **Observer (ref)** that is involved in the creation of this **Observation**. If a reference to an **Observer** is provided, the association with the role **observer** SHALL be used.

8.2.11 Association host

Requirement /req/obs-cpt/Observation/host-sem	A **Host** that is involved in the creation of this **Observation** If a reference to a **Host** is provided, the association with the role **host** SHALL be used.

8.2.12 Constraint Observer or Host

Recommendation /rec/obs-cpt/Observation/observerhost-con	At least one **Observer** or **Host** SHOULD be provided

8.2.13 Constraint ObservableProperty phenomenon associated with featureOfInterest

Recommendation /rec/obs-cpt/Observation/observedProperty-con	The **ObservableProperty** referenced by **observedProperty** SHOULD correspond to a characteristic or phenomenon associated with the **featureOfInterest**

8.2.14 Constraint suitable ObservableProperty

Recommendation /rec/obs-cpt/Observation/procedure-con	The **ObservingProcedure** referenced by **procedure** SHOULD be suitable for the associated **ObservableProperty**

8.2.15 Constraint suitable result type

Recommendation /rec/obs-cpt/Observation/result-con	The type of the result provided by the **result** association SHOULD be suitable for the associated **ObservableProperty**

8.3 ObservableProperty

8.3.1 ObservableProperty Requirements Class

Requirements Class	/req/obs-cpt/ObservableProperty
Target type	Conceptual model
Name	Conceptual Observation - ObservableProperty
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-cpt/ObservableProperty/ObservableProperty-sem
Requirement	/req/obs-cpt/ObservableProperty/observer-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

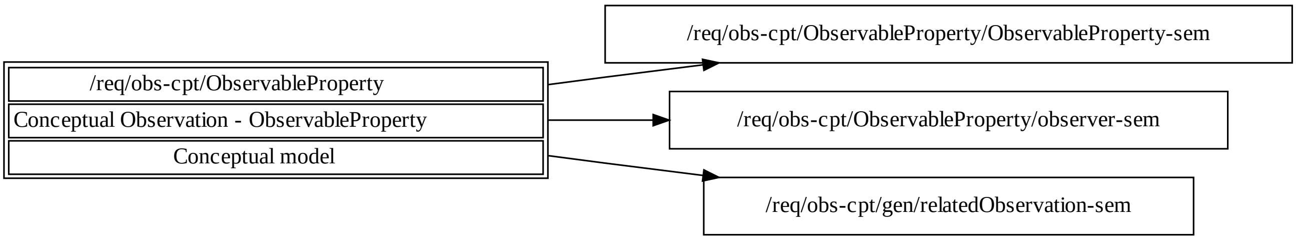

Figure 12 — (Informative) Included requirements and recommendations of the Conceptual Observation schema — ObservableProperty requirements class.

8.3.2 Interface ObservableProperty

Requirement /req/obs-cpt/ObservableProperty/ObservableProperty-sem	An observable quality (property, characteristic) of the **feature of interest** that may be observed.

EXAMPLE 1 The height of a tree, the depth of a water body, or the temperature of a surface are examples of observable properties, while the value of a classic car is not (directly) observable but asserted.

EXAMPLE 2 Groundwater Level

On a groundwater well we

a) monitor Groundwater Level (1 observable property)

b) with an automated probe (that remains in the ground all year, constituting 1 procedure).

Then we have physical campaigns where we revisit the groundwater well and

— measure the Groundwater Level (still the same observable property as above)

— but with a manual probe, this is a different procedure.

This allows to check whether the probe needs recalibration.

8.3.3 Association observer

Requirement /req/obs-cpt/ObservableProperty/observer-sem	An **Observer** capable of observing this **ObservableProperty**. If a reference to the **Observer** is provided, the association with the role **observer** SHALL be used.

8.4 Procedure

8.4.1 Procedure Requirements Class

Requirements Class	/req/obs-cpt/Procedure
Target type	Conceptual model
Name	Conceptual Observation - Procedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-cpt/Procedure/Procedure-sem

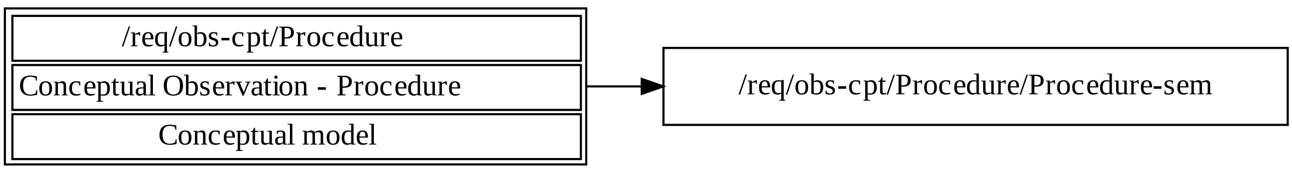

Figure 13 — (Informative) Included requirements and recommendations of the Conceptual Observation schema — Procedure requirements class.

8.4.2 Interface Procedure

Requirement /req/obs-cpt/Procedure/Procedure-sem	A description of steps performed.

NOTE Procedure is an abstract concept that is then further specialized in the various procedure types defined in this document. All share the commonality of describing a defined series of steps to a specific purpose.

The term process that was used in ISO 19156:2011 has been purposely dropped in this version to avoid unnecessary confusion between the terms procedure and process.

8.5 ObservingProcedure

8.5.1 ObservingProcedure Requirements Class

Requirements Class	/req/obs-cpt/ObservingProcedure
Target type	Conceptual model
Name	Conceptual Observation **-** ObservingProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Procedure
Requirement	/req/obs-cpt/ObservingProcedure/ObservingProcedure-sem
Requirement	/req/obs-cpt/ObservingProcedure/observer-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

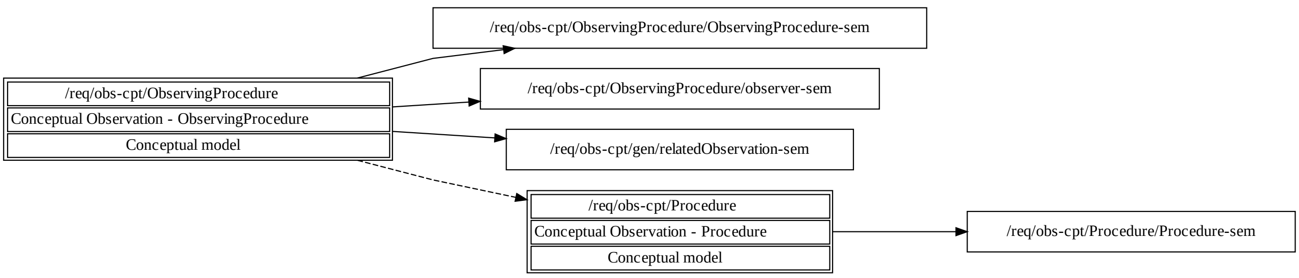

Figure 14 — (Informative) Included requirements and recommendations of the Conceptual Observation schema — ObservingProcedure requirements class.

8.5.2 Interface ObservingProcedure

Requirement /req/obs-cpt/ObservingProcedure/ObservingProcedure-sem	The description of steps performed in order to determine the value of an **observableProperty** by an **Observer.**

NOTES

— Depending on the complexity of the use case, the procedure will be more or less explicitly described. Especially pertaining to historical data, there may be very little or no information available - this information should also be provided.

— The recipe that the observer (cook) follows to generate the observation.

— The procedure is often referred to as the method.

— Different observers can follow the same (reusable) procedure for the creation of different observations.

— The procedure is a workflow, protocol, plan, algorithm, or computational method specifying how to make an observation.

— the observing procedure cannot describe a sensor instance, but it can describe the sensor type.

NOTE The term process that was used in ISO 19156:2011 has been purposely dropped in this version to avoid unnecessary confusion between the terms procedure and process.

EXAMPLE An instance of Procedure is a description of the process utilized by an observer, this could be a chemical analysis method, a protocol for measuring an object, but could also be a checklist utilized by a human observer during a biodiversity campaign. It could further describe the algorithms behind simulators or models used to generate a result from other inputs.

8.5.3 Association observer

Requirement /req/obs-cpt/ObservingProcedure/observer-sem	An **Observer** capable of performing this **ObservingProcedure**. If a reference to an **Observer** is provided, the association with the role **observer** SHALL be used.

8.6 Observer

8.6.1 Observer Requirements Class

Requirements Class	/req/obs-cpt/Observer
Target type	Conceptual model
Name	Conceptual Observation - Observer
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-cpt/Observer/Observer-sem
Requirement	/req/obs-cpt/Observer/observableProperty-sem
Requirement	/req/obs-cpt/Observer/observingProcedure-sem
Requirement	/req/obs-cpt/Observer/deployment-sem

Requirement	/req/obs-cpt/gen/relatedObservation-sem

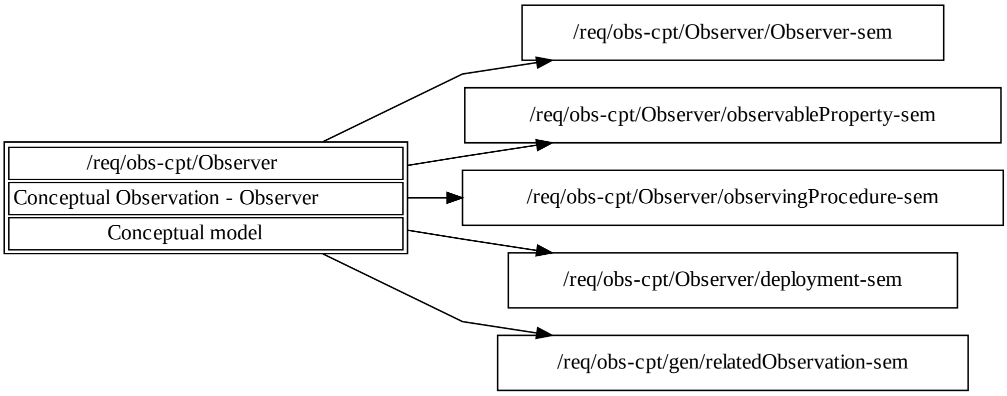

Figure 15 — (Informative) Included requirements and recommendations of the Conceptual Observation — Observer requirements class.

8.6.2 Interface Observer

Requirement /req/obs-cpt/Observer/Observer-sem	An identifiable entity that may generate **Observations** pertaining to an **observableProperty** by implementing a **Procedure**.

NOTES

— Different Observers can follow the same (reusable) observing Procedure for the creation of different Observations.

— The Observer is the entity instance, not the entity type. Pertaining to sensors, the Observer would reference the explicit sensor, while the Procedure would reference the methodology utilized by that sensor type.

— An Observer is closely linked with an observableProperty that it generates results for.

— An Observer can be hosted by one or more Host.

— The Observer is an instance of a sensor, instrument, implementation of an algorithm or a being such as a person.

— An Observer responds to a stimulus, e.g., a change in the environment, or input data composed from the Results of prior Observations, and generates a Result.

EXAMPLE Accelerometers, gyroscopes, barometers, magnetometers, and so forth are Sensors that are typically mounted on a modern smartphone (which acts as Host). Other examples of Sensors include the human eyes.

8.6.3 Association observableProperty

Requirement /req/obs-cpt/Observer/observableProperty-sem	An **ObservableProperty** that this **Observer** can observe. If a reference to **ObservableProperty**(s) is provided, the association with the role **observableProperty** SHALL be used.

8.6.4 Association observingProcedure

Requirement /req/obs-cpt/Observer/observingProcedure-sem	An **ObservingProcedure** that this **Observer** can perform. If a reference to **ObservingProcedure**(s) is provided, the association with the role **observingProcedure** SHALL be used.

8.6.5 Association deployment

Requirement /req/obs-cpt/Observer/deployment-sem	A **Deployment** to which this **Observer** is either physically or organizationally attached. If a reference to **Deployment**(s) is provided, the association with the role **deployment** SHALL be used.

8.7 Host

8.7.1 Host Requirements Class

Requirements Class	/req/obs-cpt/Host
Target type	Conceptual model
Name	Conceptual Observation - Host
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-cpt/Host/Host-sem
Requirement	/req/obs-cpt/Host/deployment-sem
Requirement	/req/obs-cpt/Host/relatedHost-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

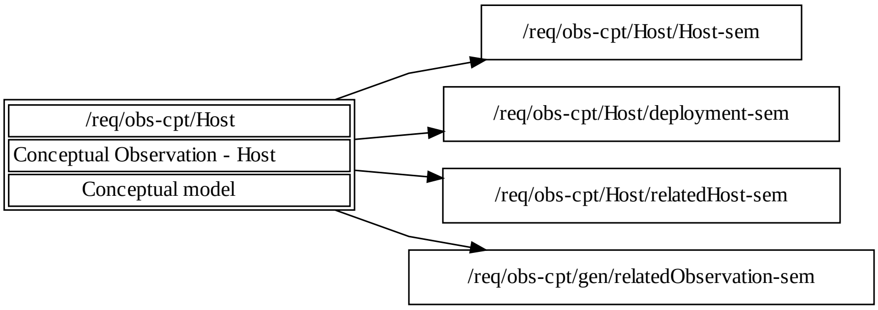

Figure 16 — (Informative) Included requirements and recommendations of the Conceptual Observation — Host requirements class.

8.7.2 Interface Host

Requirement /req/obs-cpt/Host/Host-sem	A grouping of **Observer**s for a specific reason.

NOTES

— In many use cases, the Host is the environmental monitoring facility

— The Host can be a platform that hosts a set of sensors

— An alternative usage could pertain to a biodiversity survey campaign; in this scenario, the team performing the survey would be modelled as observers whereas the entire survey campaign can be represented as a Host.

8.7.3 Association deployment

Requirement /req/obs-cpt/Host/deployment-sem	A **Deployment** at this **Host**. If a reference to a **Deployment** is provided, the association with the role **host** SHALL be used.

8.7.4 Association relatedHost

Requirement /req/obs-cpt/Host/relatedHost-sem	A **Host** the **Host** is related to. If a reference to a related **Host** is provided, the association with role **relatedHost** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

8.8 Deployment

8.8.1 Deployment Requirements Class

Requirements Class	/req/obs-cpt/Deployment
Target type	Conceptual model
Name	Conceptual Observation - Deployment
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-cpt/Deployment/Deployment-sem
Requirement	/req/obs-cpt/Deployment/observer-sem
Requirement	/req/obs-cpt/Deployment/host-sem

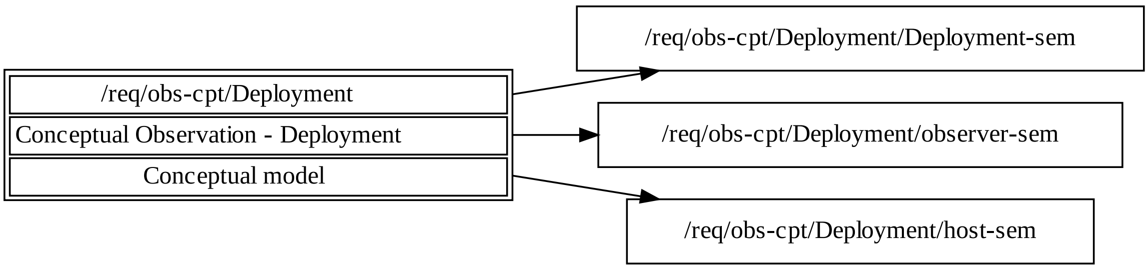

Figure 17 — (Informative) Included requirements and recommendations of the Conceptual Observation — Deployment requirements class.

8.8.2 Interface Deployment

Requirement /req/obs-cpt/Deployment/Deployment-sem	Information on the assignment of an **Observer** to a **Host.**

NOTE A Deployment can range from

— information regarding a sensor being attached to a pole

— the monitoring facilities pertaining to an environmental monitoring network

— the description of a ship cruise to

— the participation of a citizen in a citizen-science project involving crowd sensing.

8.8.3 Association observer

Requirement /req/obs-cpt/Deployment/observer-sem	The **Observer** associated with this **Deployment**. If a reference to an **Observer** is provided, the association with the role **observer** SHALL be used.

8.8.4 Association host

Requirement /req/obs-cpt/Deployment/host-sem	The **Host** to which this **Deployment** pertains. If a reference to a **Host** is provided, the association with the role **host** SHALL be used

9 Abstract Observation Core

9.1 General

9.1.1 Abstract Observation Core Package Requirements Class

Requirements Class	/req/obs-core
Target type	Logical model
Name	Abstract Observation core package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-core/AbstractObservationCharacteristics
Imports	/req/obs-core/AbstractObservation
Imports	/req/obs-core/AbstractObservableProperty
Imports	/req/obs-core/AbstractObservingProcedure
Imports	/req/obs-core/AbstractObserver
Imports	/req/obs-core/AbstractHost
Imports	/req/obs-core/AbstractDeployment

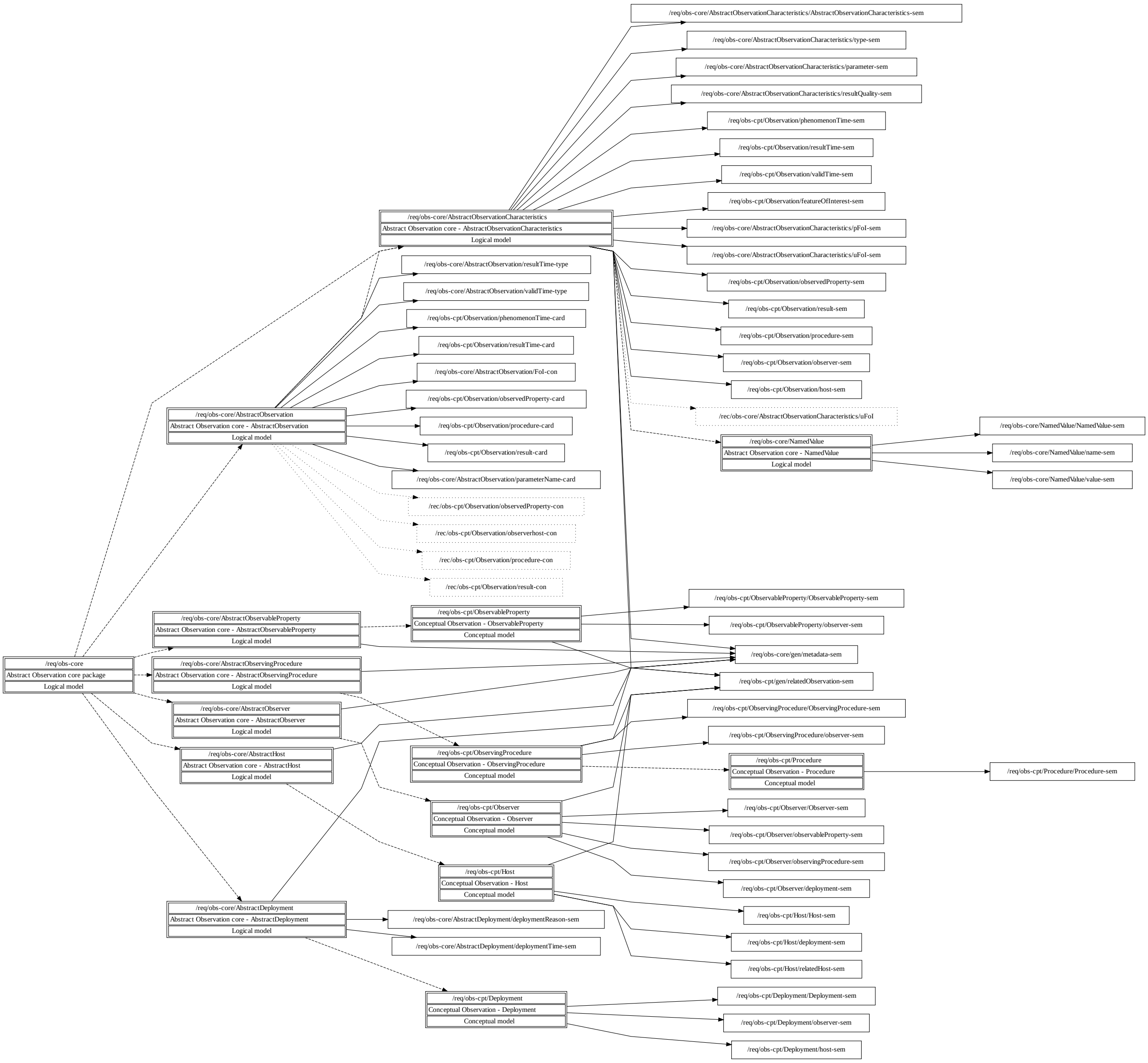

Figure 18 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core package requirements class.

9.1.2 Association metadata

Requirement /req/obs-core/gen/metadata-sem	If descriptive metadata is provided, the association role **metadata** SHALL link to descriptive metadata as commonly understood by communities.

NOTE Attention should be given not to reinvent semantic that is explicitly modeled in the O&M model.

9.2 AbstractObservationCharacteristics

9.2.1 AbstractObservationCharacteristics Requirements Class

Requirements Class	/req/obs-core/AbstractObservationCharacteristics
Target type	Logical model
Name	Abstract Observation core - AbstractObservationCharacteristics
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for data transfer conformance class
Requirement	/req/obs-core/AbstractObservationCharacteristics/AbstractObservationCharacteristics-sem
Requirement	/req/obs-core/AbstractObservationCharacteristics/type-sem
Requirement	/req/obs-core/AbstractObservationCharacteristics/parameter-sem
Requirement	/req/obs-core/AbstractObservationCharacteristics/resultQuality-sem
Requirement	/req/obs-cpt/Observation/phenomenonTime-sem
Requirement	/req/obs-cpt/Observation/resultTime-sem
Requirement	/req/obs-cpt/Observation/validTime-sem
Requirement	/req/obs-cpt/Observation/featureOfInterest-sem
Requirement	/req/obs-core/AbstractObservationCharacteristics/pFoI-sem
Requirement	/req/obs-core/AbstractObservationCharacteristics/uFoI-sem
Requirement	/req/obs-cpt/Observation/observedProperty-sem
Requirement	/req/obs-cpt/Observation/result-sem
Requirement	/req/obs-cpt/Observation/procedure-sem
Requirement	/req/obs-cpt/Observation/observer-sem

Requirement	/req/obs-cpt/Observation/host-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem
Requirement	/req/obs-core/gen/metadata-sem
Recommendation	/rec/obs-core/AbstractObservationCharacteristics/uFoI
Imports	/req/obs-core/NamedValue

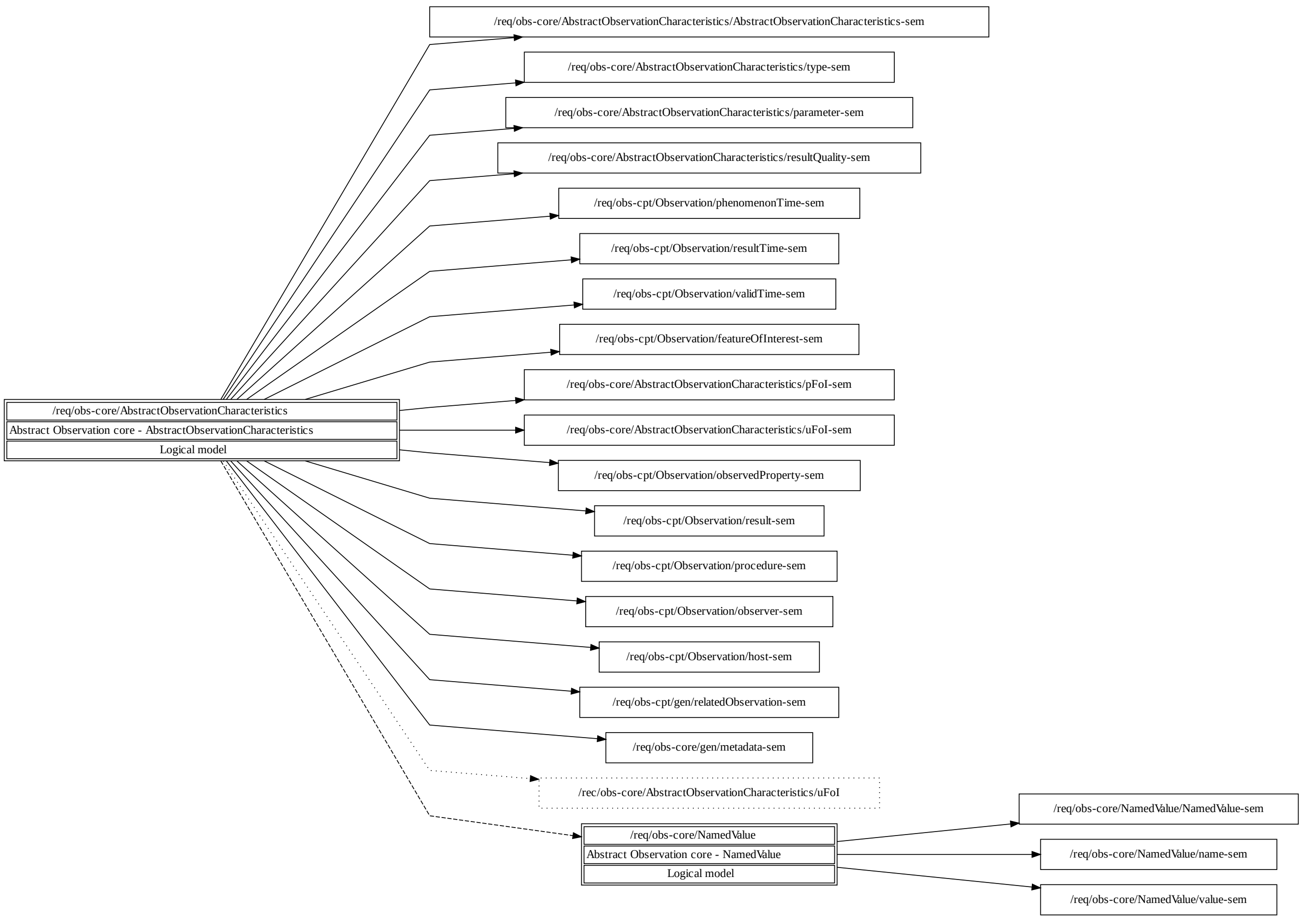

Figure 19 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractObservationCharacteristics requirements class

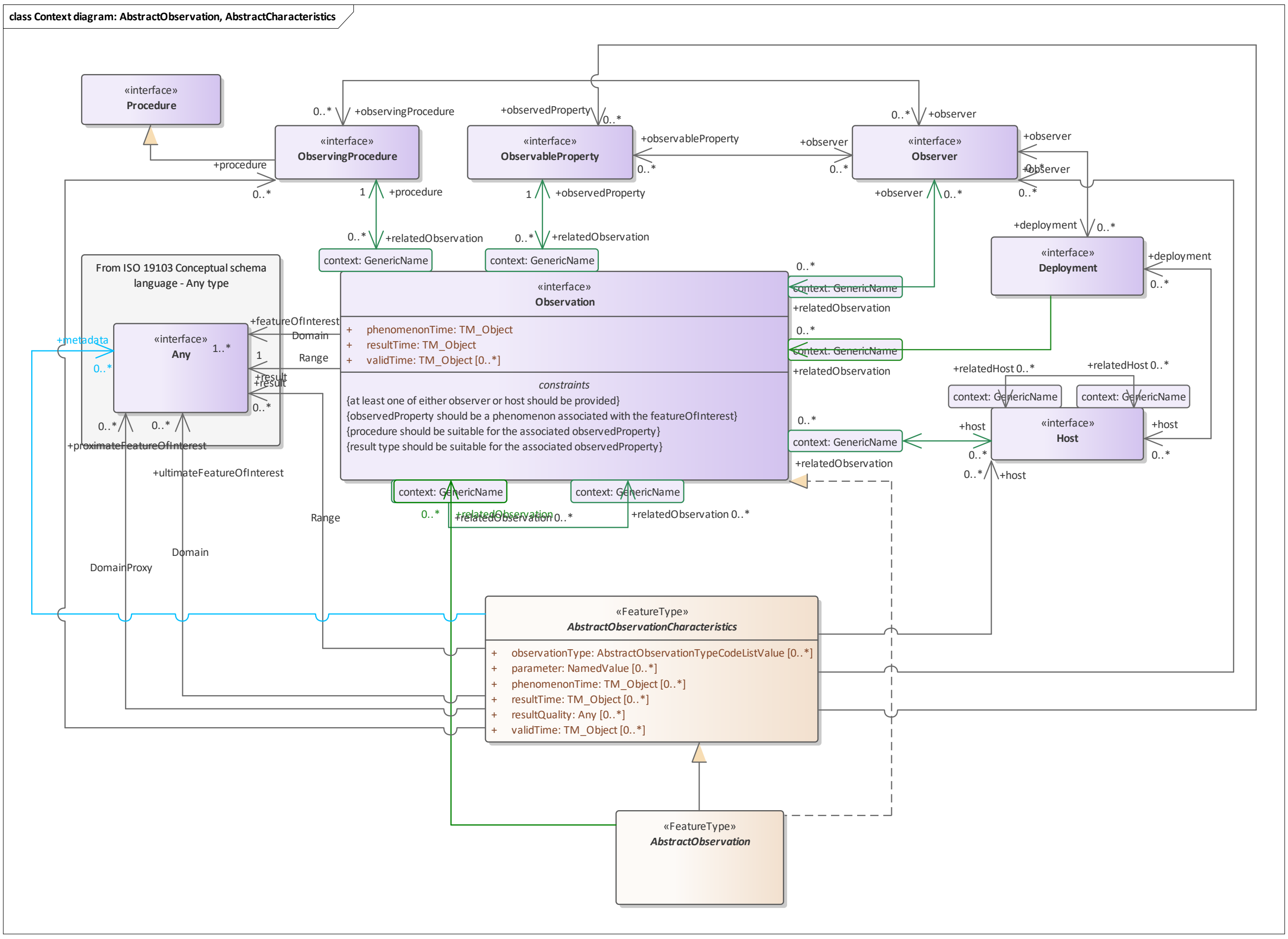

Figure 20 — Context diagram for Abstract Observation core — AbstractObservationCharacteristics and AbstractObservation

9.2.2 Feature type AbstractObservationCharacteristics

Requirement /req/obs-core/AbstractObservationCharacteristics/AbstractObservationCharacteristics-sem	Set of common characteristics used for describing an **Observation** or a collection of Observations.

9.2.3 Attribute observationType

Requirement /req/obs-core/AbstractObservationCharacteristics/type-sem	Information providing further detail on the type of Observations being described by the **AbstractObservationCharacteristics.** If information on the type of Observation is provided, the property **observationType:AbstractObservationTypeCodeListValue** SHALL be used.

NOTE Observation type allows describing the formalism, encoding, etc. to be expected when accessing objects associated to the Observation.

NOTE Multiple types may be applied to one Observation, e.g. in the case where the Observation is being typed both by the feature-of-interest geometry as well as Result type.

9.2.4 Attribute parameter

Requirement /req/obs-core/AbstractObservationCharacteristics/parameter-sem	Arbitrary event-specific parameter relevant to the **AbstractObservationCharacteristics**. If additional parameter information is provided, the property **parameter:NamedValue** SHALL be used.

NOTE 1 Disambiguation: Parameter should not be used instead of the procedure to describe the steps performed in order to determine the value of the ObservableProperty

NOTE 2 Parameter should NOT be utilized to provide information already contained in the model by existing attributes or associations.

NOTE 3 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 feature-of-interest or to the observation procedure. To avoid ambiguity, there shall be no more than one parameter with the same name.

NOTE 4 Parameters that are tightly bound to the procedure can be recorded as part of the procedure description.

The AbstractObservingProcedure is a generic or standard procedure, rather than an event-specific process. In this context, parameters bound to the observation act, such as instrument settings, calibrations or inputs, local position, detection limits, asset identifier, operator, may augment the description of a standard procedure.

EXAMPLE A time sequence of observations of water quality in a well might be made at variable depths within the well. While these can be associated with specimens taken from the well at this depth as the features-of-interest, a more common approach is to identify the well itself as the feature-of-interest, and add a “samplingDepth” parameter to the observation. The sampling depth is of secondary interest compared to the temporal variation of water quality at the site.

9.2.5 Attribute resultQuality

Requirement /req/obs-core/AbstractObservationCharacteristics/resultQuality-sem	Information pertaining to the data quality of the **Result** (**ref**). If additional data quality information is provided, the property **resultQuality:Any** SHALL be used.

NOTE This instance-specific description complements the description of the observation Procedure, which provides information concerning the quality of all observations using this procedure. The quality of a result may be assessed following the procedures in ISO 19157. Multiple measures may be provided.

9.2.6 Association proximateFeatureOfInterest

Requirement /req/obs-core/AbstractObservationCharacteristics/pFoI-sem	The entity that is directly of interest in the act of observing. If a reference to the entity being directly observed is provided, the association with the role **proximateFeatureOfInterest** SHALL be used. This association is a specialization of the **featureOfInterest** role.

NOTE The measurement process may be performed on an intermediary entity referred to as proximateFeatureOfInterest that acts as a proxy to the ultimate feature of interest that is being observed (measured, estimated or calculated).

9.2.7 Association ultimateFeatureOfInterest

Requirement /req/obs-core/AbstractObservationCharacteristics/uFoI-sem	The entity that is ultimately of interest in the act of observing. If a reference to the entity ultimately being observed is provided, the association with the role **ultimateFeatureOfInterest** SHALL be used. This association is a specialization of the **featureOfInterest** role.

NOTE 1 The measurement process may be performed on an intermediary entity that acts as a proxy to the ultimate feature of interest that is being observed (measured, estimated or calculated).

If in the real world both ultimateFeatureOfInterest and proximateFeatureOfInterest exist but not both have a digital representation, then the appropriate relation should be selected that best describes the nature of the entity being referenced.

Recommendation /rec/obs-core/AbstractObservationCharacteristics/uFoI	In the case where ultimate and proximate features-of-interest are the same object, the association SHOULD be provided using the **ultimateFeatureOfInterest** association role.

NOTE 2 There will often be a specifiable relationship between the proximate and ultimate feature of interest, such as a sampling-chain, see Clause 7.2.2 for examples.

EXAMPLE a river, an aquifer, soil layer, outcrop, a butterfly, a survey area, a room, Abby's car, a specific human being, this document

EXAMPLE 1: To determine the concentrations of chemical compounds in a river, a sample is taken in a predefined location in the river. This sample is taken to a laboratory where the required chemical analysis is done. In this case, the river is the ultimateFeatureOfInterest, while the sample is the proximateFeatureOfInterest .

EXAMPLE 2: Pertaining to document and observations on the consistency thereof, for the Observation "This clause is inconsistent", the ultimateFeatureOfInterest is the entire document, while the proximateFeatureOfInterest is the specific clause being addressed.

EXAMPLE 3: the determination of the species of the butterfly, in this case the butterfly is the ultimateFeatureOfInterest, no proximateFeatureOfInterest need be provided.

9.3 AbstractObservation

9.3.1 AbstractObservation Requirements Class

Requirements Class	/req/obs-core/AbstractObservation
Target type	Logical model
Name	Abstract Observation core - AbstractObservation
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for data transfer conformance class
Imports	/req/obs-core/AbstractObservationCharacteristics
Requirement	/req/obs-core/AbstractObservation/resultTime-type
Requirement	/req/obs-core/AbstractObservation/validTime-type
Requirement	/req/obs-cpt/Observation/phenomenonTime-card
Requirement	/req/obs-cpt/Observation/resultTime-card
Requirement	/req/obs-core/AbstractObservation/FoI-con
Requirement	/req/obs-cpt/Observation/observedProperty-card
Requirement	/req/obs-cpt/Observation/procedure-card
Requirement	/req/obs-cpt/Observation/result-card
Recommendation	/rec/obs-cpt/Observation/observedProperty-con
Recommendation	/rec/obs-cpt/Observation/observerhost-con
Recommendation	/rec/obs-cpt/Observation/procedure-con

Recommendation	/rec/obs-cpt/Observation/result-con
Requirement	/req/obs-core/AbstractObservation/parameterName-card

Figure 21— (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractObservation requirements class

9.3.2 Constraint resultTime instant

Requirement /req/obs-core/AbstractObservation/resultTime-type	If the result time of the **Observation** is provided, the **resultTime** attribute SHALL be of type **TM_Instant.**

9.3.3 Constraint validTime period

Requirement /req/obs-core/AbstractObservation/validTime-type	If the result time of the **Observation** is provided, the **validTime** attribute SHALL be of type **TM_Period.**

9.3.4 Constraint parameter unique name

Requirement /req/obs-core/AbstractObservation/parameterName-card	The **name** attribute of a **parameter NamedValue** SHALL be unique within an **Observation.**

9.3.5 Constraint proximate or ultimate featureOfInterest

Requirement /req/obs-core/AbstractObservation/FoI-con	at least one **proximateFeatureOfInterest** or **ultimateFeatureOfInterest** SHALL be given.

9.4 AbstractObservableProperty

9.4.1 AbstractObservableProperty Requirements Class

Requirements Class	/req/obs-core/AbstractObservableProperty
Target type	Logical model
Name	Abstract Observation core - AbstractObservableProperty
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/ObservableProperty
Requirement	/req/obs-core/gen/metadata-sem

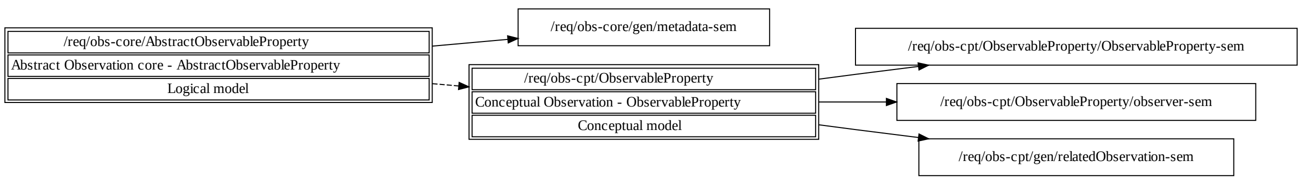

Figure 22— (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractObservableProperty requirements class

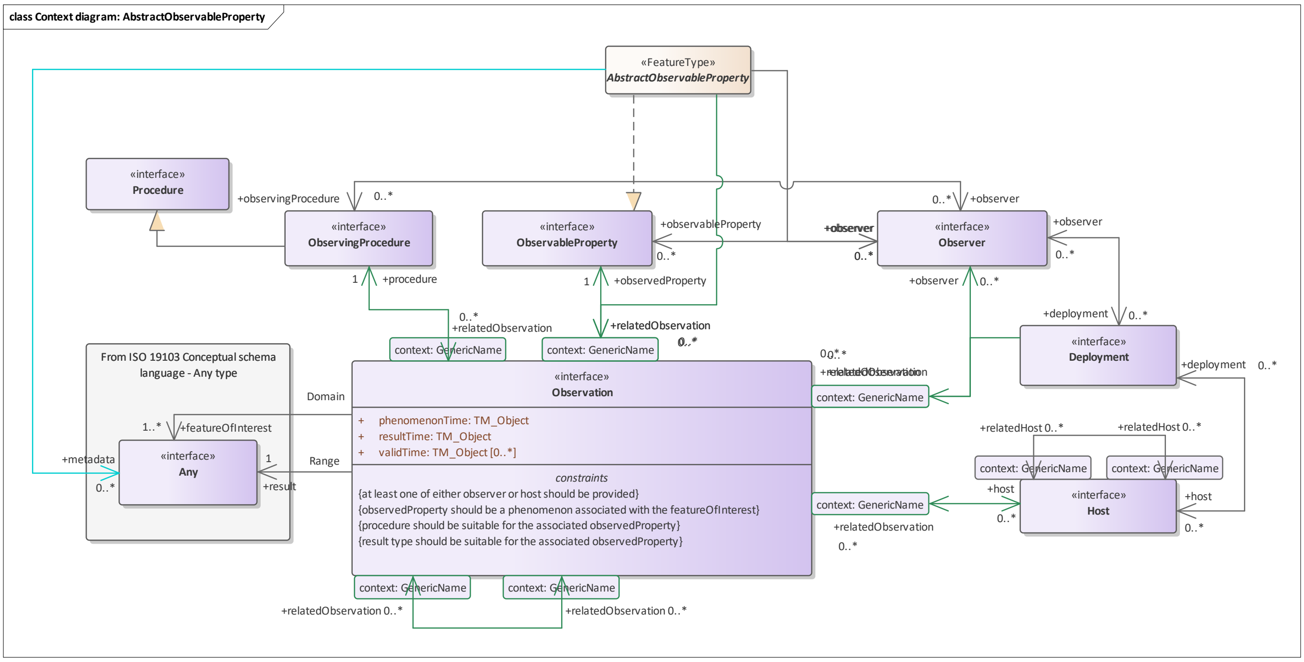

Figure 23 — Context diagram for Abstract Observation core — AbstractObservableProperty

9.5 AbstractObservingProcedure

9.5.1 AbstractObservingProcedure Requirements Class

Requirements Class	/req/obs-core/AbstractObservingProcedure
Target type	Logical model
Name	Abstract Observation core - AbstractObservingProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/ObservingProcedure
Requirement	/req/obs-core/gen/metadata-sem

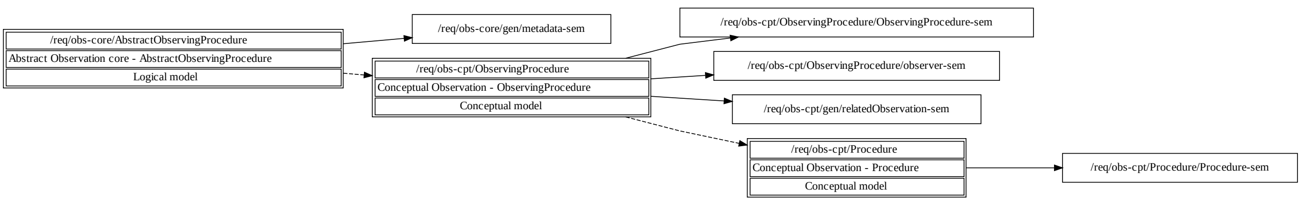

Figure 24 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractObservingProcedure requirements class

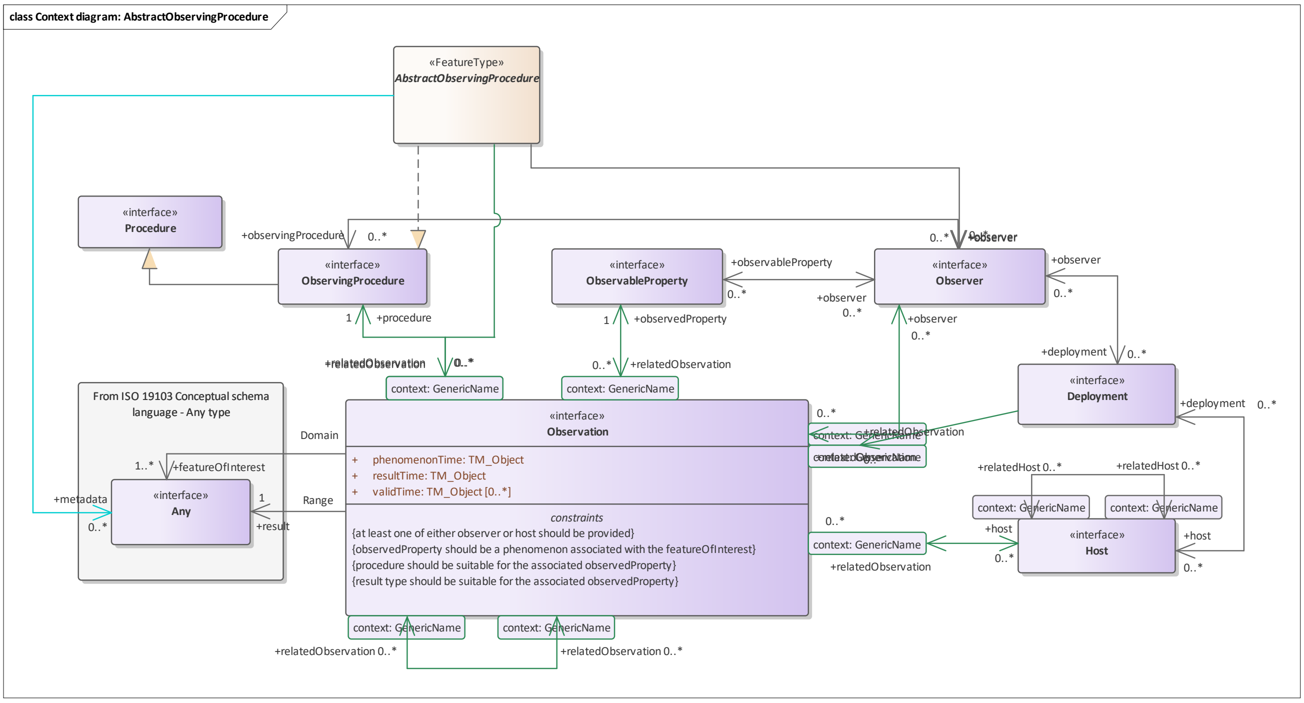

Figure 25 — Context diagram for Abstract Observation core — AbstractObservingProcedure

9.6 AbstractObserver

9.6.1 AbstractObserver Requirements Class

Requirements Class	/req/obs-core/AbstractObserver
Target type	Logical model
Name	Abstract Observation core - AbstractObserver
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Observer
Requirement	/req/obs-core/gen/metadata-sem

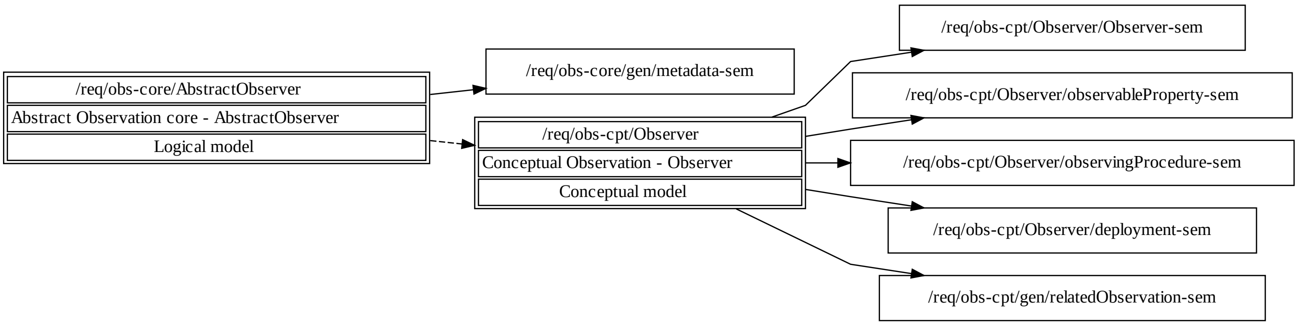

Figure 26 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractObserver requirements class

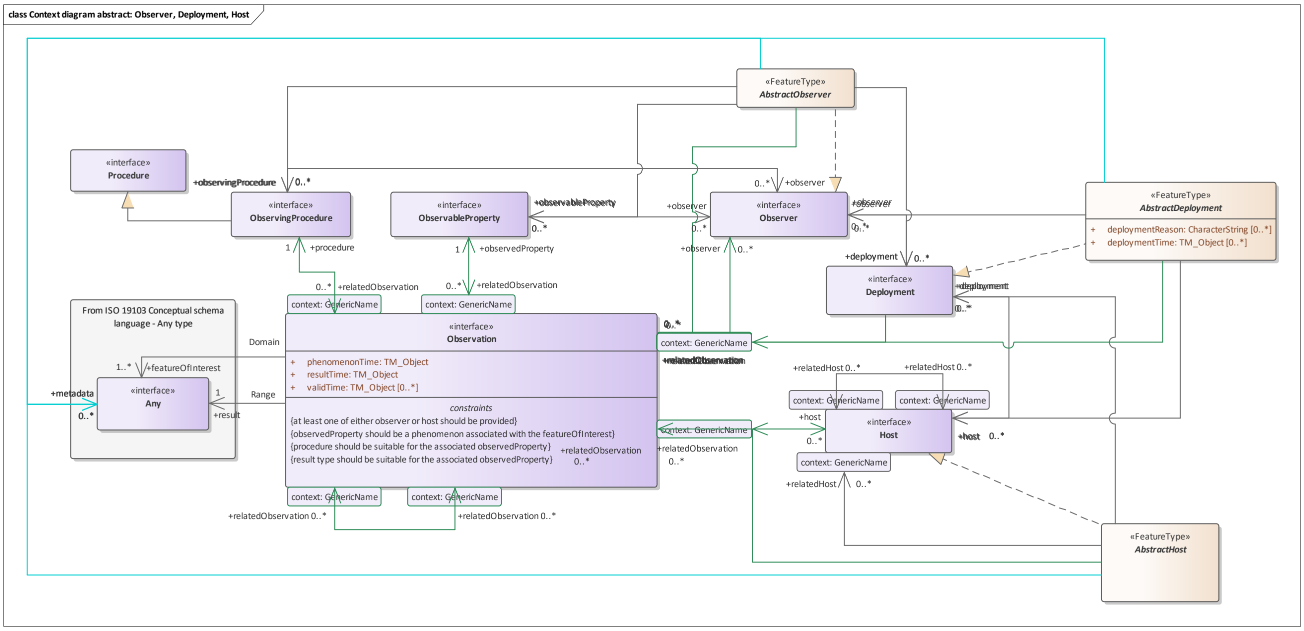

Figure 27 — Context diagram for Abstract Observation core — AbstractObserver, AbstractHost and AbstractDeployment

9.7 AbstractHost

9.7.1 AbstractHost Requirements Class

Requirements Class	/req/obs-core/AbstractHost
Target type	Logical model
Name	Abstract Observation core - AbstractHost
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Host
Requirement	/req/obs-core/gen/metadata-sem

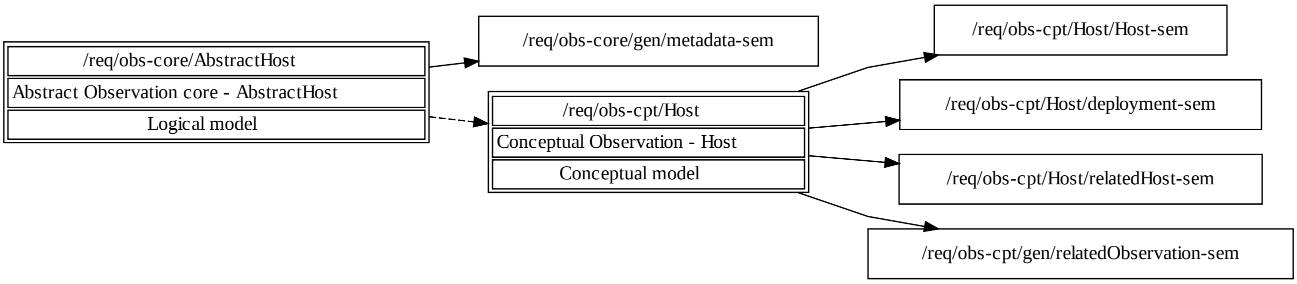

Figure 28 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractHost requirements class

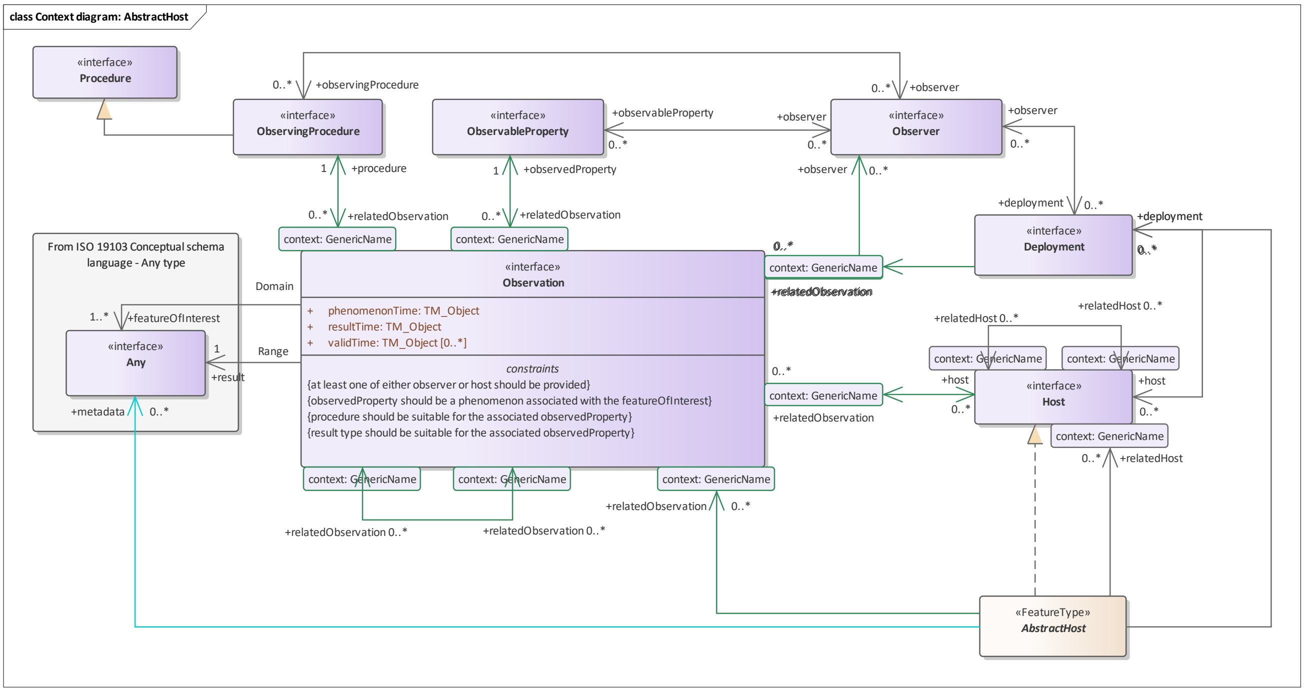

Figure 29 — Context diagram for Abstract Observation core — AbstractHost

9.8 AbstractDeployment

9.8.1 AbstractDeployment Requirements Class

Requirements Class	/req/obs-core/AbstractDeployment
Target type	Logical model
Name	Abstract Observation core - AbstractDeployment
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for

	data transfer conformance class
Imports	/req/obs-cpt/Deployment
Requirement	/req/obs-core/AbstractDeployment/deploymentReason-sem
Requirement	/req/obs-core/AbstractDeployment/deploymentTime-sem
Requirement	/req/obs-core/gen/metadata-sem

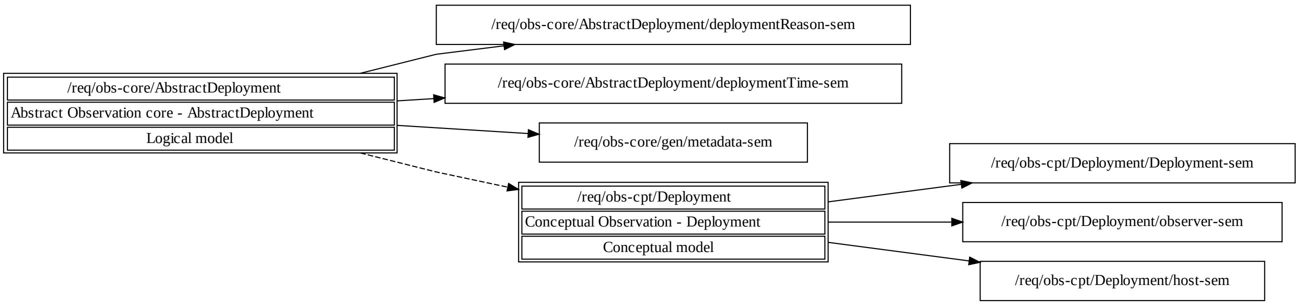

Figure 30 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — AbstractDeployment requirements class

9.8.2 Attribute deploymentReason

Requirement /req/obs-core/AbstractDeployment/deploymentReason-sem	A human readable description of the reason for the **Deployment**. If the reason for the **Deployment** is provided, the property ***deploymentReason:CharacterString*** SHALL be used.

EXAMPLES

— A researcher involved in a biodiversity survey campaign assessing the distribution of selected alien species. The deploymentReason describes the fact that this individual was involved in this campaign for the reason of identifying alien species.

— A sensor is mounted on a building to monitor seismic activities

— A new sensor type is rolled out within a regional or thematic network due to new legal reporting requirements.

9.8.3 Attribute deploymentTime

Requirement /req/obs-core/AbstractDeployment/deploymentTime-sem	The time that the **Deployment** pertains to. If the time of the **Deployment** is provided, property ***deploymentTime:TM_Period*** SHALL be used.

EXAMPLES

— A researcher involved in a biodiversity survey campaign assessing the distribution of selected alien species. The deploymentTime provides the time period(s) during which this person carried out this activity in the framework of the campaign.

— A sensor is mounted on a building to monitor seismic activities. The deploymentTime provides the time period(s) during which this sensor is mounted or active.

9.9 NamedValue

9.9.1 NamedValue Requirements Class

Requirements Class	/req/obs-core/NamedValue
Target type	Logical model
Name	Abstract Observation core - NamedValue
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Requirement	/req/obs-core/NamedValue/NamedValue-sem
Requirement	/req/obs-core/NamedValue/name-sem
Requirement	/req/obs-core/NamedValue/value-sem

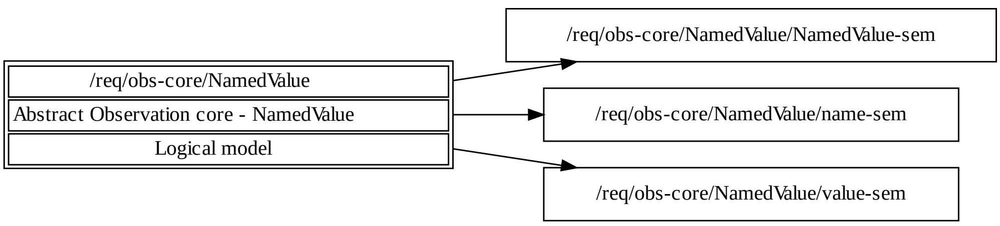

Figure 31 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Observation core — NamedValue requirements class

9.9.2 Data type NamedValue

Requirement /req/obs-core/NamedValue/NamedValue-sem	The class **NamedValue** provides for a generic soft-typed parameter value.

9.9.3 Attribute name

Requirement /req/obs-core/NamedValue/name-sem	The attribute **name:GenericName** SHALL indicate the meaning of the named value.

NOTE The value of the name should be taken from a well-governed source if possible.

EXAMPLE When used as the value of an Observation:parameter, the name might take values like 'procedureOperator', 'detectionLimit', 'amplifierGain', 'samplingDepth', 'analysisIteration', ...

9.9.4 Attribute value

Requirement /req/obs-core/NamedValue/value-sem	The attribute **value:Any** SHALL provide the value.

NOTE The type "Any" should be substituted by a suitable concrete type, such as CI_ResponsibleParty or Measure.

10 Basic Observations

10.1 General

10.1.1 Basic Observations Package Requirements Class

Requirements Class	/req/obs-basic
Target type	Logical model
Name	Basic Observations package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-basic/Observation
Imports	/req/obs-basic/ObservationCharacteristics
Imports	/req/obs-basic/ObservationCollection
Imports	/req/obs-basic/ObservingCapability
Imports	/req/obs-basic/ObservableProperty
Imports	/req/obs-basic/ObservingProcedure

Imports	/req/obs-basic/Observer
Imports	/req/obs-basic/Host
Imports	/req/obs-basic/Deployment
Imports	/req/obs-basic/GenericDomainFeature

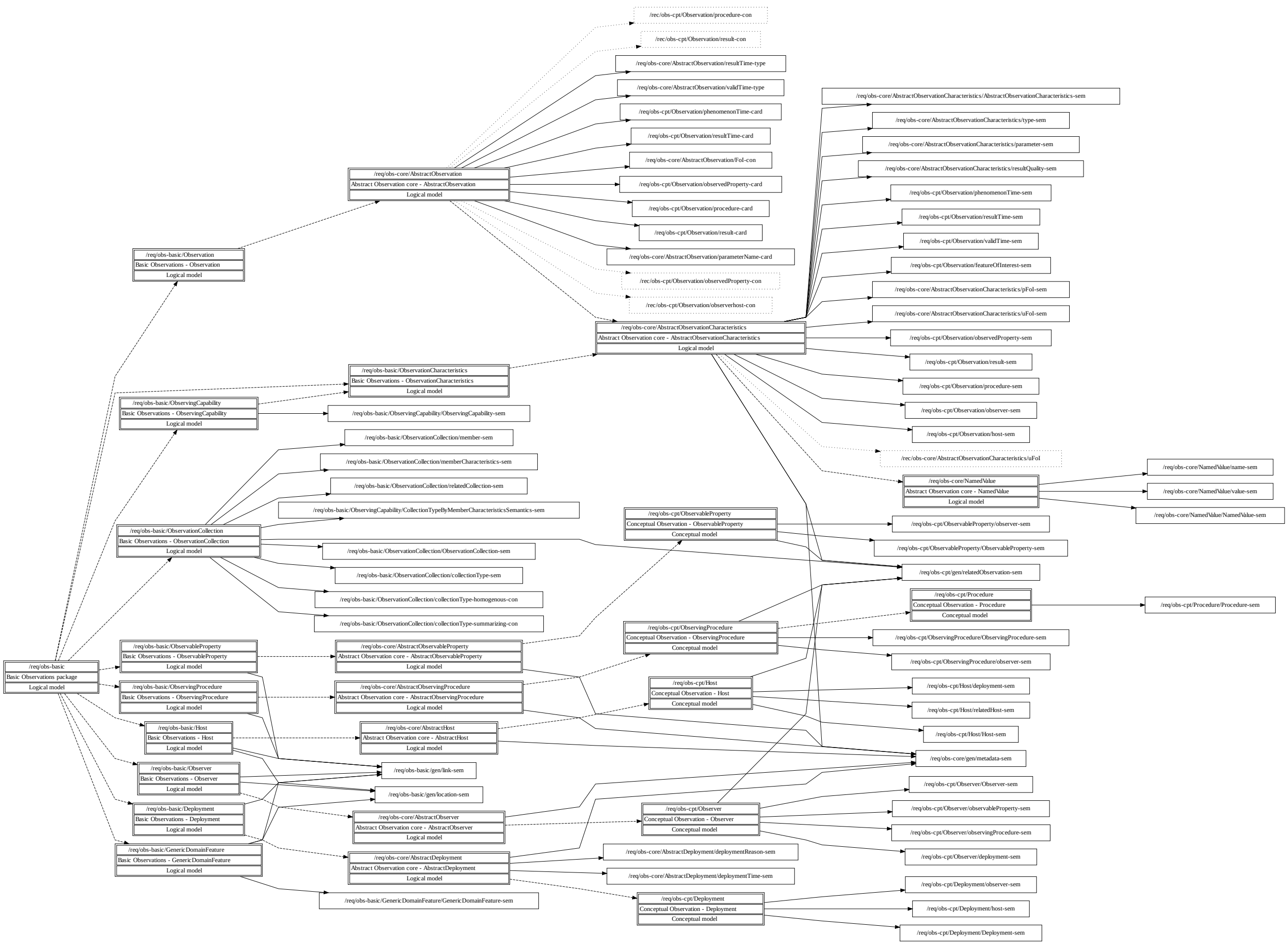

Figure 32 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations package requirements class

10.1.2 Attribute link

Requirement /req/obs-basic/gen/link-sem	Additional descriptive resources pertaining to a feature. If a link to a descriptive resource is provided, the attribute **link:URI** SHALL be used.

10.1.3 Attribute location

Requirement /req/obs-basic/gen/location-sem	Location information pertaining to a feature. If location information is provided, the attribute **location:Geometry** SHALL be used.

10.2 Observation

10.2.1 Observation Requirements Class

Requirements Class	/req/obs-basic/Observation
Target type	Logical model
Name	Basic Observations - Observation
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-core/AbstractObservation

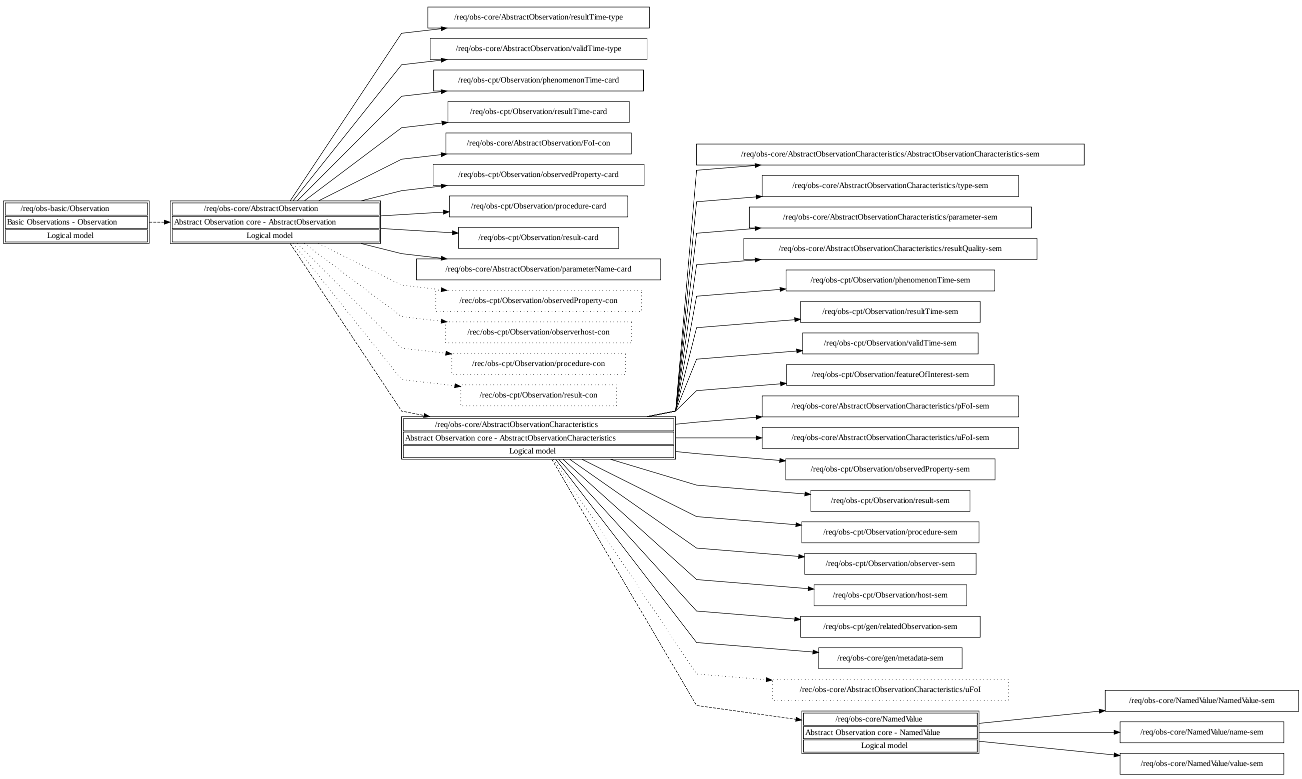

Figure 33 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — Observation requirements class

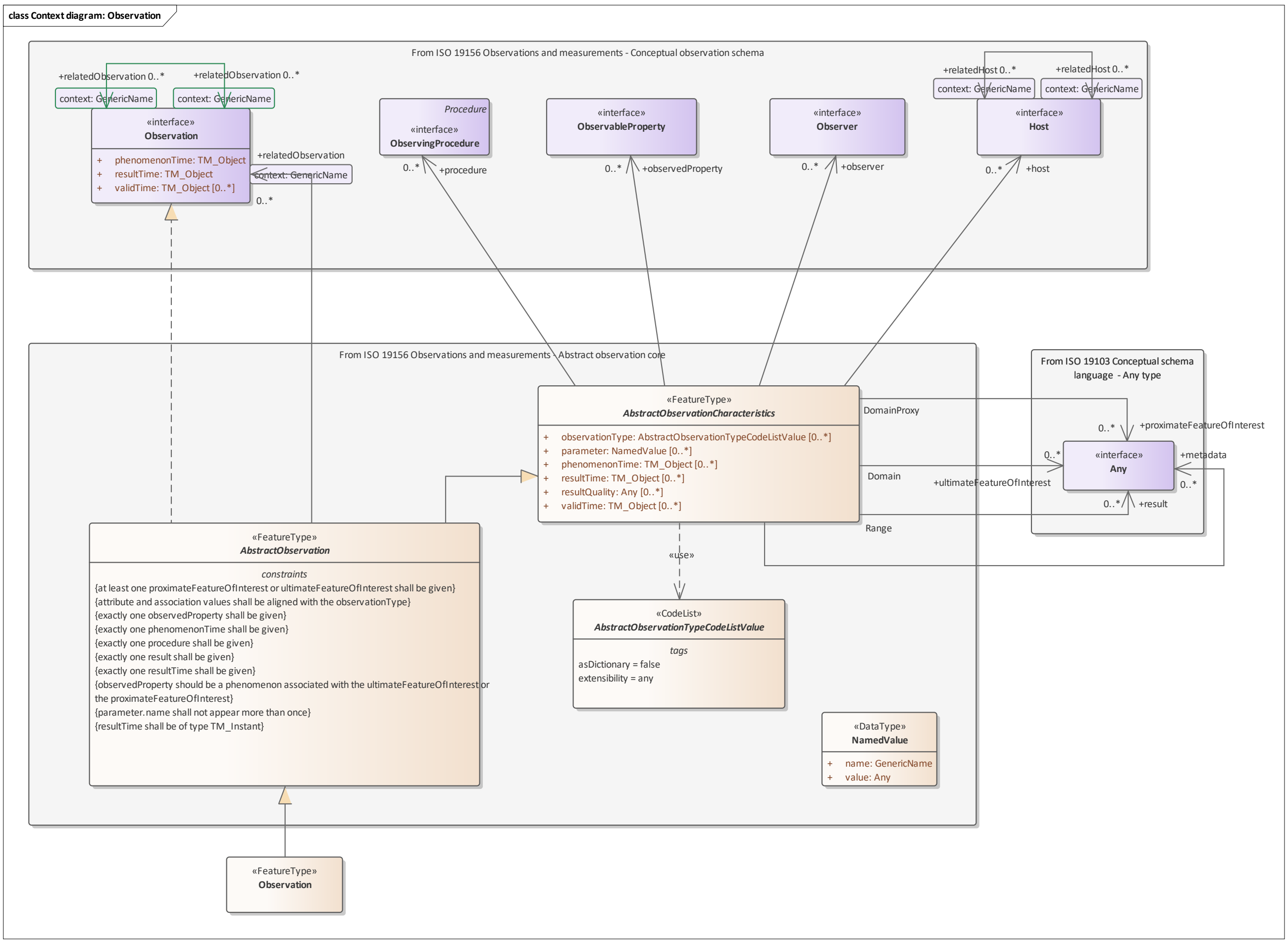

Figure 34 — Context diagram for Basic Observations — Observation

10.3 ObservationCharacteristics

10.3.1 ObservationCharacteristics Requirements Class

Requirements Class	/req/obs-basic/ObservationCharacteristics
Target type	Logical model
Name	Basic Observations - ObservationCharacteristics
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-core/AbstractObservationCharacteristics

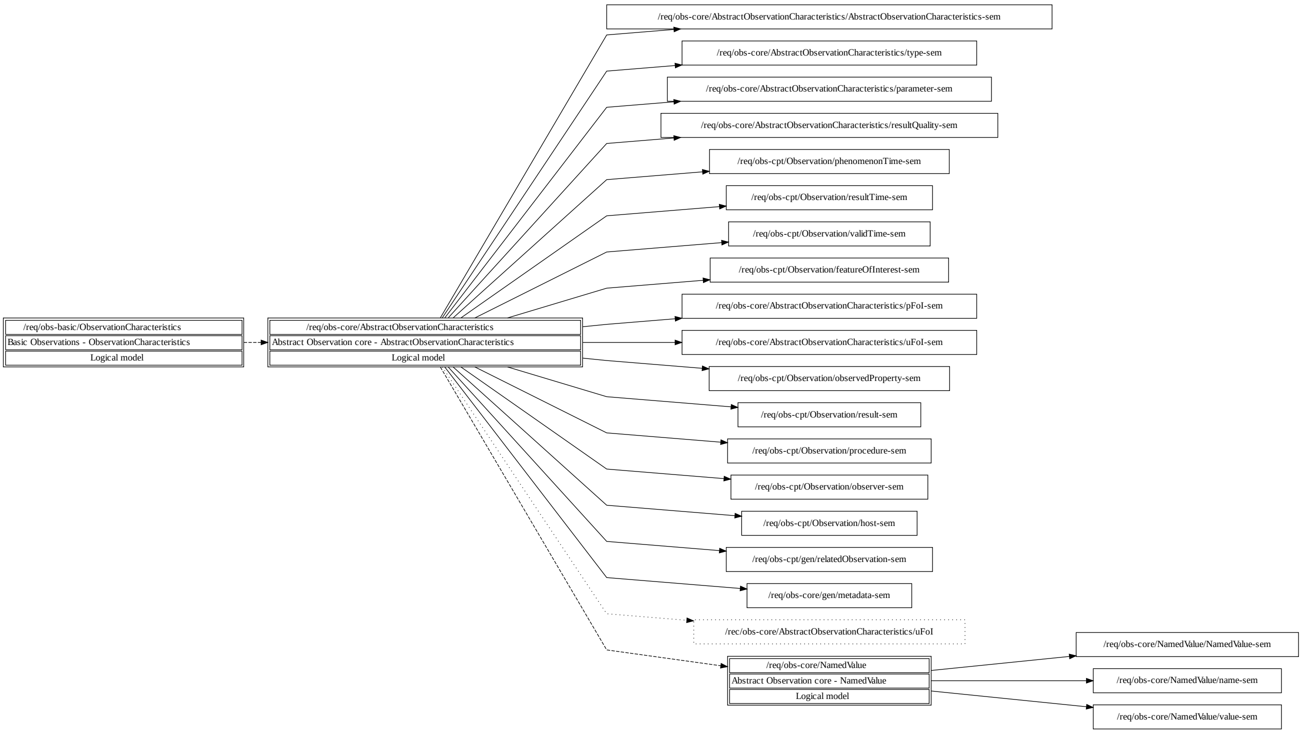

Figure 35 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — ObservationCharacteristics requirements class

10.4 ObservationCollection

10.4.1 ObservationCollection Requirements Class

Requirements Class	/req/obs-basic/ObservationCollection
Target type	Logical model
Name	Basic Observations - ObservationCollection
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/obs-basic/ObservationCollection/ObservationCollection-sem
Requirement	/req/obs-basic/ObservationCollection/collectionType-sem
Requirement	/req/obs-basic/ObservationCollection/collectionType-homogenous-con
Requirement	/req/obs-basic/ObservationCollection/collectionType-summarizing-con
Requirement	/req/obs-basic/ObservationCollection/member-sem
Requirement	/req/obs-basic/ObservationCollection/memberCharacteristics-sem

Requirement	/req/obs-basic/ObservationCollection/relatedCollection-sem
Requirement	/req/obs-basic/ObservingCapability/CollectionTypeByMemberCharacteristicsSemantics-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

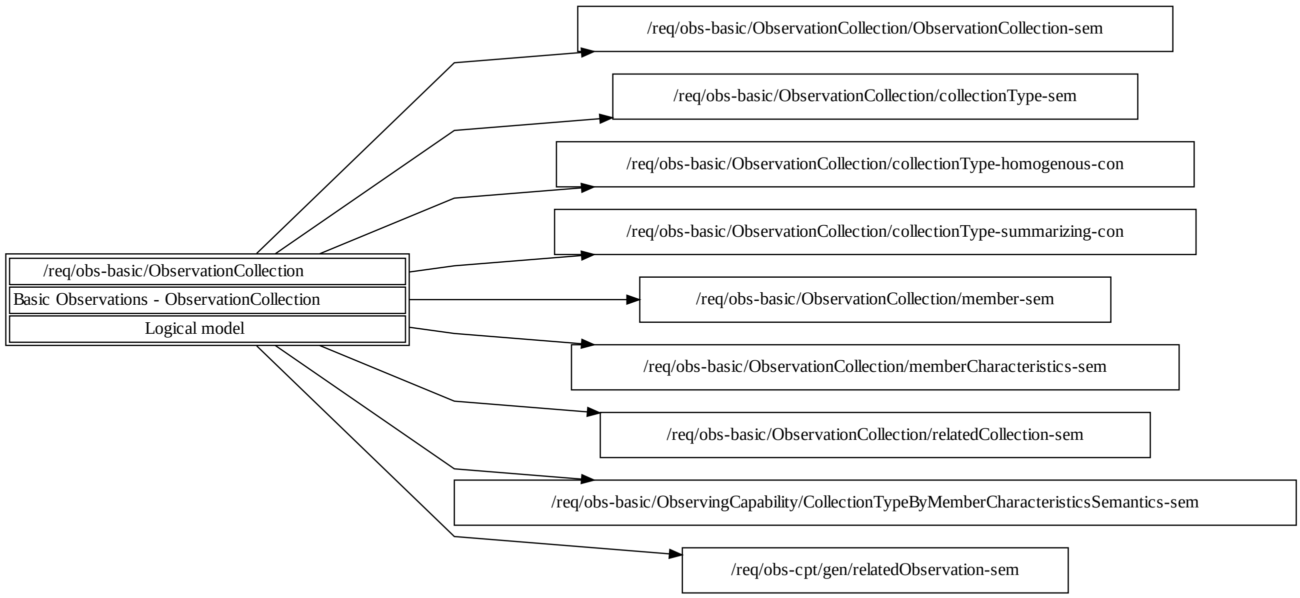

Figure 36 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — ObservationCollection requirements class

10.4.2 Feature type ObservationCollection

Requirement /req/obs-basic/ObservationCollection/ObservationCollection-sem	A collection of similar **Observation**s

10.4.3 Attribute collectionType

Requirement /req/obs-basic/ObservationCollection/collectionType-sem	Information on the type of the **ObservationCollection.** If information on the collection type is provided, the attribute **collectionType:AbstractObservationCollectionTypeCodeListValue** shall be used.
Requirement /req/obs-basic/ObservationCollection/collectionType-homogenous-con	If collectionType is specified homogenousObservationCollection as from the CollectionTypeByMemberCharacteristicsSemantics Codelist, the following constraints apply to the associated

	ObservationCharacteristics and all Observation instances referenced via the member association. If a property value is provided within the ObservationCharacteristics, this value applies to all contained observations (note: the observations need not contain this attribute as supplied via the ObservationCharacteristics): ● property not provided - values may be provided by the observations but is not provided at this level ● property provided but with no content - no observation within the collection provides this property ● property = value - this value applies to all observations within the collection ● property = value set/range - this value set/range applies to all observations within the collection

EXAMPLE 1 If the collection has the value “A” for property “foo” then all Observations in the collection have value “A” for that property.

EXAMPLE 2 If the collection states the ObservableProperty X, then all observations contained shall refer to that ObservableProperty.

Requirement /req/obs-basic/ObservationCollection/collectionType-summarizing-con	If **collectionType** is specified as **summarizingObservationCollection** from the **CollectionTypeByMemberCharacteristicsSemantics Codelist**, the following constraints apply to the associated **ObservationCharacteristics** and all **Observation** instances referenced via the **member** association. If multiple values for a property are available in the contained observations, ALL values for this attribute (or the range of values contained in all Observations) are provided in the ObservationCharacteristics. A property may also be empty in the ObservationCharacteristics - in this case any value can be provided for this attribute within the contained Observations: ● property not provided - values may be provided by the observations but a summary is not provided at this level ● property provided but with no content - no observation within the collection provides this property ● property = value - this value applies to all observations within the collection ● property = value set/range - all observations provide a value within this set/range

NOTE If a summarizing collection provides a set/range for an attribute it may be that all observations have this exact set/range as value for this attribute, or they could have different values that fall in the set/range.

EXAMPLE 1

If the summarizing collection supplies: phenomenonTime=2020-01-01T00:00:00Z/2020-02-01T00:00:00Z, validTime=[empty/NIL/null] and no other properties, this would mean that:

— Observations in the collection can have any value for the resultTime property, since it is absent from the collection.

— None of the Observations in the collection provide a value for validTime

— Observations can have any value for the phenomenonTime property that falls completely in the given time range. Valid examples would be:

  — 2020-01-05T00:00:00+05:00

  — 2020-01-05T10:00:00Z/2020-01-05T11:00:00Z

  — 2020-01-01T00:00:00Z/2020-02-01T00:00:00Z

EXAMPLE 2

If the summarizing collection supplies: result=1, this would mean that all the Observations in the collection have a value of 1 for the result property.

EXAMPLE 3

If the summarizing collection supplies: result=1, 2, 5, [8 - 11] (the values 1, 2 and 5, and the range 8-11), then examples of possible values for the result property on the contained Observations are:

— 1

— 9

— 2, 5 (a set with the two values)

— [8.1 - 9.2] (a range of 8.1 to 9.2)

— 1, 2, 5, [8 - 11] (the exact set of values from the collection)

EXAMPLE 4

If the summarizing collection supplies:

— ultimateFeatureOfInterest=https://example.org/collections/42/items/42,

— deployment=[empty/NIL/null] (i.e. property provided but with no content),

— observer=[https://example.org/v1.1/Sensors/41, https://example.org/v1.1/Sensors/43]

then this means:

a) the Observations in the collection all have the same ultimateFeatureOfInterest (a reference to https://example.org/collections/42/items/42),

b) none of the Observations in the collection have a (reference to a) deployment,

c) all Observations in the collection have either one, or both, of the referenced Observers.

d) Since the proximateFeatureOfInterest is not specified in the collection, the Observations in the collection can have any value for this field.

10.4.4 Association member

Requirement /req/obs-basic/ObservationCollection/member-sem	An **Observation** that is part of this **ObservationCollection**. If a reference to a member **Observation** is provided, the association with the role **member** SHALL be used.

10.4.5 Association memberCharacteristics

Requirement /req/obs-basic/ObservationCollection/memberCharacteristics-sem	Information on **ObservationCharacteristics** of **Observations** contained within the **ObservationCollection**. If a reference to **ObservationCharacteristics** pertaining to the collection members is provided, the association with the role **memberCharacteristics** SHALL be used.

10.4.6 Association relatedCollection

Requirement /req/obs-basic/ObservationCollection/relatedCollection-sem	A **ObservationCollection** the **ObservationCollection** is related to. If a reference to a related **ObservationCollection** is provided, the association with role **relatedCollection** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

10.5 ObservingCapability

10.5.1 ObservingCapability Requirements Class

Requirements Class	/req/obs-basic/ObservingCapability
Target type	Logical model
Name	Basic Observations - ObservingCapability
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class

Imports	/req/obs-basic/ObservationCharacteristics
Requirement	/req/obs-basic/ObservingCapability/ObservingCapability-sem

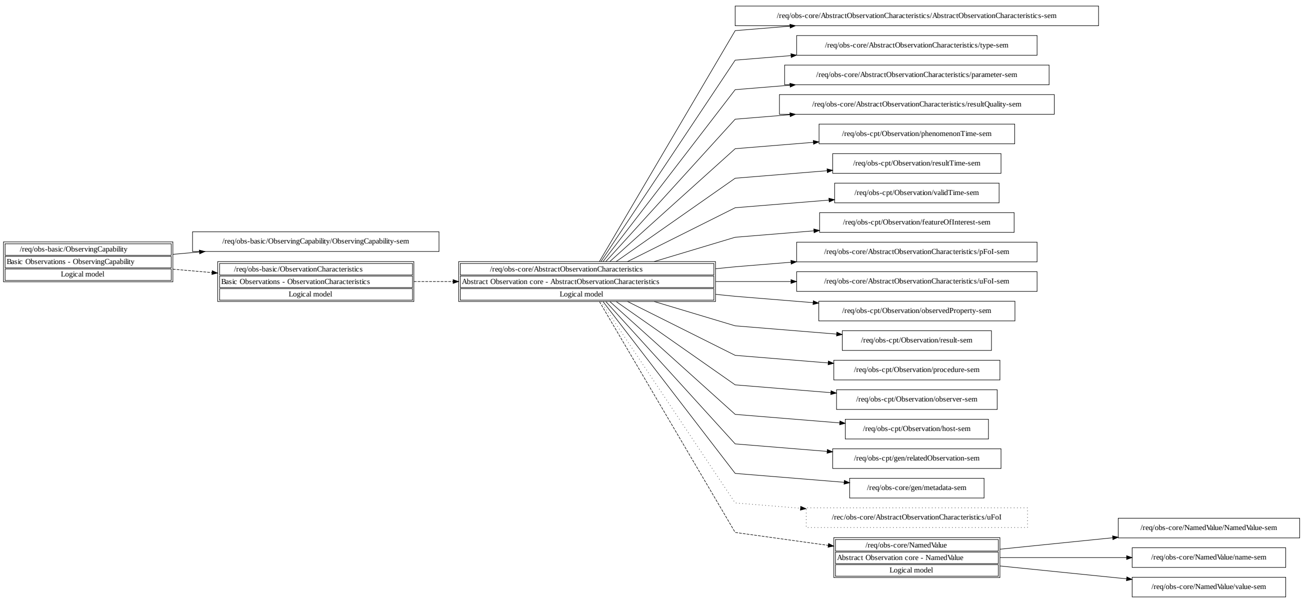

Figure 37 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — ObservingCapability requirements class

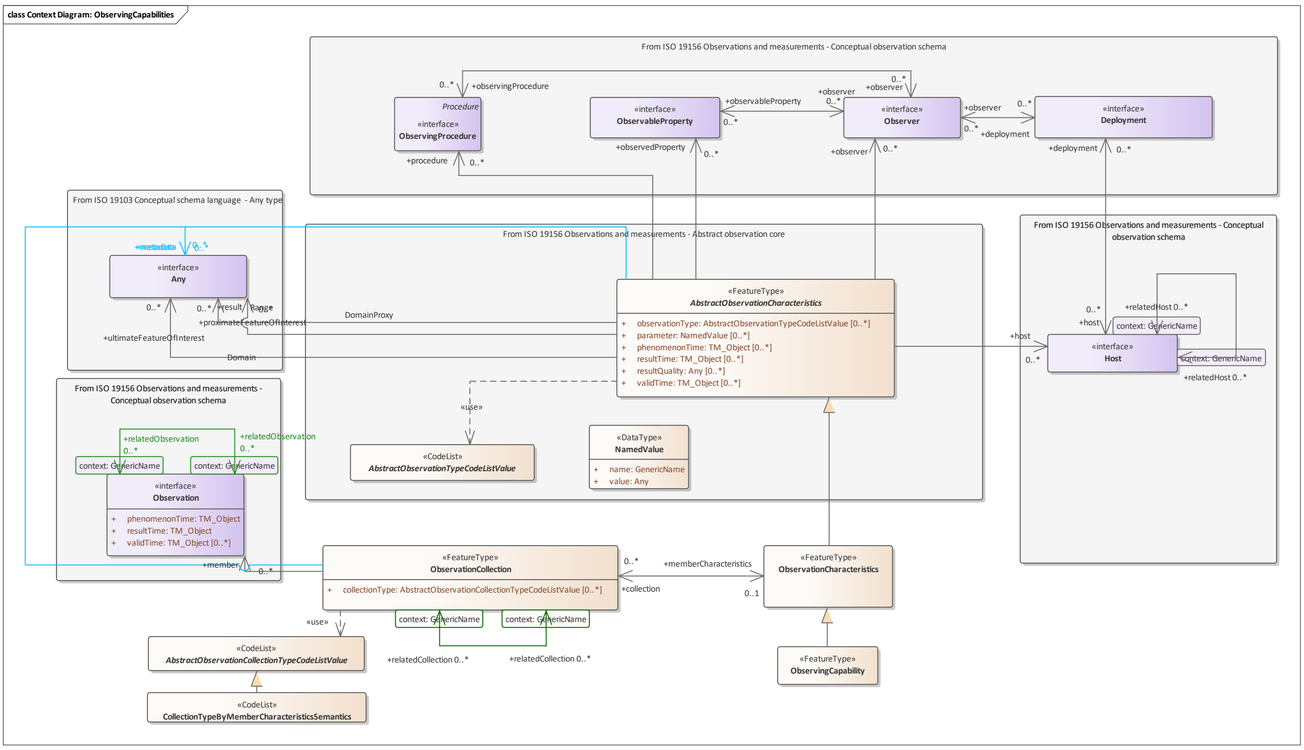

Figure 38 — Context diagram for Basic Observations — ObservingCapability and ObservationCollection

10.5.2 Feature type ObservingCapability

Requirement /req/obs-basic/ObservingCapability/ObservingCapability-sem	Information on **Observation**(s) that could potentially be provided.

EXAMPLE In order to explicitly describe the capabilities of an Environmental Monitoring Facility, one must provide information on what Observable Properties are being measured with which methodology.

For example in a national groundwater quantity monitoring network, depending on the equipment and the underlying observational strategies:

— some monitoring may have just one ObservingCapability: ultimateFeatureOfInterest:'Hydrogeological Unit 121AS', proximateFeatureOfInterest:'xyz', procedure: 'Groundwater depth measurement by electronic probe', observedProperty: 'GroundWaterDepth'

— some other, may have several such as

— ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05', proximateFeatureOfInterest: 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE', procedure: 'Groundwater depth measurement by electronic probe', observedProperty: 'GroundWaterDepth'

— ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05', proximateFeatureOfInterest: 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE', procedure: 'Digital recording teletransmitted', observedProperty: 'Water Temperature'

— ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05', proximateFeatureOfInterest: 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE', procedure: 'Digital recording teletransmitted', observedProperty: 'Water conductivity measured at 25°C'

NOTE In the example above, URIs have been removed and only the labels provided for better readability.

10.6 ObservableProperty

10.6.1 ObservableProperty Requirements Class

Requirements Class	/req/obs-basic/ObservableProperty
Target type	Logical model
Name	Basic Observations - ObservableProperty
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class

Imports	/req/obs-core/AbstractObservableProperty
Requirement	/req/obs-basic/gen/link-sem

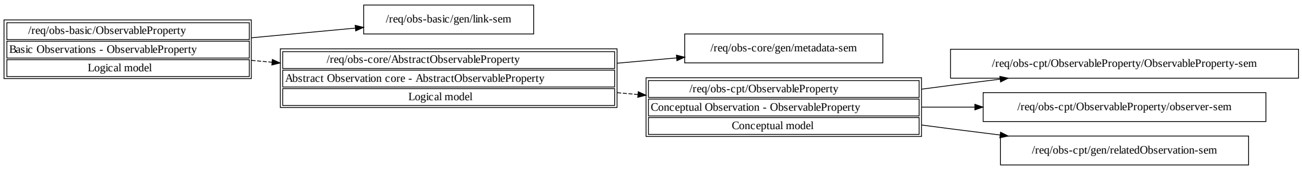

Figure 39 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — ObservableProperty requirements class

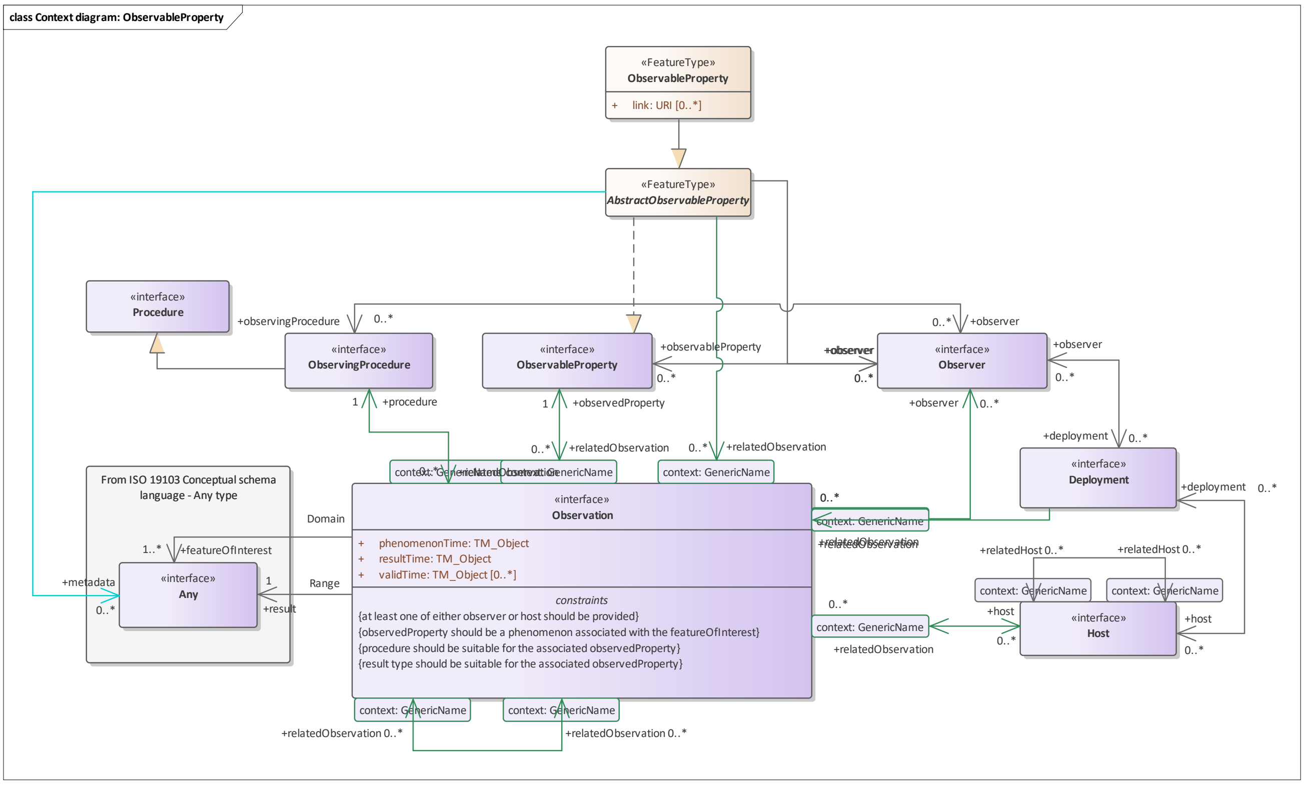

Figure 40 — Context diagram for the Basic Observations — ObservableProperty

10.7 ObservingProcedure

10.7.1 ObservingProcedure Requirements Class

Requirements Class	/req/obs-basic/ObservingProcedure
Target type	Logical model
Name	Basic Observations - ObservingProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2

	conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Imports	/req/obs-core/AbstractObservingProcedure
Requirement	/req/obs-basic/gen/link-sem

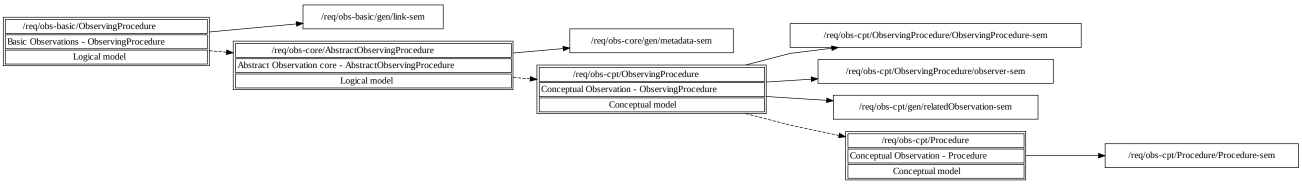

Figure 41 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — ObservingProcedure requirements class

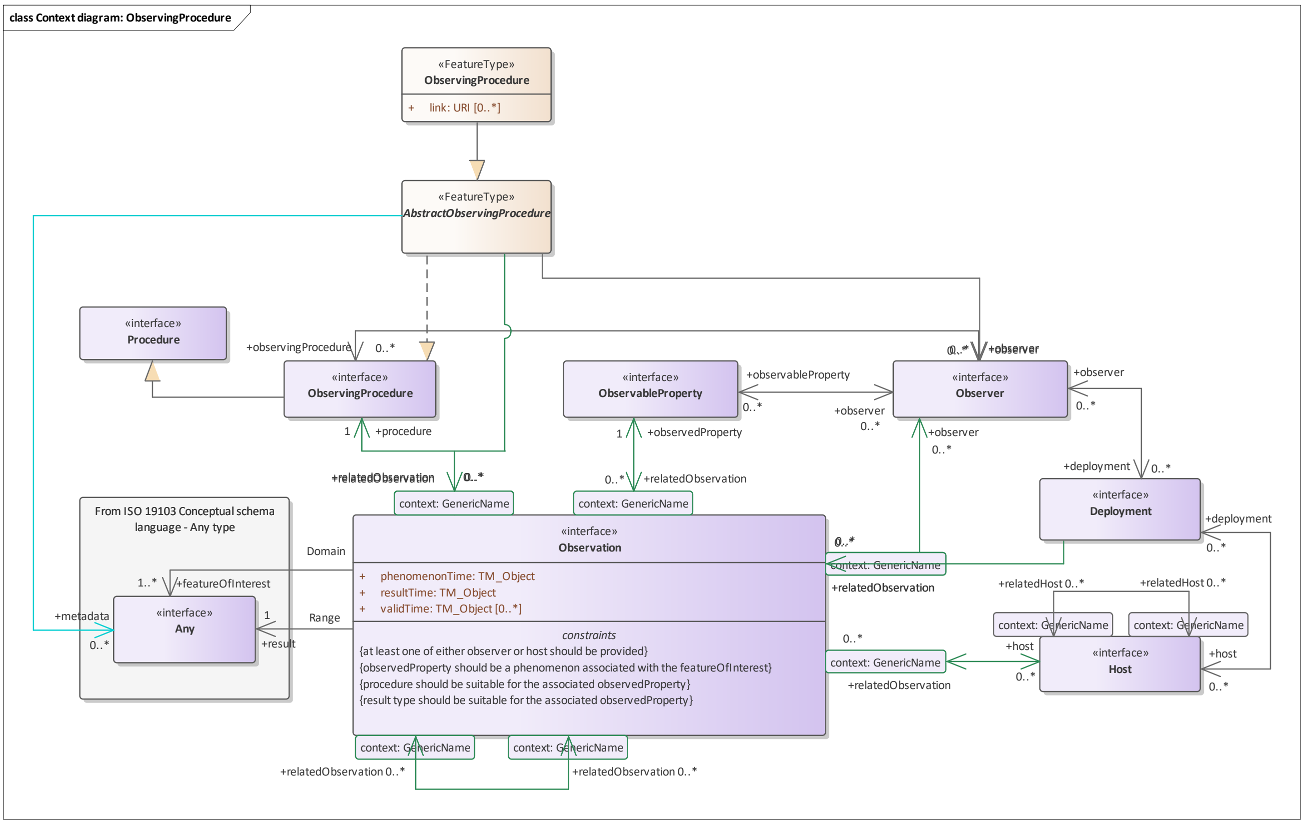

Figure 42 — Context diagram for Basic Observations — ObservingProcedure

10.8 Observer

10.8.1 Observer Requirements Class

Requirements Class	/req/obs-basic/Observer

Target type	Logical model
Name	Basic Observations - Observer
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Imports	/req/obs-core/AbstractObserver
Requirement	/req/obs-basic/gen/link-sem
Requirement	/req/obs-basic/gen/location-sem

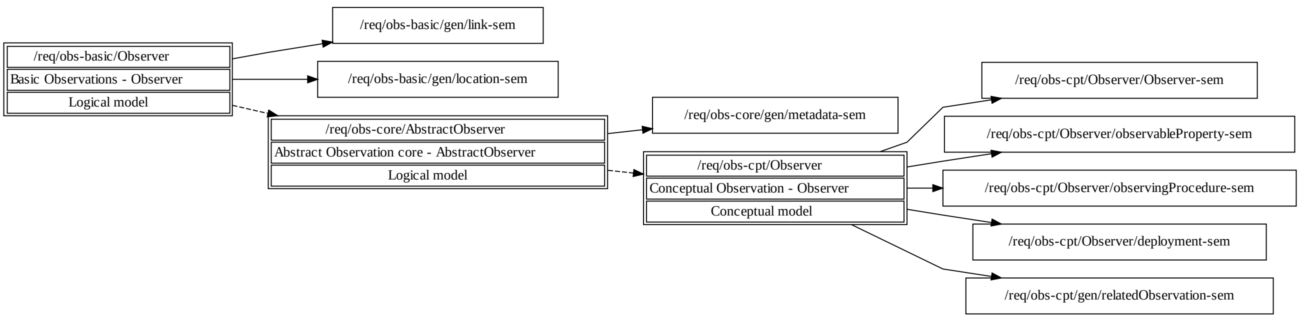

Figure 43 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — Observer requirements class

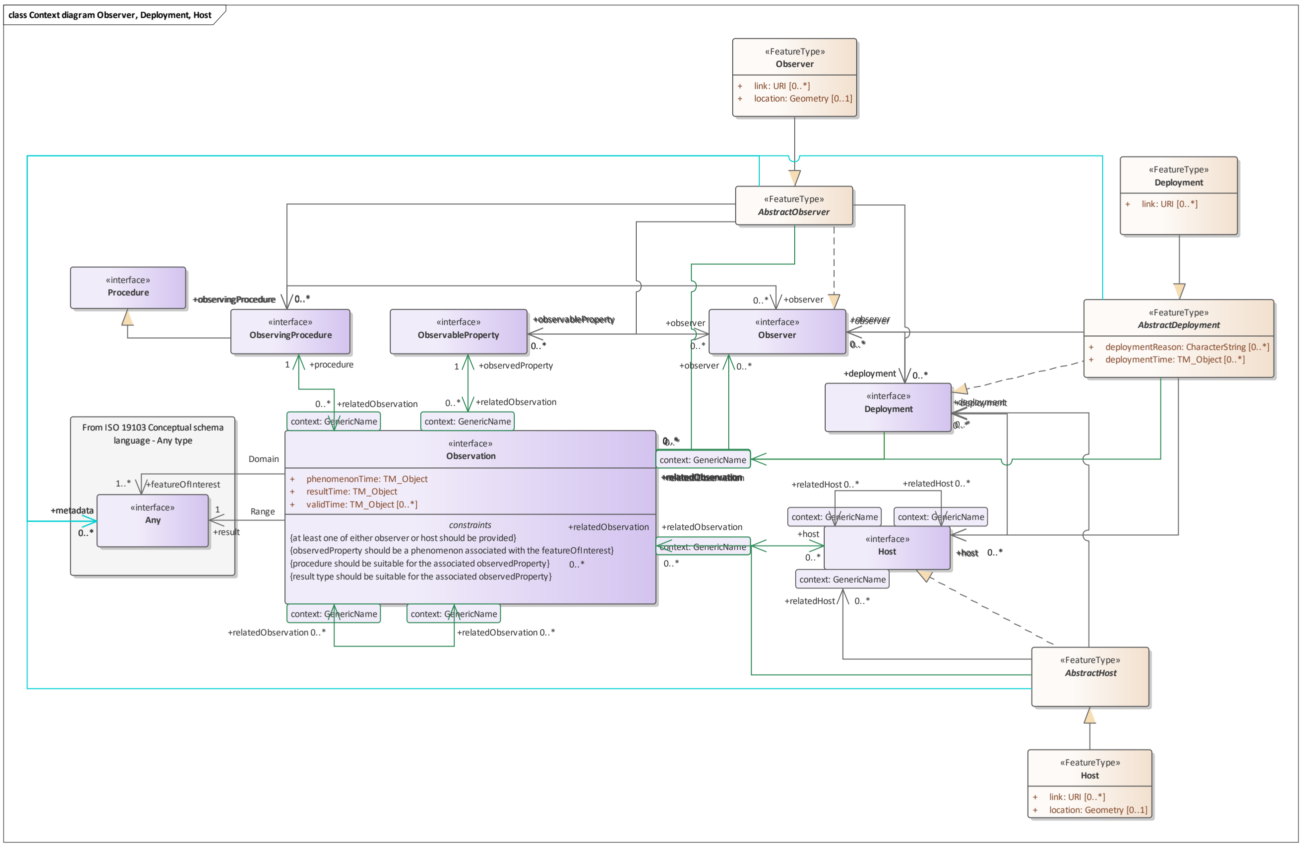

Figure 44 — Context diagram for Basic Observations — Observer, Host and Deployment

10.9 Host

10.9.1 Host Requirements Class

Requirements Class	/req/obs-basic/Host
Target type	Logical model
Name	Basic Observations - Host
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Imports	/req/obs-core/AbstractHost
Requirement	/req/obs-basic/gen/link-sem

Requirement	/req/obs-basic/gen/location-sem

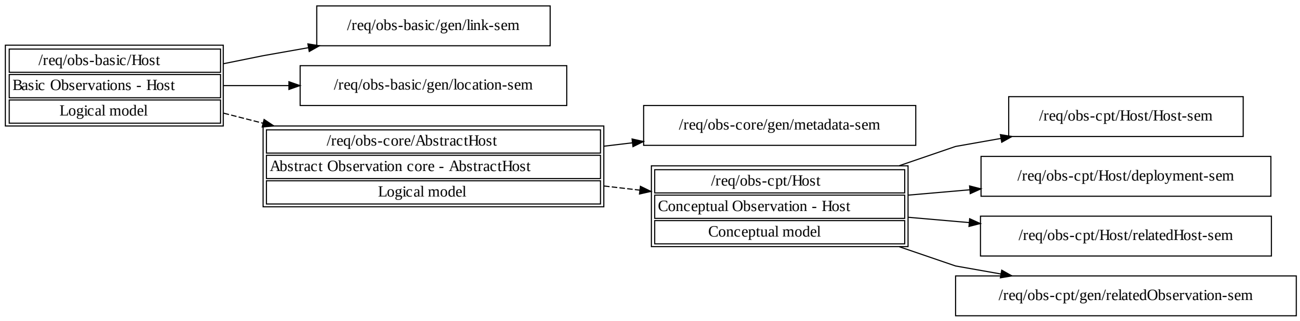

Figure 45 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — Host requirements class

10.10 Deployment

10.10.1 Deployment Requirements Class

Requirements Class	/req/obs-basic/Deployment
Target type	Logical model
Name	Basic Observations - Deployment
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Imports	/req/obs-core/AbstractDeployment
Requirement	/req/obs-basic/gen/link-sem

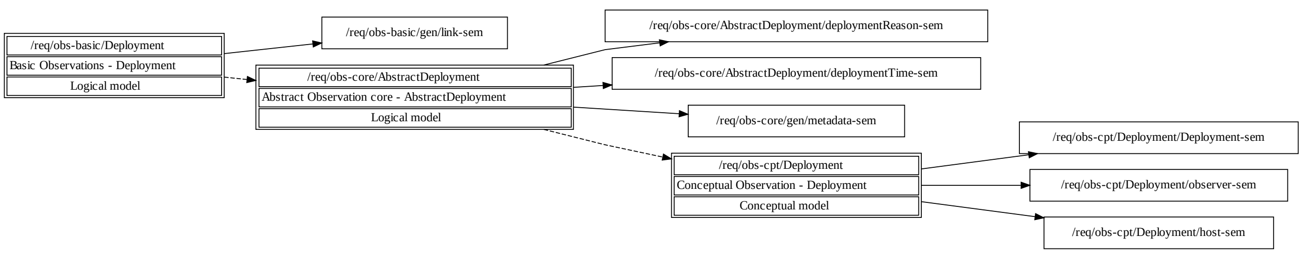

Figure 46 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — Deployment requirements class

10.11 GenericDomainFeature

10.11.1 GenericDomainFeature Requirements Class

Requirements Class	/req/obs-basic/GenericDomainFeature
Target type	Logical model
Name	Basic Observations - GenericDomainFeature
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Requirement	/req/obs-basic/GenericDomainFeature/GenericDomainFeature-sem
Requirement	/req/obs-basic/gen/link-sem
Requirement	/req/obs-basic/gen/location-sem

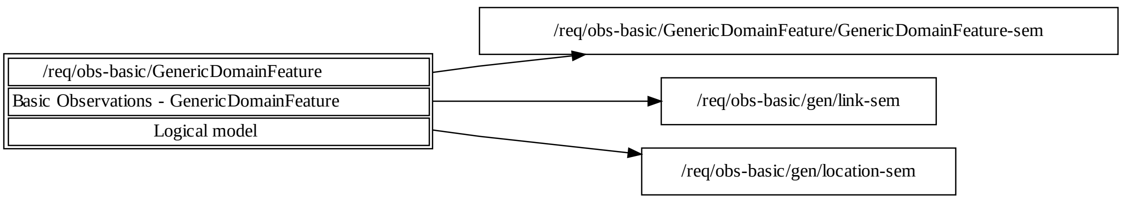

Figure 47 — (Informative) Included direct and indirect requirements and recommendations of the Basic Observations — GenericDomainFeature requirements class

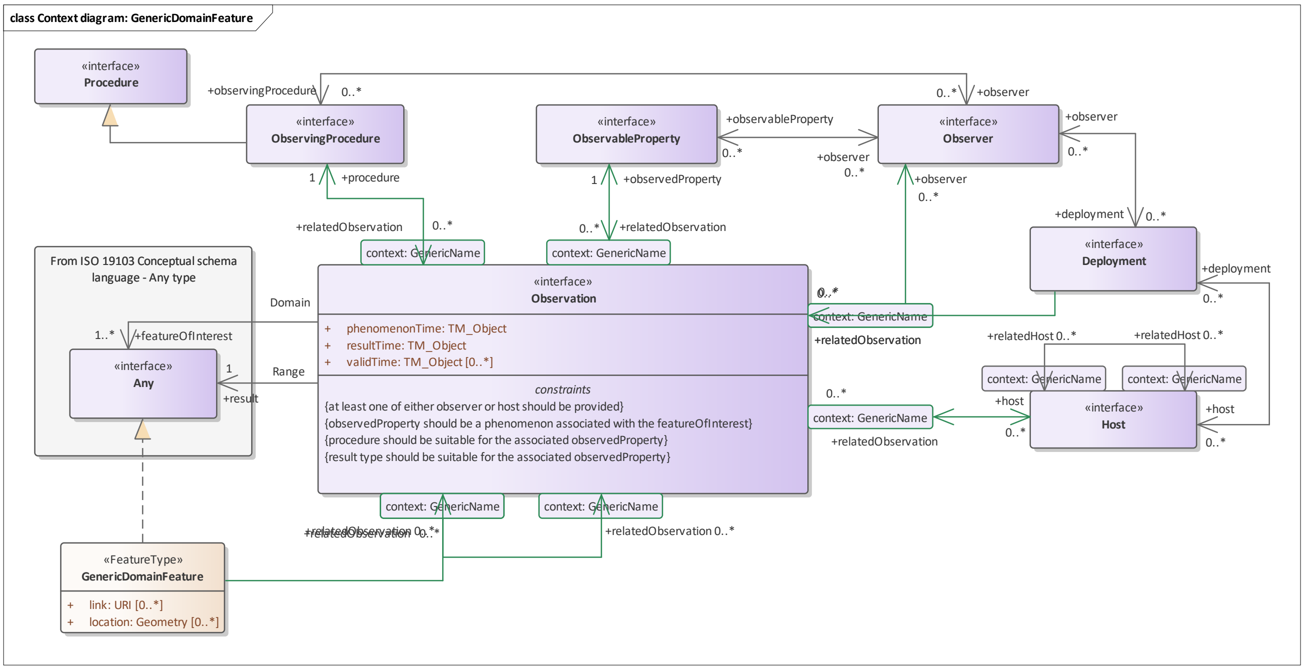

Figure 48 — Context diagram for Basic Observations — GenericDomainFeature

10.11.2 Feature type GenericDomainFeature

Requirement /req/obs-basic/GenericDomainFeature/GenericDomainFeature-sem	A concrete featureType to be utilized as **featureOfInterest** of an **Observation.**

NOTE This type is foreseen as a placeholder for specialized domain features in order to enable rapid prototyping.

10.12 Codelists

10.12.1 CollectionTypeByMemberCharacteristicsSemantics

Code lists has the following values defined in this International Standard: "homogenousObservationCollection" and "summarizingObservationCollection".

Requirement /req/obs-basic/ObservingCapability/CollectionTypeByMemberCharacteristicsSemantics-sem	The following entries SHALL be provided: ● homogenousObservationCollection: all observations contained are of a similar nature ● summarizingObservationCollection: a wider grab-bag type of collection

11 Conceptual Sample schema

11.1 General

11.1.1 Conceptual Sample schema model

The Conceptual Sample schema described as a class diagram in Figure 49. It is fully described in the Clause 11.1.2.

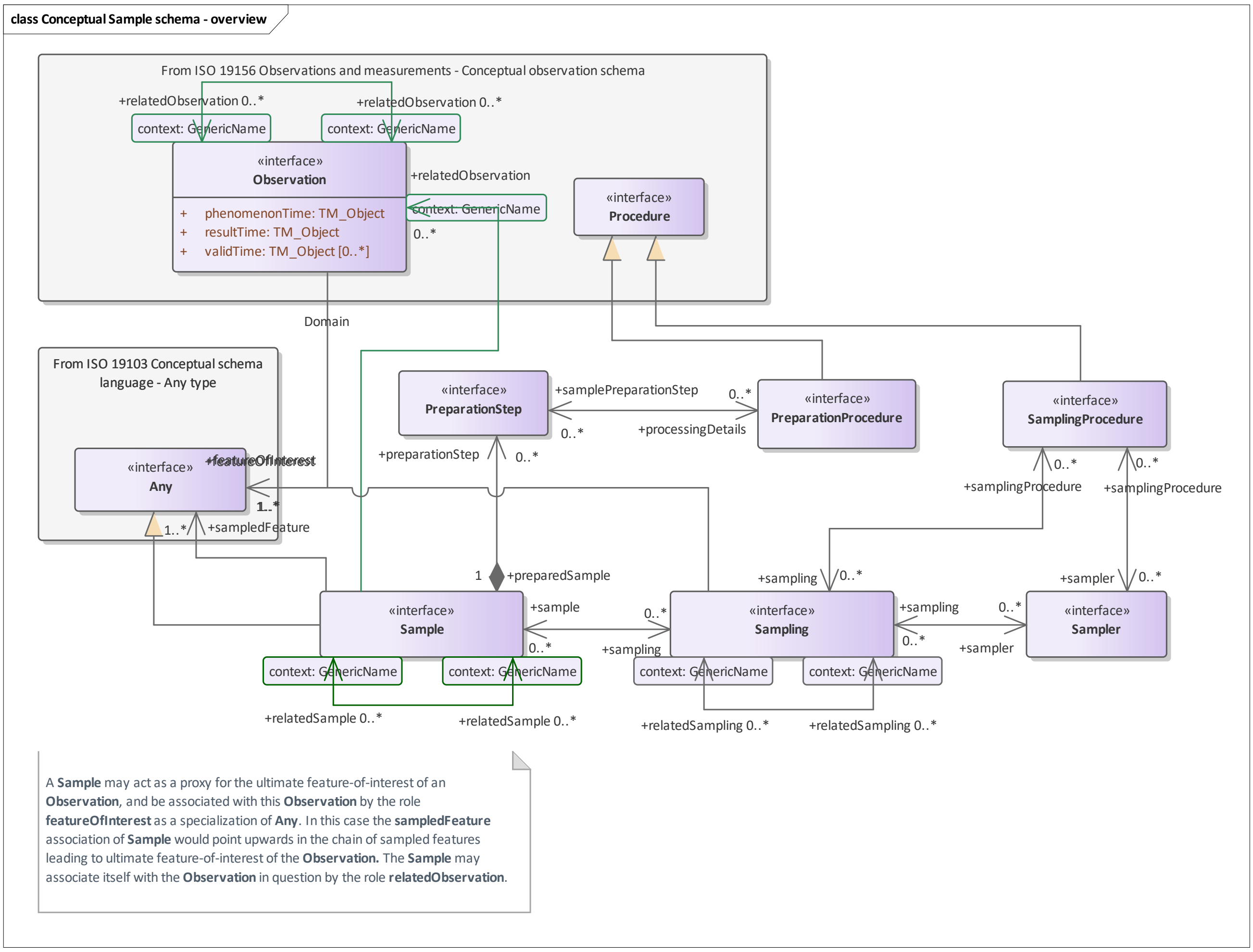

Figure 49 — Conceptual Sample schema overview

11.1.2 Conceptual Sample Schema Package Requirements Class

Requirements Class	/req/sam-cpt
Target type	Conceptual model
Name	Conceptual Sample schema package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class

Imports	/req/sam-cpt/Sample
Imports	/req/sam-cpt/Sampling
Imports	/req/sam-cpt/Sampler
Imports	/req/sam-cpt/PreparationStep
Imports	/req/sam-cpt/PreparationProcedure
Imports	/req/sam-cpt/SamplingProcedure

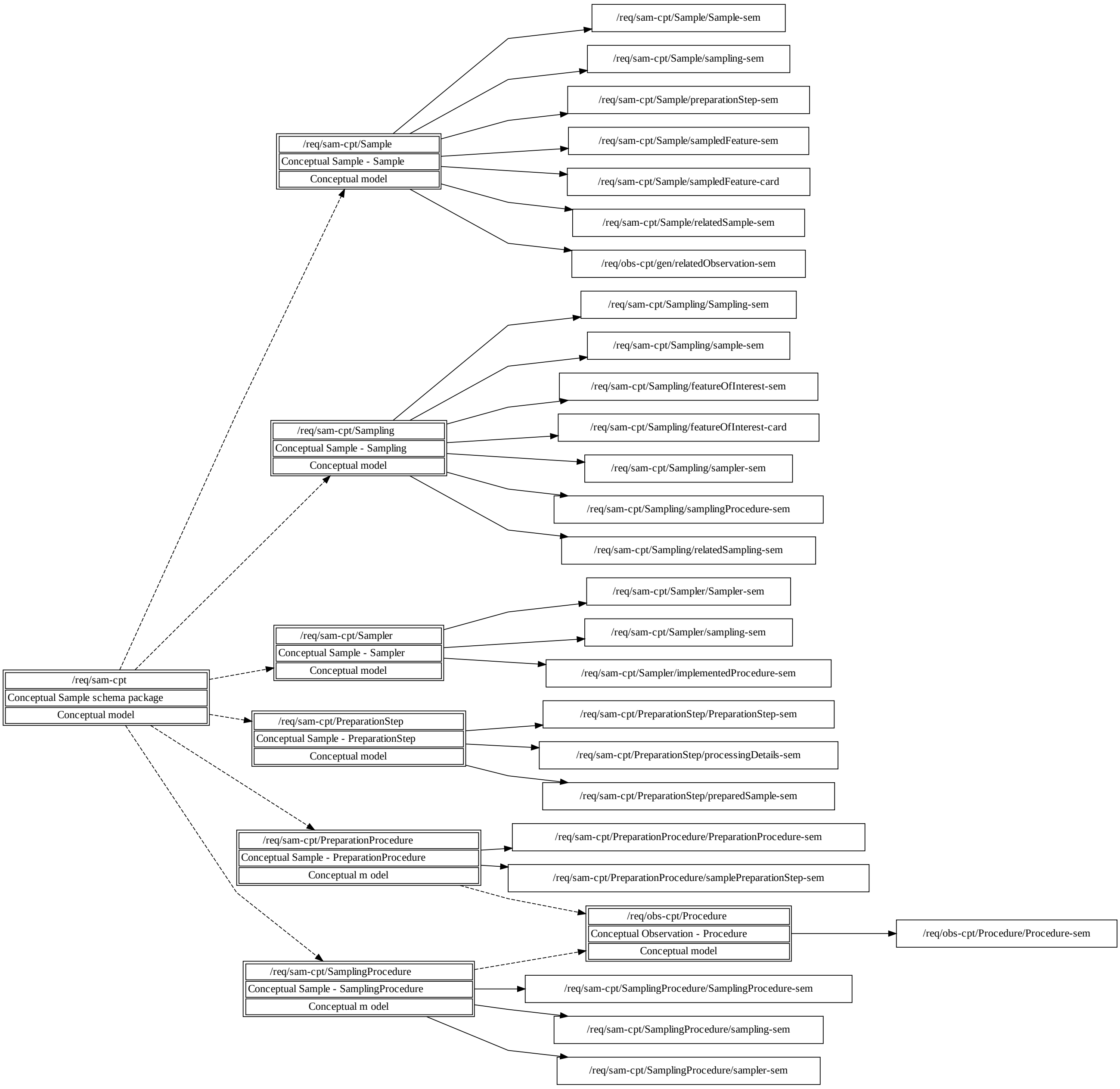

Figure 50 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema package requirements class

11.2 Sample

11.2.1 Sample Requirements Class

Requirements Class	/req/sam-cpt/Sample
Target type	Conceptual model
Name	Conceptual Sample - Sample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2

	conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Requirement	/req/sam-cpt/Sample/Sample-sem
Requirement	/req/sam-cpt/Sample/sampling-sem
Requirement	/req/sam-cpt/Sample/preparationStep-sem
Requirement	/req/sam-cpt/Sample/sampledFeature-sem
Requirement	/req/sam-cpt/Sample/sampledFeature-card
Requirement	/req/sam-cpt/Sample/relatedSample-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

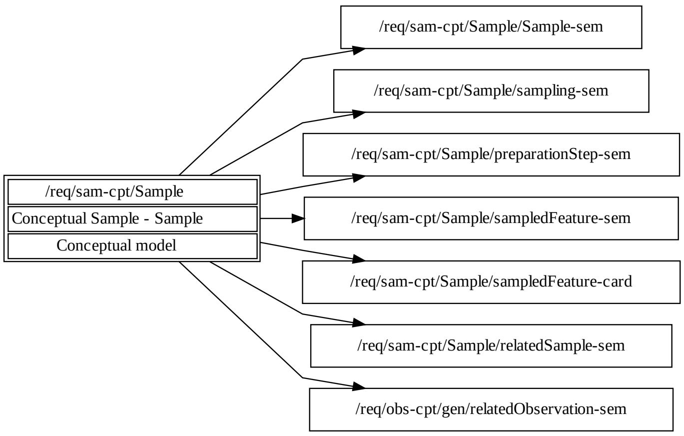

Figure 51 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — Sample requirements class

11.2.2 Interface Sample

Requirement /req/sam-cpt/Sample/Sample-sem	A **Sample** is an object that is representative of a concept, real-world object or phenomenon.

NOTE

— The way the sample is taken is typically guided by a sampling strategy. Sample are often artefacts of an observational strategy, and have often no significant function outside of their role in the observation process (although 'specimen preservation could be considered a specific activity per se').

— The physical characteristics of the features themselves are of little interest, except perhaps to the manager of a sampling campaign.

— Typically, the Sample is a Feature which is intended to be representative of a FeatureOfInterest on which Observations may be made. As such, it may carry a characteristic pertaining to the observedProperty being evaluated by the Observation.

EXAMPLE 1 A profile typically samples a water- or atmospheric-column; a well samples the water in an aquifer; a tissue specimen samples a part of an organism.

EXAMPLE 2 A statistical sample is often designed to be characteristic of an entire population, so that Observations can be made regarding the sample that provide a good estimate of the properties of the population.

11.2.3 Association sampling

Requirement /req/sam-cpt/Sample/sampling-sem	The **Sampling** the **Sample** is the result of. If **Sampling**(s) are described they SHALL be referred to using the association with the role **sampling**.

11.2.4 Association preparationStep

Requirement /req/sam-cpt/Sample/preparationStep-sem	The **PreparationStep**(s) applied to prepare the **Sample**. If **PreparationSteps** are described they SHALL be referred to using the association with the role **preparationStep**.

11.2.5 Association sampledFeature

Requirement /req/sam-cpt/Sample/sampledFeature-sem	The **sampledFeature** is the feature the **Sample** is intended to be representative of. References to the sampled feature SHALL be provided using the association with the role **sampledFeature**.

NOTE The sampled feature is usually a real-world feature from an application domain.

EXAMPLE 1 A profile typically samples a water or atmospheric column; a well samples the water in an aquifer; a tissue specimen samples a part of an organism.

EXAMPLE 2 A statistical sample is often designed to be characteristic of an entire population, so that Observations can be made regarding the sample that provide a good estimate of the properties of the population.

11.2.6 Association relatedSample

Requirement /req/sam-cpt/Sample/relatedSample-sem	A **Sample** the **Sample** is related to. If a reference to a related **Sample** is provided, the association with role **relatedSample** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

NOTE Sample are frequently related to each other, as parts of complexes, and in other ways.

EXAMPLE Sampling points are often located along a sampling curve; material sample are usually obtained from a sampling point; pixels are part of a scene; stations are often part of an array.

11.3 Sampling

11.3.1 Sampling Requirements Class

Requirements Class	/req/sam-cpt/Sampling
Target type	Conceptual model
Name	Conceptual Sample - Sampling
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Requirement	/req/sam-cpt/Sampling/Sampling-sem
Requirement	/req/sam-cpt/Sampling/sample-sem
Requirement	/req/sam-cpt/Sampling/featureOfInterest-sem
Requirement	/req/sam-cpt/Sampling/featureOfInterest-card
Requirement	/req/sam-cpt/Sampling/sampler-sem
Requirement	/req/sam-cpt/Sampling/samplingProcedure-sem
Requirement	/req/sam-cpt/Sampling/relatedSampling-sem

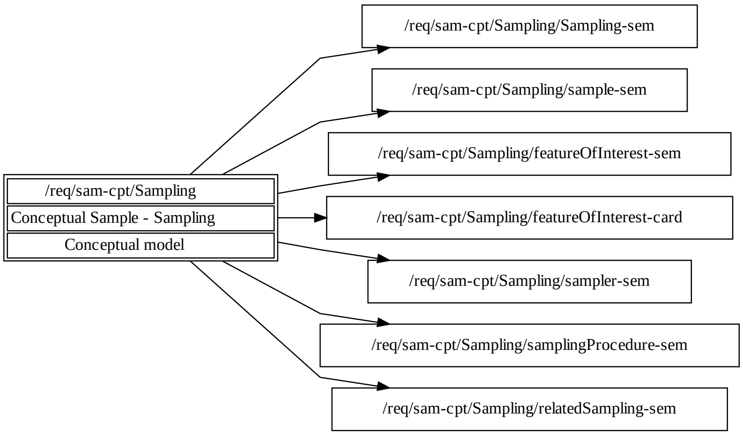

Figure 52 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — Sampling requirements class

11.3.2 Interface Sampling

Requirement /req/sam-cpt/Sampling/Sampling-sem	**Sampling** is an act applying a **SamplingProcedure** to create or transform one or more **Sample**(s).

EXAMPLES

— Crushing a rock sample in a ball mill,

— Digging a pit through a soil sequence,

— Dividing a field site into quadrants,

— Drawing blood from a patient,

— Extracting water from an observation well,

— Extracting a sample from a defined environmental monitoring station,

— Registering an image of the landscape,

— Sieving a powder to separate the subset finer than 100-mesh,

— Selecting a subset of a population,

— Splitting a piece of drill-core to create two new samples,

— Taking a diamond-drill core from a rock outcrop.

11.3.3 Association sample

Requirement /req/sam-cpt/Sampling/sample-sem	The **Sample** generated by the **Sampling.** If **Samples** are described they SHALL be referred to using the association with the role **sample**.

11.3.4 Association featureOfInterest

Requirement /req/sam-cpt/Sampling/featureOfInterest-sem	The concept, real-world object or phenomenon (feature-of-interest) the **Sample**(s) of the **Sampling** represent.

Requirement /req/sam-cpt/Sampling/featureOfInterest-card	Reference to the feature-of-interest SHALL be done using the association with the role **featureOfInterest.**

11.3.5 Association sampler

Requirement /req/sam-cpt/Sampling/sampler-sem	The **Sampler** that performed the **Sampling**. If **Sampler**(s) are described they SHALL be referred to using the association with the role **sampler**.

11.3.6 Association samplingProcedure

Requirement /req/sam-cpt/Sampling/samplingProcedure-sem	The **SamplingProcedure** used by the **Sampling.** If **SamplingProcedures** are described they SHALL be referred to using the association with the role **samplingProcedure**.

11.3.7 Association relatedSampling

Requirement /req/sam-cpt/Sampling/relatedSampling-sem	Related **Sampling**(s). If a reference to a related **Sampling** is provided, the association with role **relatedSampling** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

11.4 Sampler

11.4.1 Sampler Requirements Class

Requirements Class	/req/sam-cpt/Sampler
Target type	Conceptual model
Name	Conceptual Sample - Sampler

Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/sam-cpt/Sampler/Sampler-sem
Requirement	/req/sam-cpt/Sampler/sampling-sem
Requirement	/req/sam-cpt/Sampler/implementedProcedure-sem

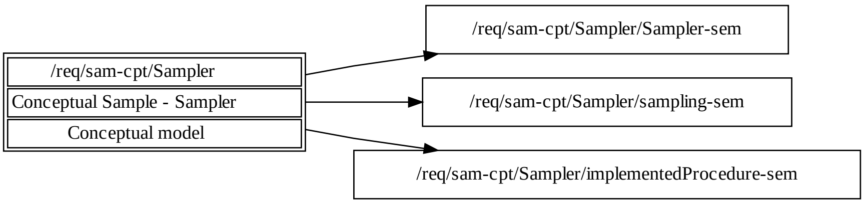

Figure 53 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — Sampler requirements class

11.4.2 Interface Sampler

Requirement /req/sam-cpt/Sampler/Sampler-sem	A **Sampler** is a device or entity (including humans) that is used by, or implements, a **SamplingProcedure** to create or transform one or more **Sample**(s).

EXAMPLES

— A ball mill, diamond drill, hammer,

— hypodermic syringe and needle,

— image sensor, a soil auger,

— a human being.

NOTE All the examples above can act as sampling devices (i.e., be Samplers). However, sometimes the distinction between the Sampler and the Sensor is not evident, as they are packaged as a unit. A Sampler need not be a physical device.

11.4.3 Association sampling

Requirement /req/sam-cpt/Sampler/sampling-sem	The **Sampling** act performed by the **Sampler**. If **Sampling**(s) are described they SHALL be referred to using the association with the role **sampling**.

11.4.4 Association implementedProcedure

Requirement /req/sam-cpt/Sampler/implementedProcedure-sem	The **Procedure** implemented by the **Sampler**. If **Procedure**(s) are described they SHALL be referred to using the association with the role **implementedProcedure**.

11.5 PreparationStep

11.5.1 PreparationStep Requirements Class

Requirements Class	/req/sam-cpt/PreparationStep
Target type	Conceptual model
Name	Conceptual Sample - PreparationStep
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Requirement	/req/sam-cpt/PreparationStep/PreparationStep-sem
Requirement	/req/sam-cpt/PreparationStep/processingDetails-sem
Requirement	/req/sam-cpt/PreparationStep/preparedSample-sem

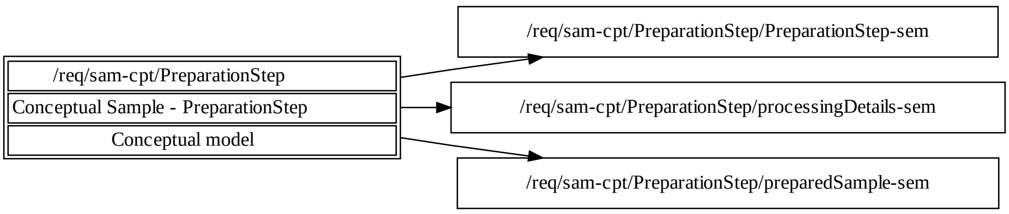

Figure 54 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — PreparationStep requirements class

11.5.2 Interface PreparationStep

Requirement /req/sam-cpt/PreparationStep/PreparationStep-sem	A **PreparationStep** is an individual step pertaining to a **PreparationProcedure**.

11.5.3 Association processingDetails

Requirement /req/sam-cpt/PreparationStep/processingDetails-sem	A **PreparationProcedure** step performed on the **Sample** the **PreparationStep** pertains to. If **PreparationProcedure**(s) are described they SHALL be referred to using the association with the role **processingDetails**.

11.5.4 Association preparedSample

Requirement /req/sam-cpt/PreparationStep/preparedSample-sem	The **Sample** on which the **PreparationProcedure** is performed. The **Sample** SHALL be referred to using the association with the role **preparedSample**.

11.6 PreparationProcedure

11.6.1 PreparationProcedure Requirements Class

Requirements Class	/req/sam-cpt/PreparationProcedure
Target type	Conceptual model
Name	Conceptual Sample - PreparationProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Procedure
Requirement	/req/sam-cpt/PreparationProcedure/PreparationProcedure-sem
Requirement	/req/sam-cpt/PreparationProcedure/samplePreparationStep-sem

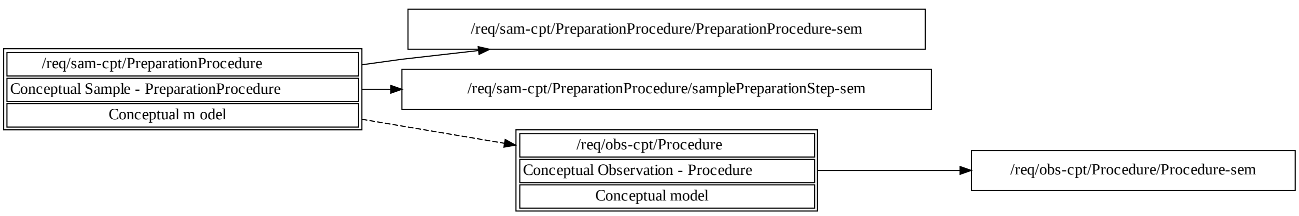

Figure 55 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — PreparationProcedure requirements class

11.6.2 Interface PreparationProcedure

Requirement /req/sam-cpt/PreparationProcedure/PreparationProcedure-sem	The description of preparation steps performed on a **Sample**.

11.6.3 Association samplePreparationStep

Requirement /req/sam-cpt/PreparationProcedure/samplePreparationStep-sem	If the **PreparingProcedure** provides information on the **PreparationStep** where this procedure has been used, the association with the role **samplePreparationStep** SHALL be used.

11.7 SamplingProcedure

11.7.1 SamplingProcedure Requirements Class

Requirements Class	/req/sam-cpt/SamplingProcedure
Target type	Conceptual model
Name	Conceptual Sample - SamplingProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/obs-cpt/Procedure
Requirement	/req/sam-cpt/SamplingProcedure/SamplingProcedure-sem
Requirement	/req/sam-cpt/SamplingProcedure/sampling-sem
Requirement	/req/sam-cpt/SamplingProcedure/sampler-sem

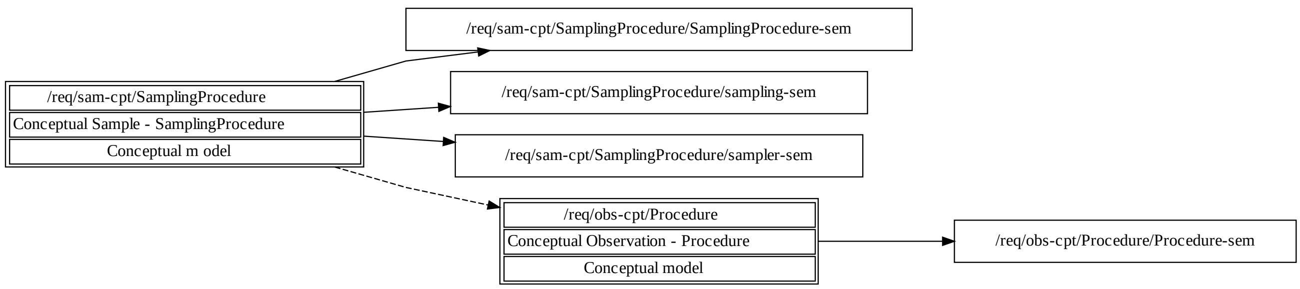

Figure 56 — (Informative) Included direct and indirect requirements and recommendations of the Conceptual Sample schema — SamplingProcedure requirements class

11.7.2 Interface SamplingProcedure

Requirement /req/sam-cpt/SamplingProcedure/SamplingProcedure-sem	The description of steps performed by a **Sampler** in order to extract a **Sample** from its **sampledFeature** in the frame of a **Sampling.**

11.7.3 Association sampling

Requirement /req/sam-cpt/SamplingProcedure/sampling-sem	If the **SamplingProcedure** provides information on the **Sampling** where this procedure has been used, the association with the role **sampling** SHALL be used.

11.7.4 Association sampler

Requirement /req/sam-cpt/SamplingProcedure/sampler-sem	If the **SamplingProcedure** provides information on the **Sampler** that implements this procedure has been used, the association with the role **sampler** SHALL be used.

12 Abstract Sample Core

12.1 General

12.1.1 Abstract Sample Core Package Requirements

Requirements Class	/req/sam-core
Target type	Logical model
Name	Abstract Sample core package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-core/AbstractSample
Imports	/req/sam-core/AbstractSampling
Imports	/req/sam-core/AbstractSampler
Imports	/req/sam-core/AbstractSamplingProcedure
Imports	/req/sam-core/AbstractPreparationProcedure
Imports	/req/sam-core/AbstractPreparationStep

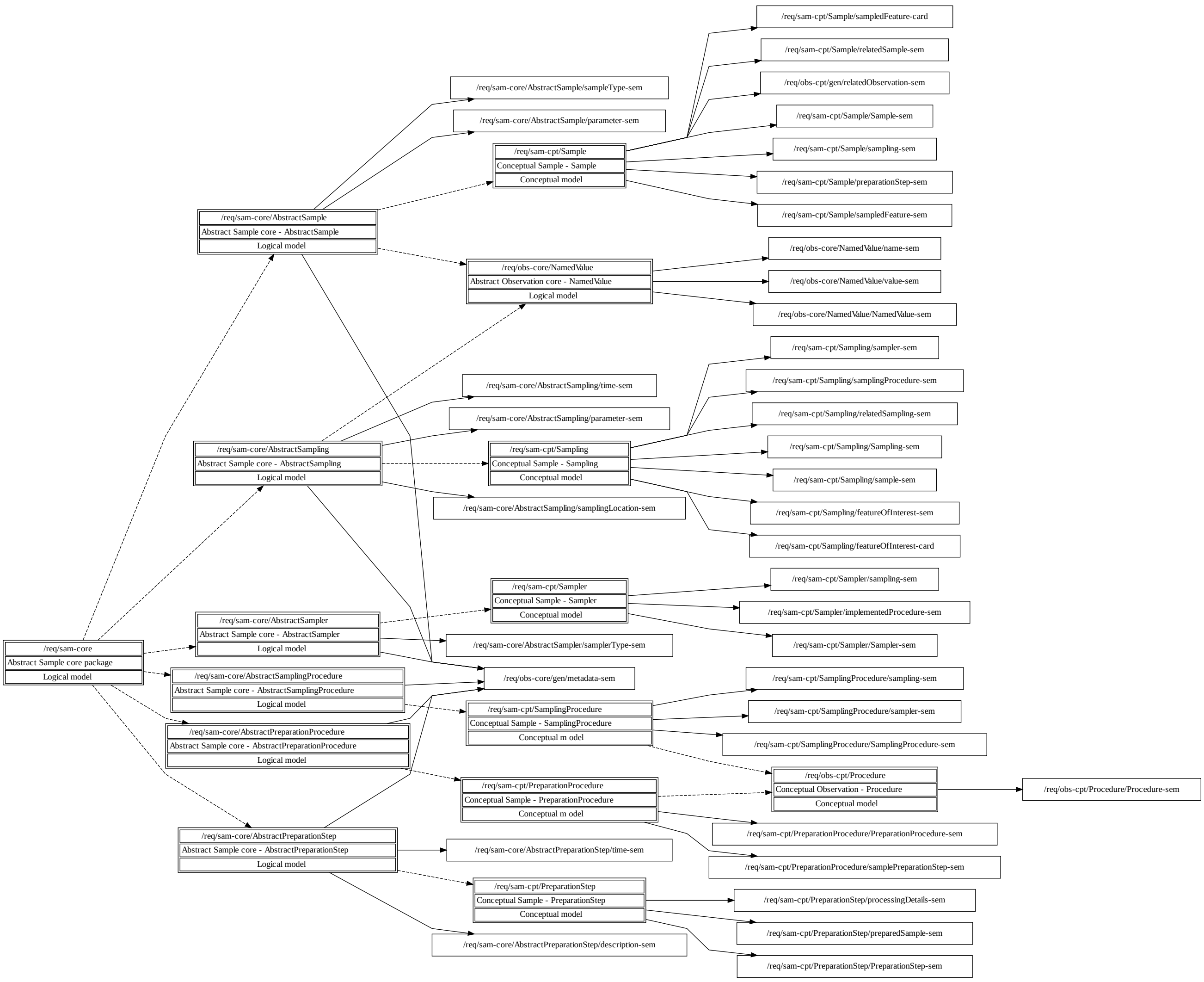

Figure 57 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core package requirements class

12.2 AbstractSample

12.2.1 AbstractSample Requirements Class

Requirements Class	/req/sam-core/AbstractSample
Target type	Logical model
Name	Abstract Sample core - AbstractSample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-cpt/Sample
Imports	/req/obs-core/NamedValue

Requirement	/req/sam-core/AbstractSample/sampleType-sem
Requirement	/req/sam-core/AbstractSample/parameter-sem
Requirement	/req/obs-core/gen/metadata-sem

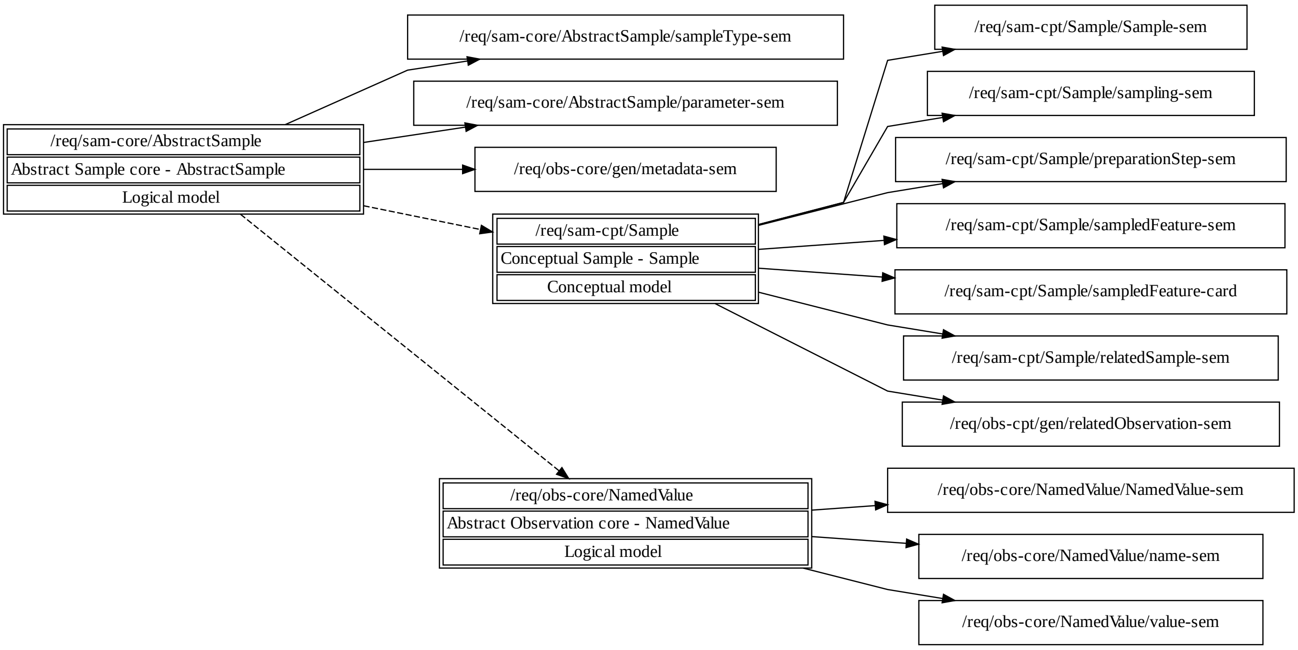

Figure 58 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractSample requirements class

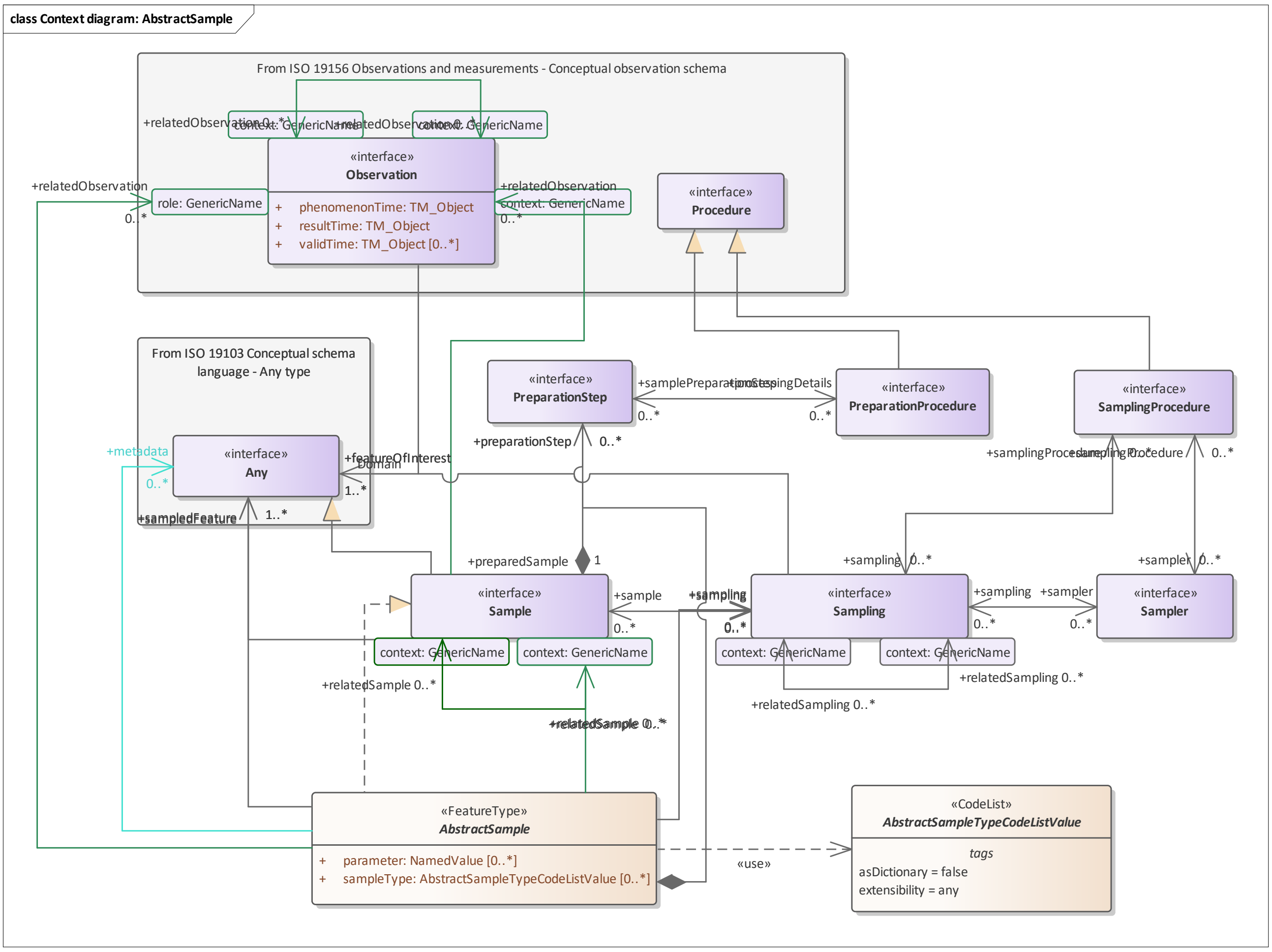

Figure 59 — Context diagram for Abstract Sample core — AbstractSample

12.2.2 Attribute sampleType

Requirement /req/sam-core/AbstractSample/sampleType-sem	The type of **Sample** according to a community agreed typology. If information on the type of **AbstractSample** is provided, the attribute **sampleType:AbstractSampleTypeCodeListValue** shall be used.

12.2.3 Attribute parameter

Requirement /req/sam-core/AbstractSample/parameter-sem	Arbitrary event-specific parameter relevant to the **Sample.** If additional parameter information is provided, the property **parameter**:**NamedValue** SHALL be used.

EXAMPLE When taking water samples, the sampling procedure specifies that an amount of time must pass to allow sediments to settle. The exact waiting time for a specific sample can be stored in the parameter.

NOTE Parameter should NOT be utilized to provide information already contained in the model by existing attributes or associations.

12.3 AbstractSampling

12.3.1 AbstractSampling Requirements Class

Requirements Class	/req/sam-core/AbstractSampling
Target type	Logical model
Name	Abstract Sample core - AbstractSampling
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for data transfer conformance class
Imports	/req/sam-cpt/Sampling
Imports	/req/obs-core/NamedValue
Requirement	/req/sam-core/AbstractSampling/samplingLocation-sem
Requirement	/req/sam-core/AbstractSampling/time-sem
Requirement	/req/sam-core/AbstractSampling/parameter-sem
Requirement	/req/obs-core/gen/metadata-sem

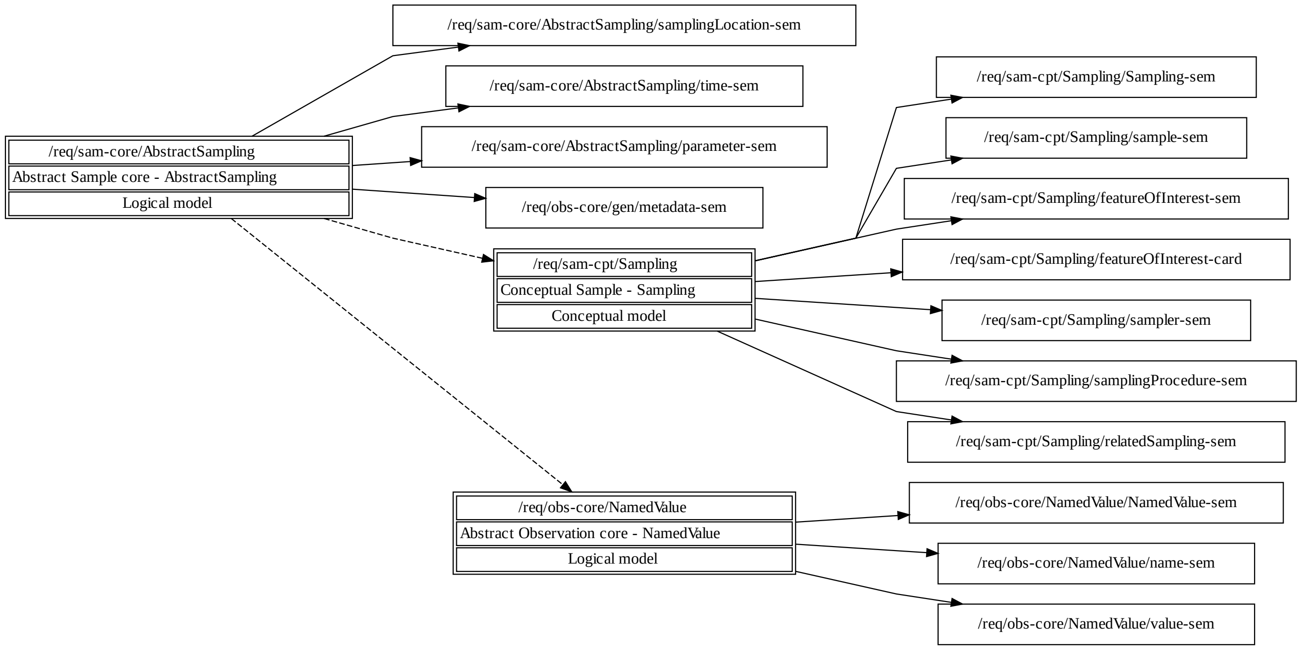

Figure 60 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractSampling requirements class

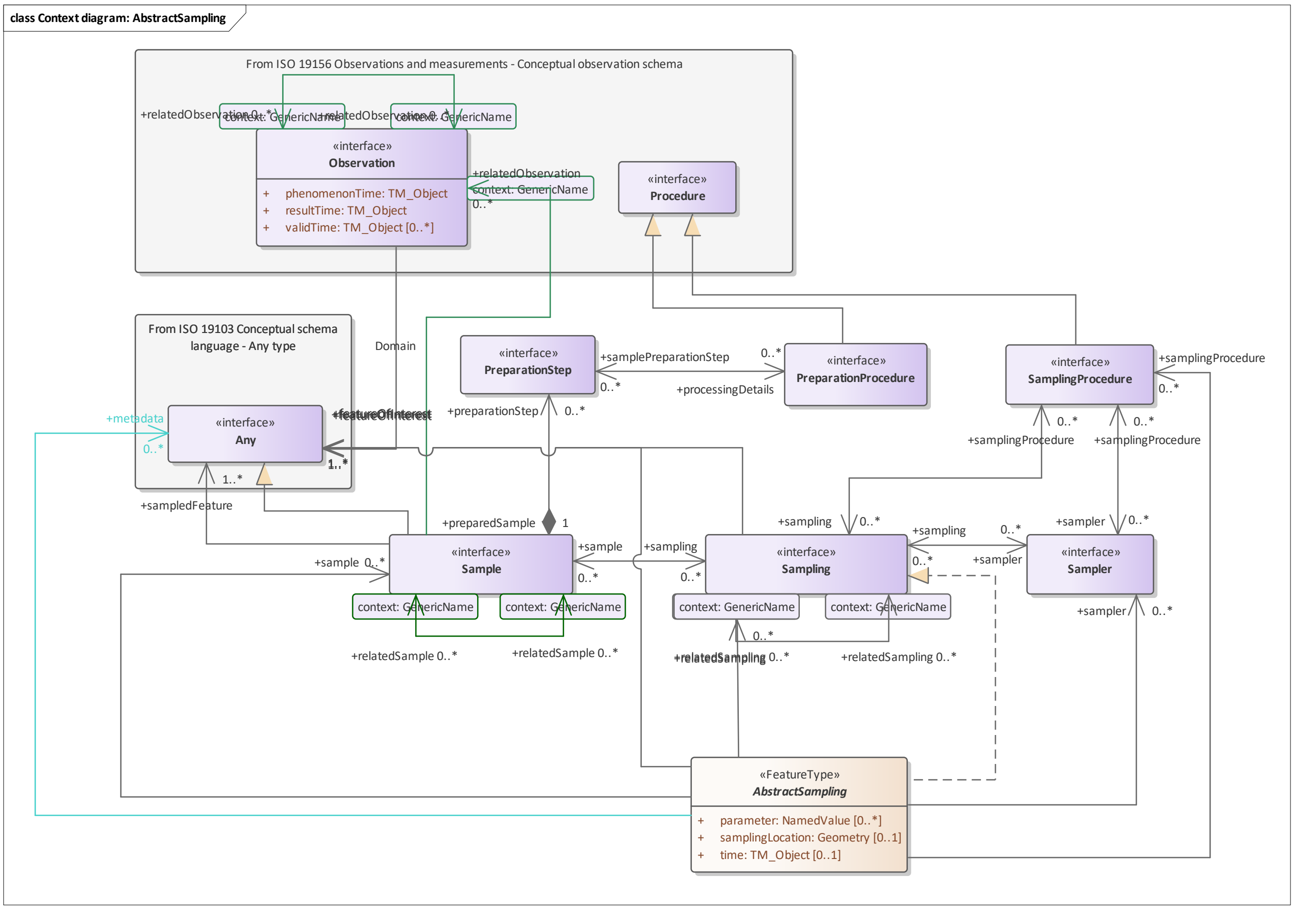

Figure 61 — Context diagram for Abstract Sample core — AbstractSampling

12.3.2 Attribute samplingLocation

Requirement /req/sam-core/AbstractSampling/samplingLocation-sem	If location information pertaining to the **Sampling** is provided, the attribute **samplingLocation:Geometry** SHALL be used.

12.3.3 Attribute time

Requirement /req/sam-core/AbstractSampling/time-sem	If information on the time of the **Sampling** is provided, the attribute **time:TM_Object** SHALL be used.

12.3.4 Attribute parameter

Requirement /req/sam-core/AbstractSampling/parameter-sem	Arbitrary event-specific parameter relevant to the **Sampling**. If additional parameter information is provided, the property **parameter**:**NamedValue** SHALL be used.

EXAMPLE When taking water samples, the sampling procedure specifies that an amount of time must pass to allow sediments to settle. The exact waiting time used in this Sampling can be stored in the parameter.

NOTE Parameter should NOT be utilized to provide information already contained in the model by existing attributes or associations.

12.4 AbstractSampler

12.4.1 AbstractSampler Requirements Class

Requirements Class	/req/sam-core/AbstractSampler
Target type	Logical model
Name	Abstract Sample core - AbstractSampler
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-cpt/Sampler
Requirement	/req/sam-core/AbstractSampler/samplerType-sem
Requirement	/req/obs-core/gen/metadata-sem

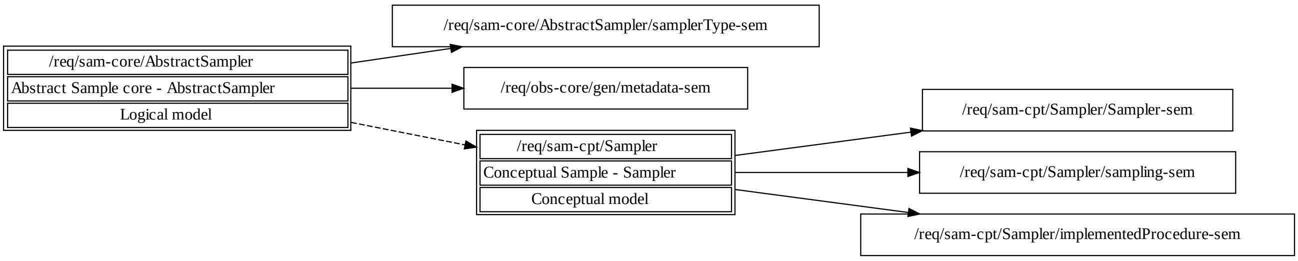

Figure 62 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractSampler requirements class

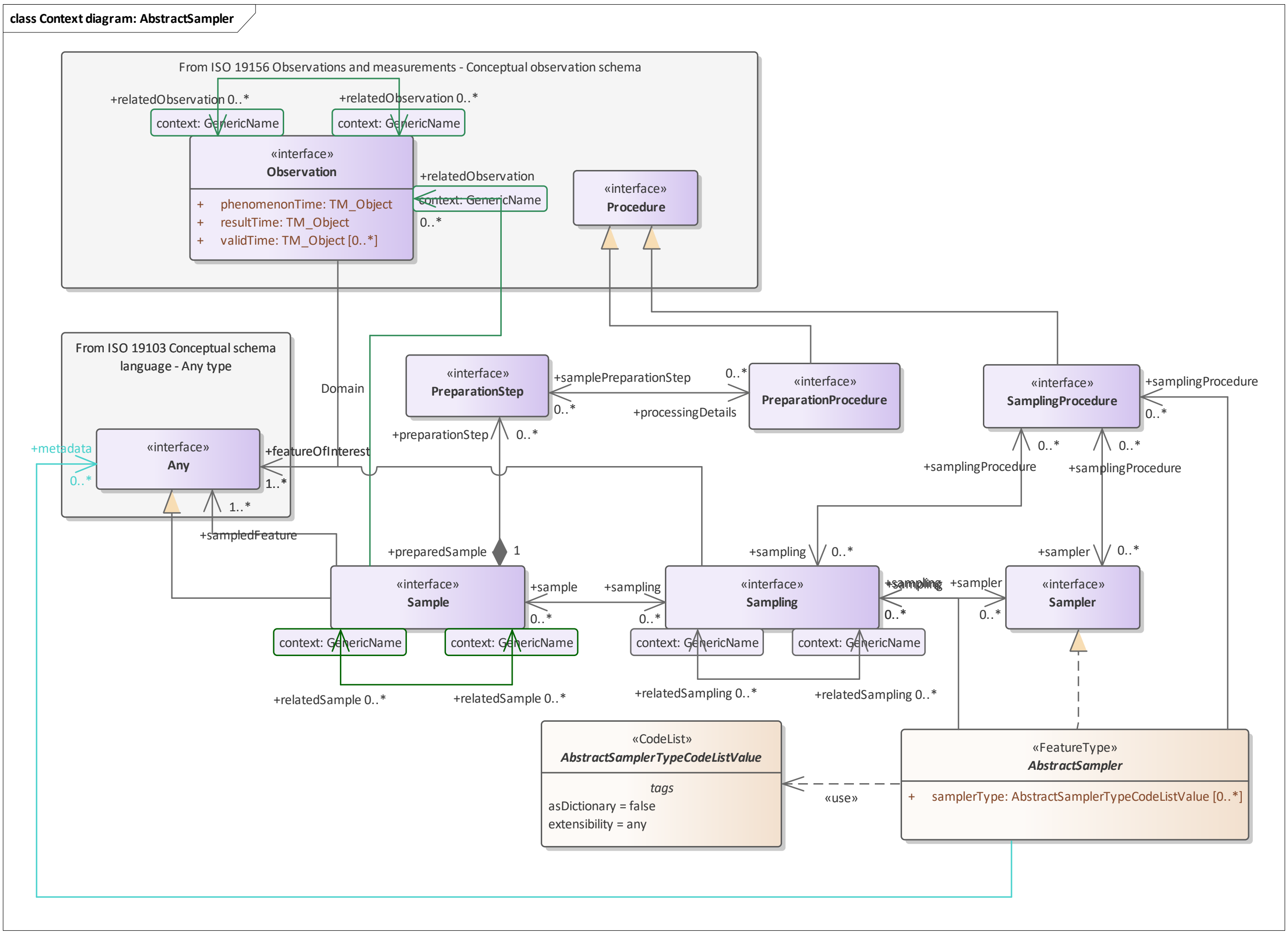

Figure 63 — Context diagram for the Abstract Sample core — AbstractSampler

12.4.2 Attribute samplerType

Requirement /req/sam-core/AbstractSampler/samplerType-sem	The type of **Sampler** according to a community agreed typology. If information on the type of **AbstractSampler** is provided, the attribute **samplerType:AbstractSamplerTypeCodeListValue** shall be used.

EXAMPLES

— A ball mill, diamond drill, hammer,

— hypodermic syringe and needle,

— image sensor, a soil auger,

— a human being.

12.5 AbstractSamplingProcedure

12.5.1 AbstractSamplingProcedure Requirements Class

Requirements Class	/req/sam-core/AbstractSamplingProcedure
Target type	Logical model
Name	Abstract Sample core - AbstractSamplingProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-cpt/SamplingProcedure
Requirement	/req/obs-core/gen/metadata-sem

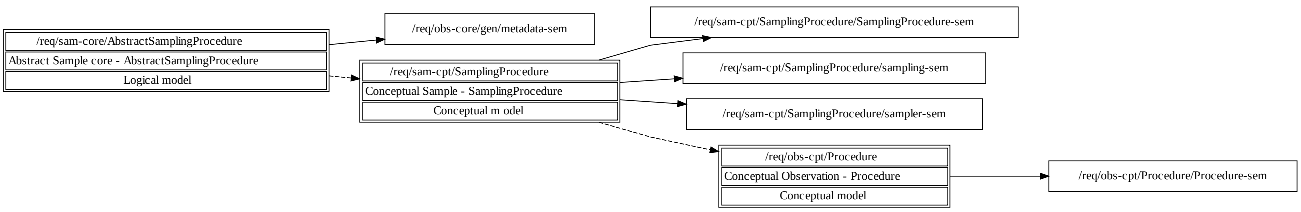

Figure 64 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractSamplingProcedure requirements class

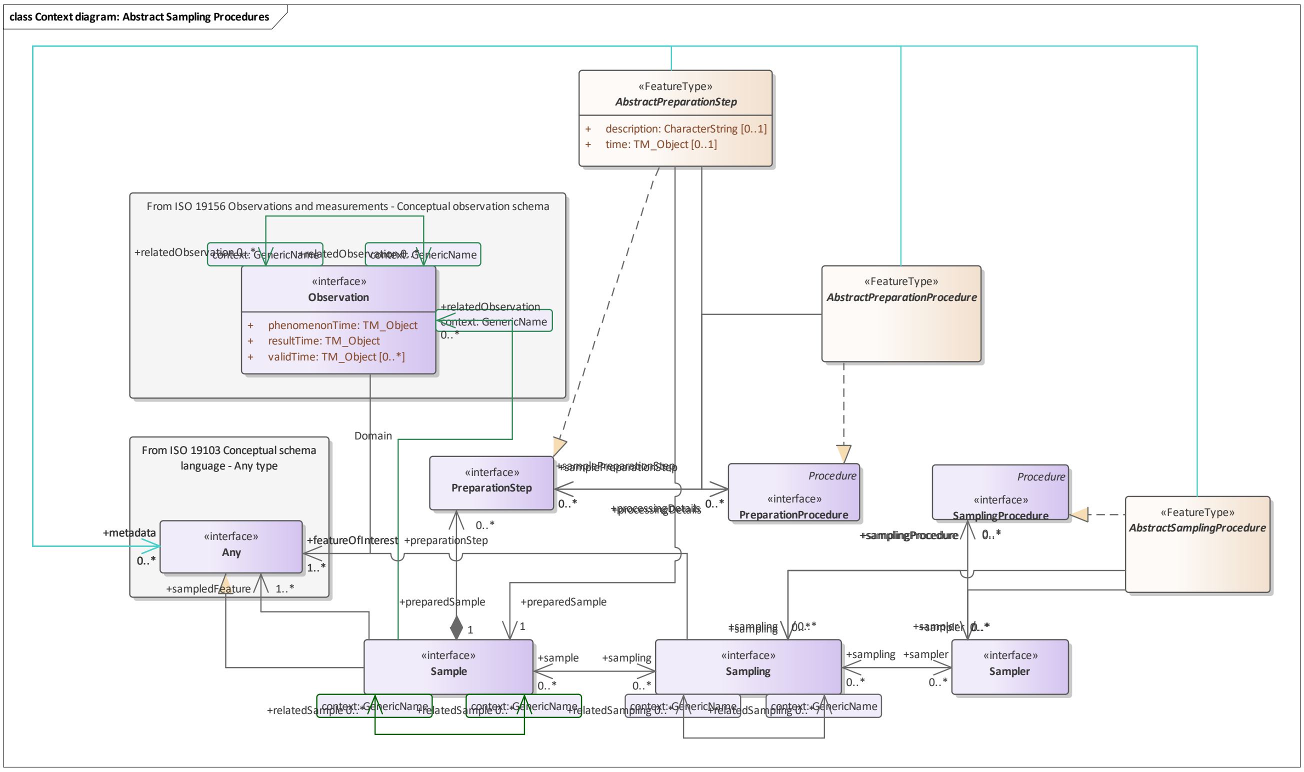

Figure 65 — Context diagram for Abstract Sample core — AbstractSamplingProcedure, AbstractPreparationProcedure and AbstractPreparationStep

12.6 AbstractPreparationProcedure

12.6.1 AbstractPreparationProcedure Requirements Class

Requirements Class	/req/sam-core/AbstractPreparationProcedure
Target type	Logical model
Name	Abstract Sample core - AbstractPreparationProcedure
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-cpt/PreparationProcedure
Requirement	/req/obs-core/gen/metadata-sem

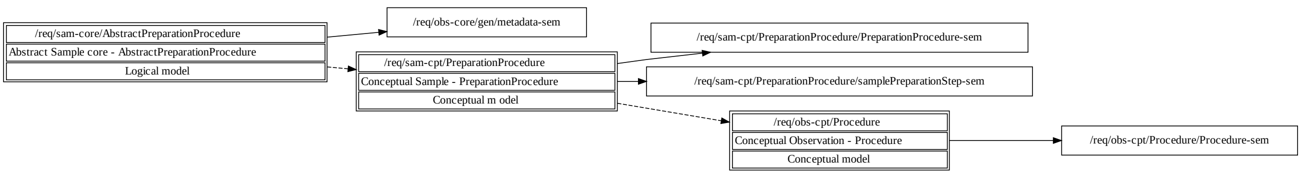

Figure 66 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractPreparationProcedure requirements class

12.7 AbstractPreparationStep

12.7.1 AbstractPreparationStep Requirements Class

Requirements Class	/req/sam-core/AbstractPreparationStep
Target type	Logical model
Name	Abstract Sample core - AbstractPreparationStep
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19108:2002 Geographic information – Temporal schema, Application schemas for data transfer conformance class
Imports	/req/sam-cpt/PreparationStep
Requirement	/req/sam-core/AbstractPreparationStep/description-sem
Requirement	/req/sam-core/AbstractPreparationStep/time-sem
Requirement	/req/obs-core/gen/metadata-sem

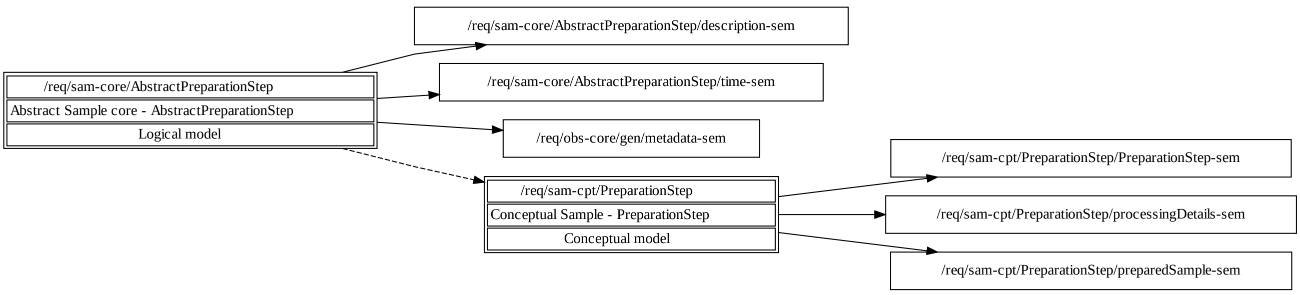

Figure 67 — (Informative) Included direct and indirect requirements and recommendations of the Abstract Sample core — AbstractPreparationStep requirements class

12.7.2 Attribute description

Requirement /req/sam-core/AbstractPreparationStep/description-sem	Description of the **preparationStep**. If a description pertaining to the **preparationStep** is provided, the attribute **description:CharacterString** SHALL be used.

12.7.3 Attribute time

Requirement /req/sam-core/AbstractPreparationStep/time-sem	Time of the **preparationStep**. If information on the time of the **preparationStep** of the **Sampling** is provided, the attribute **time:TM_Object** SHALL be used.

13 Basic Samples

13.1 General

13.1.1 Basic Samples Package Requirements Class

Requirements Class	/req/sam-basic
Target type	Logical model
Name	Basic Samples package
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-basic/Sample
Imports	/req/sam-basic/SpatialSample
Imports	/req/sam-basic/MaterialSample
Imports	/req/sam-basic/StatisticalSample
Imports	/req/sam-basic/Sampling
Imports	/req/sam-basic/Sampler
Imports	/req/sam-basic/SampleCollection

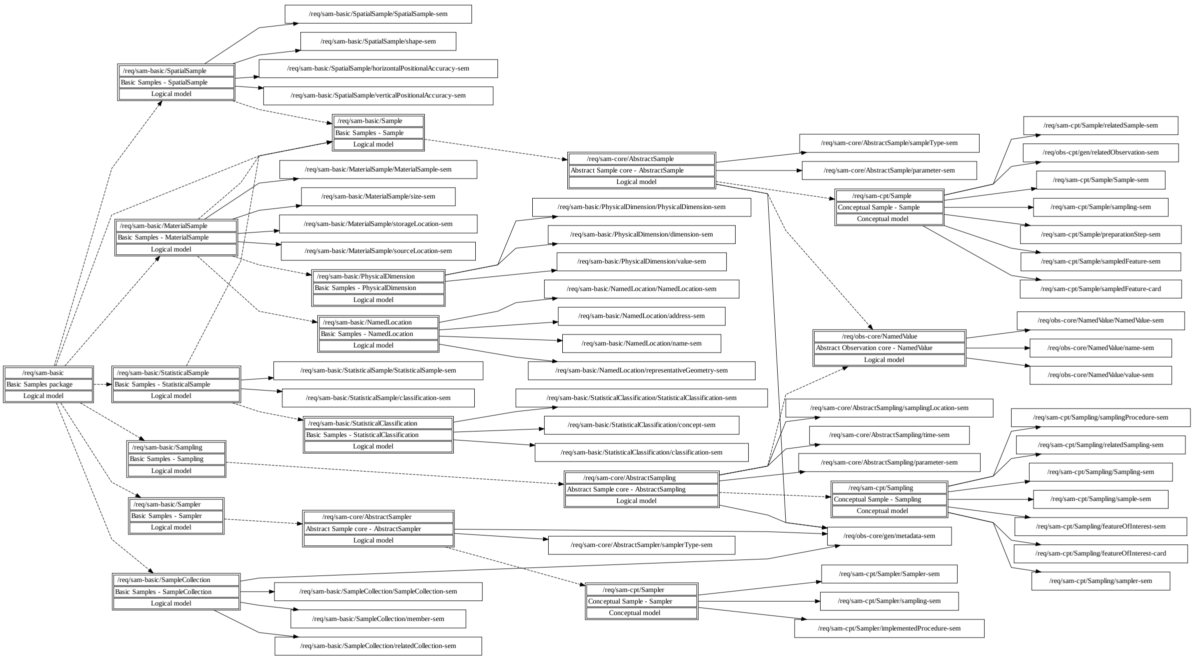

Figure 68 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples package requirements class

13.2 Sample

13.2.1 Sample Requirements Class

Requirements Class	/req/sam-basic/Sample
Target type	Logical model
Name	Basic Samples - Sample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-core/AbstractSample

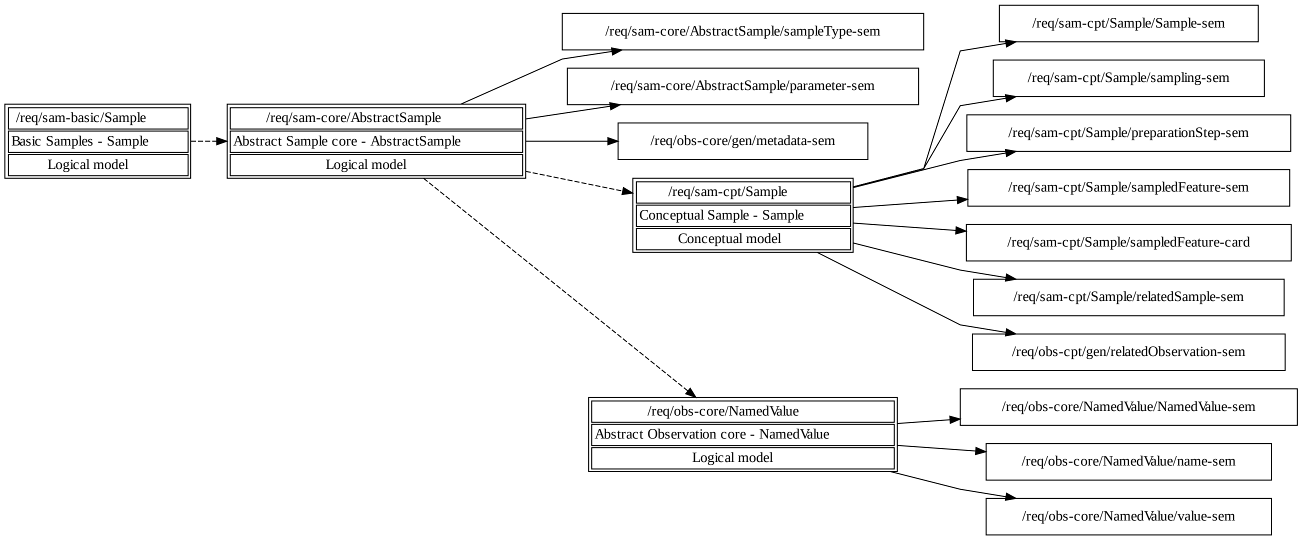

Figure 69 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — Sample requirements class

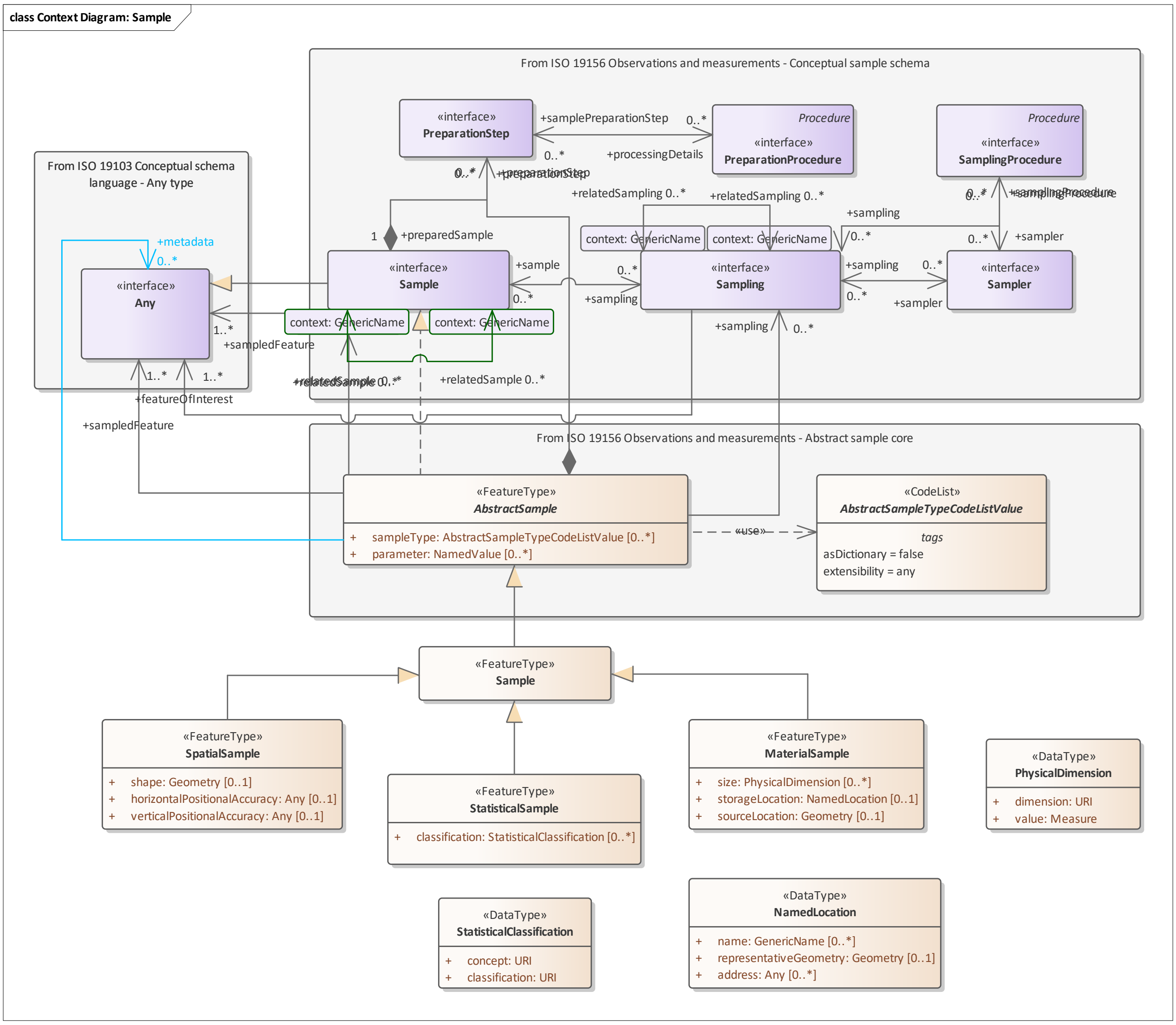

Figure 70 — Context diagram for Basic Samples — Sample, SpatialSample, StatisticalSample and MaterialSample

13.3 SpatialSample

13.3.1 SpatialSample Requirements Class

Requirements Class	/req/sam-basic/SpatialSample
Target type	Logical model
Name	Basic Samples - SpatialSample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class

Dependency	ISO 19107:2019 Geographic information – Spatial schema, Geometry conformance class
Imports	/req/sam-basic/Sample
Requirement	/req/sam-basic/SpatialSample/SpatialSample-sem
Requirement	/req/sam-basic/SpatialSample/shape-sem
Requirement	/req/sam-basic/SpatialSample/horizontalPositionalAccuracy-sem
Requirement	/req/sam-basic/SpatialSample/verticalPositionalAccuracy-sem

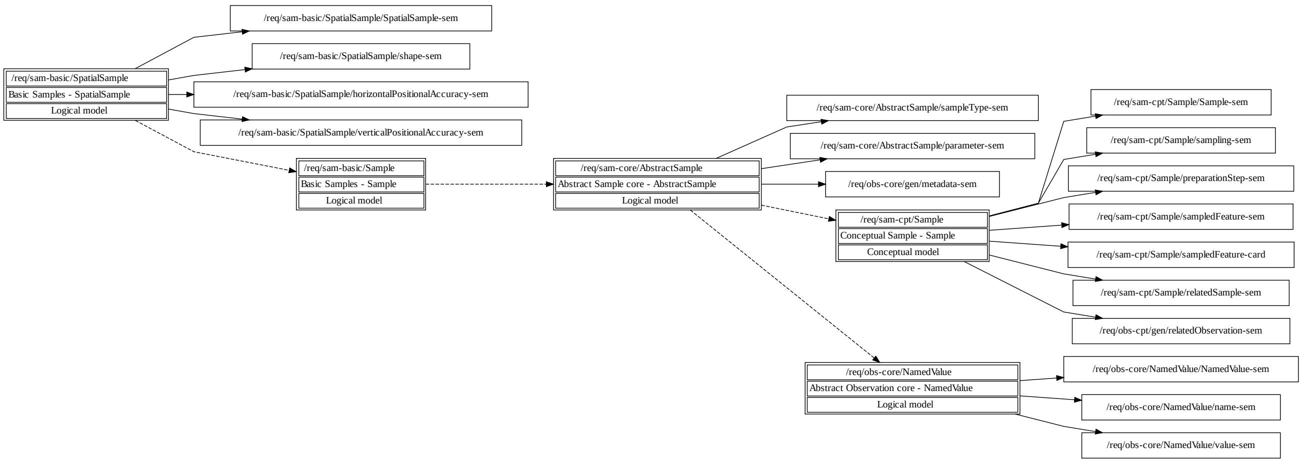

Figure 71 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — SpatialSample requirements class

13.3.2 Feature type SpatialSample

Requirement /req/sam-basic/SpatialSample/SpatialSample-sem	A **SpatialSample** is a geospatial **Sample**.

NOTE When observations are made to estimate properties of a geospatial feature, in particular where the value of a property varies within the scope of the feature, a SpatialSample is used. Depending on accessibility and on the nature of the expected property variation, the SpatialSample may be extensive in one, two or three spatial dimensions.

EXAMPLE

— Typically an Observation ‘site’ or 'station' connotes the 'world in the vicinity of the site (or station)', so the observed properties relate to the physical medium at the station, and not to any physical artifact such as a mooring, buoy, benchmark, monument, well, etc.

— Some common names for SpatialSample used in various application domains include Borehole, Flightline, Interval, Lidar Cloud, Map Horizon, Microscope Slide, Mine Level, Mine, Observation Well, Profile, Pulp, Quadrat, Scene, Section, ShipsTrack, Spot, Station, Swath, Trajectory, Traverse, etc.

13.3.3 Attribute shape

Requirement /req/sam-basic/SpatialSample/shape-sem	The **shape** is the Geometry of the **SpatialSample**. If location information pertaining to the **SpatialSample** is provided, the attribute **shape:Geometry** SHALL be used.

NOTE The shape of the SpatialSample is the context for domain decomposition.

EXAMPLE Logs of different properties along a well or borehole might use different intervals, and sub-samples might be either spatially instantaneous, or averaged in some way over an interval. The position of the samples can be conveniently described in terms of offsets in a linear coordinate reference system that is defined by the shape of the well axis.

13.3.4 Attribute horizontalPositionalAccuracy

Requirement /req/sam-basic/SpatialSample/horizontalPositionalAccuracy-sem	The **PositionalAccuracy** of the horizontal component of the Geometry of the **SpatialSample**. If horizontal **PositionalAccuracy** information pertaining to the **SpatialSample** is provided, the attribute **horizontalPositionalAccuracy:Any** SHALL be used.

13.3.5 Attribute verticalPositionalAccuracy

Requirement /req/sam-basic/SpatialSample/verticalPositionalAccuracy-sem	The **PositionalAccuracy** of the vertical component of the Geometry of the **SpatialSample**. If horizontal **PositionalAccuracy** information pertaining to the **SpatialSample** is provided, the attribute **verticalPositionalAccuracy:Any** SHALL be used.

13.4 MaterialSample

13.4.1 MaterialSample Requirements Class

Requirements Class	/req/sam-basic/MaterialSample
Target type	Logical model
Name	Basic Samples - MaterialSample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Imports	/req/sam-basic/Sample
Imports	/req/sam-basic/PhysicalDimension
Imports	/req/sam-basic/NamedLocation
Requirement	/req/sam-basic/MaterialSample/MaterialSample-sem
Requirement	/req/sam-basic/MaterialSample/size-sem
Requirement	/req/sam-basic/MaterialSample/storageLocation-sem
Requirement	/req/sam-basic/MaterialSample/sourceLocation-sem

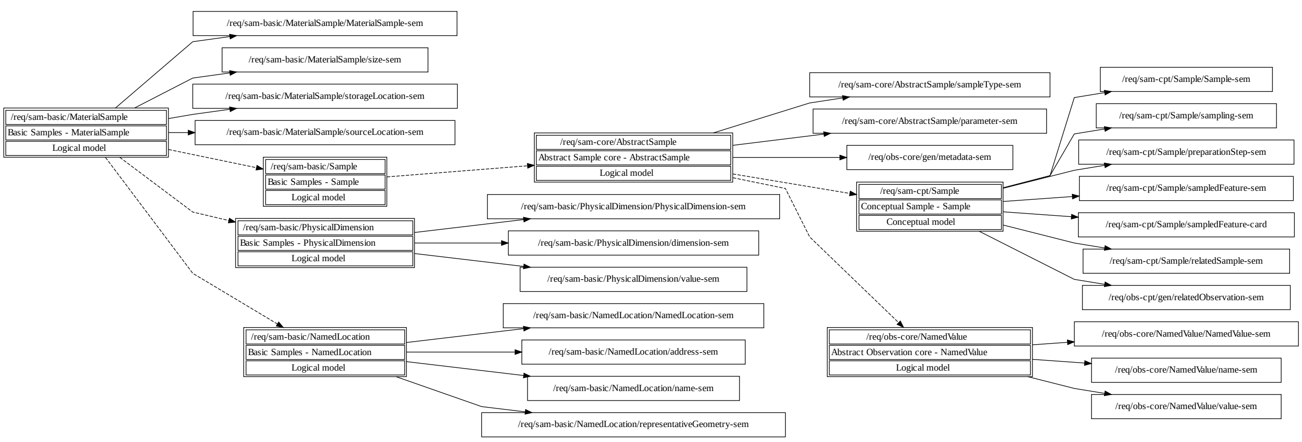

Figure 72 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — MaterialSample requirements class

13.4.2 Feature type MaterialSample

Requirement /req/sam-basic/MaterialSample/MaterialSample-sem	A **MaterialSample** is a physical, tangible **Sample**.

NOTE 1 MaterialSamples that are curated and preserved are sometimes known as 'specimens'.

NOTE 2 MaterialSamples can be destroyed in connexion with the observation act.

NOTE 3 A MaterialSample is a physical Sample of a FeatureOfInterest, obtained for Observation(s) normally carried out ex-situ, sometimes in a laboratory.

13.4.3 Attribute size

Requirement /req/sam-basic/MaterialSample/size-sem	The **size** describes a physical extent of the specimen. If size information pertaining to the **MaterialSample** is provided, the attribute **size:PhysicalDimension** SHALL be used.

NOTE The size may be length, mass, volume, etc., as appropriate for the specimen instance and its material type.

13.4.4 Attribute storageLocation

Requirement /req/sam-basic/MaterialSample/storageLocation-sem	The **storageLocation** is the location of a **MaterialSample**. If information pertaining to the storage location of the **MaterialSample** is provided, the attribute **storageLocation:NamedLocation** SHALL be used.

NOTE The storageLocation may be a location such as a shelf in a warehouse or a drawer in a museum.

13.4.5 Attribute sourceLocation

Requirement /req/sam-basic/MaterialSample/sourceLocation-sem	The **sourceLocation** is the location from where the **MaterialSample** was obtained. If information pertaining to the source location of the **MaterialSample** is provided, the attribute **sourceLocation:Geometry** SHALL be used.

NOTE Where a MaterialSample has a relatedSample whose location provides an unambiguous location then this attribute is not required. However, if the specific sampling location within the sampledFeature is important, then the sourceLocation can be used to provide it.

13.5 StatisticalSample

13.5.1 StatisticalSample Requirements Class

Requirements Class	/req/sam-basic/StatisticalSample
Target type	Logical model
Name	Basic Samples - StatisticalSample
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-basic/Sample
Imports	/req/sam-basic/StatisticalClassification
Requirement	/req/sam-basic/StatisticalSample/StatisticalSample-sem
Requirement	/req/sam-basic/StatisticalSample/classification-sem

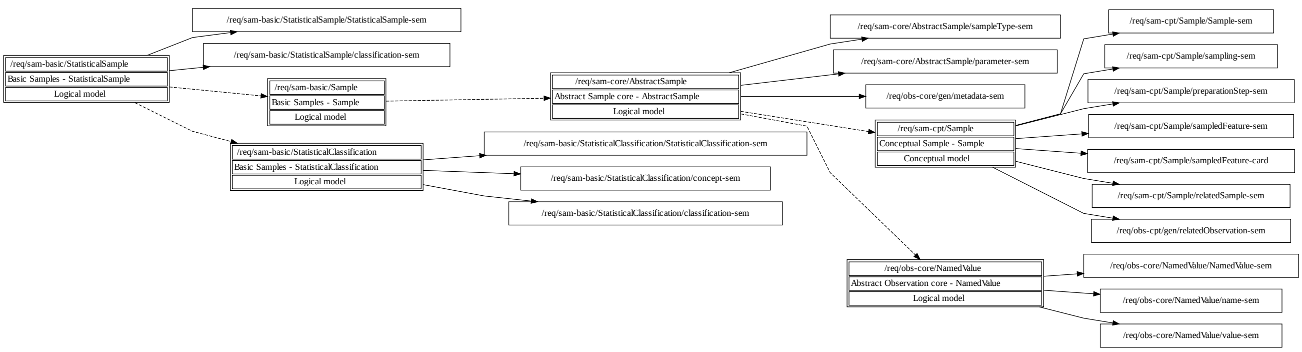

Figure 73 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — StatisticalSample requirements class

13.5.2 Feature type StatisticalSample

Requirement /req/sam-basic/StatisticalSample/StatisticalSample-sem	A **StatisticalSample** is a statistical subset of a feature-of-interest, defined for the purpose of creating **Observation**(s).

NOTE StatisticalSamples usually apply to populations or other sets, of which certain subset may be of specific interest.

EXAMPLE The male or female subset of a population.

13.5.3 Attribute classification

Requirement /req/sam-basic/StatisticalSample/classification-sem	The **classification** describes a criterion by which the subset was defined. If information pertaining to the subsetting criteria by which a **StatisticalSample** has been defined is provided, the attribute **classification:StatisticalClassification** SHALL be used.

NOTE The classification may be age, gender, etc., as appropriate for the set or population on which the subsetting is performed.

13.6 Sampling

13.6.1 Sampling Requirements Class

Requirements Class	/req/sam-basic/Sampling
Target type	Logical model
Name	Basic Samples - Sampling
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	Unified Modeling Language (UML). Version 2.3. May 2010
Imports	/req/sam-core/AbstractSampling

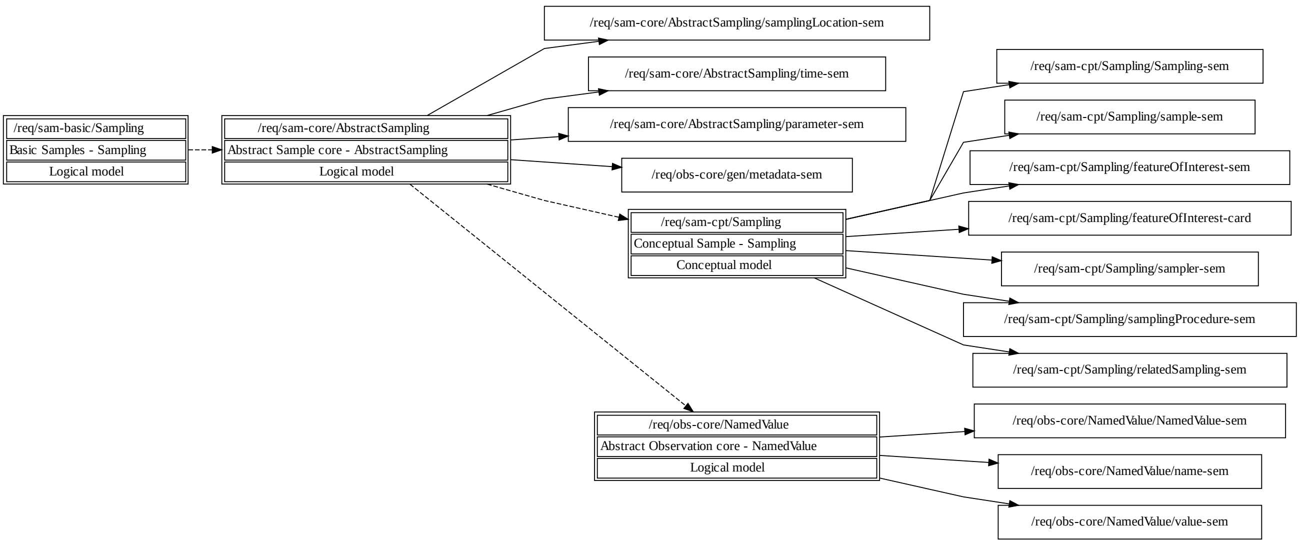

Figure 74 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — Sampling requirements class

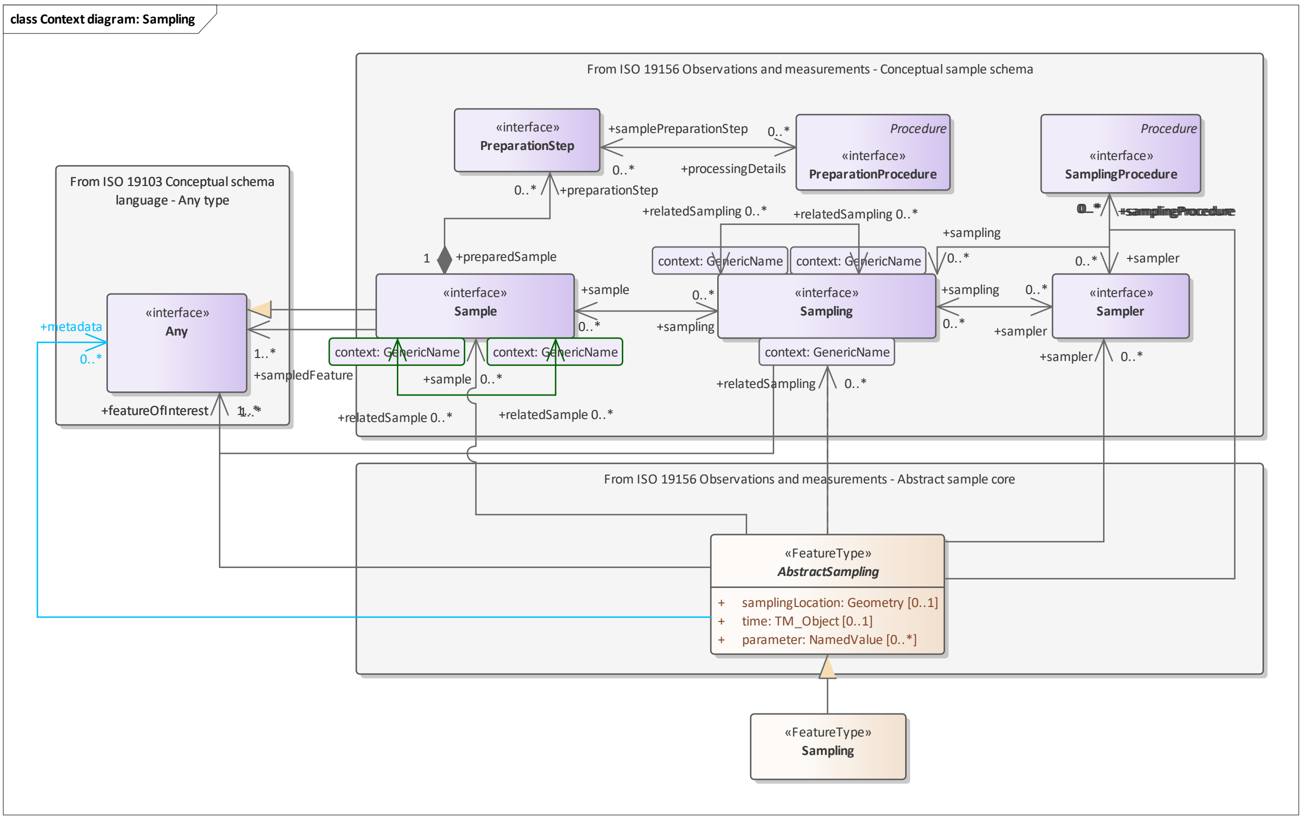

Figure 75 — Context diagram for Basic Samples — Sampling

13.7 Sampler

13.7.1 Sampler Requirements Class

Requirements Class	/req/sam-basic/Sampler
Target type	Logical model
Name	Basic Samples - Sampler
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Imports	/req/sam-core/AbstractSampler

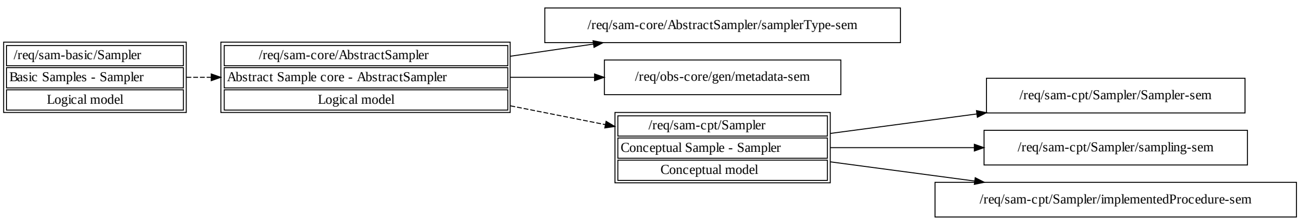

Figure 76 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — Sampler requirements class

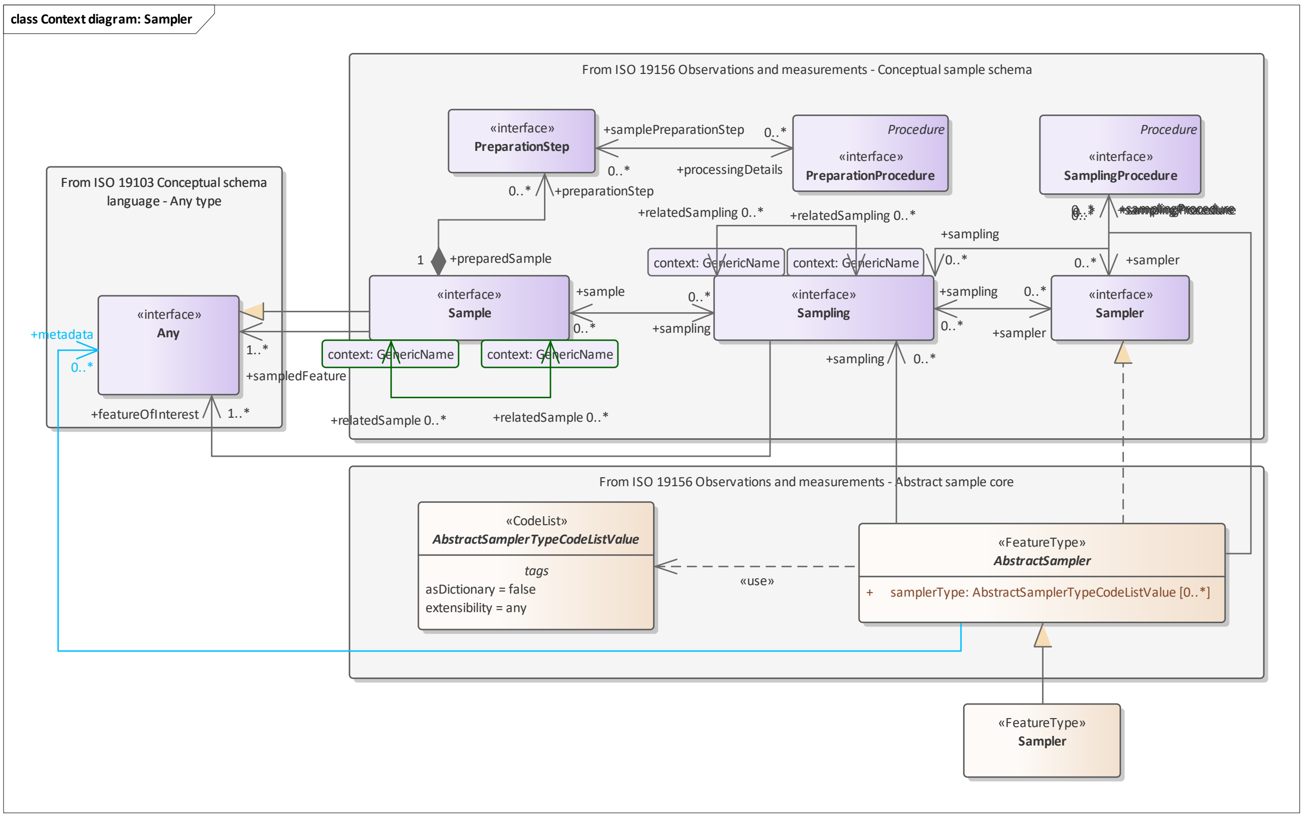

Figure 77 — Context diagram for Basic Samples — Sampler

13.8 SampleCollection

13.8.1 SampleCollection Requirements Class

Requirements Class	/req/sam-basic/SampleCollection
Target type	Logical model
Name	Basic Samples - SampleCollection
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Requirement	/req/sam-basic/SampleCollection/SampleCollection-sem
Requirement	/req/sam-basic/SampleCollection/member-sem
Requirement	/req/sam-basic/SampleCollection/relatedCollection-sem
Requirement	/req/obs-core/gen/metadata-sem

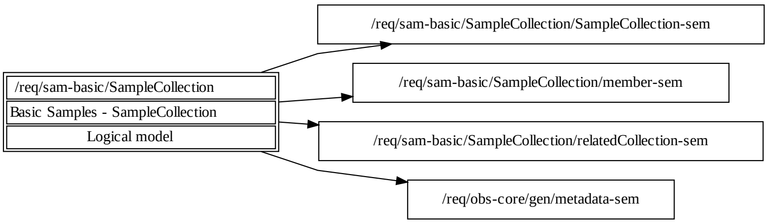

Figure 78 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — SampleCollection requirements class

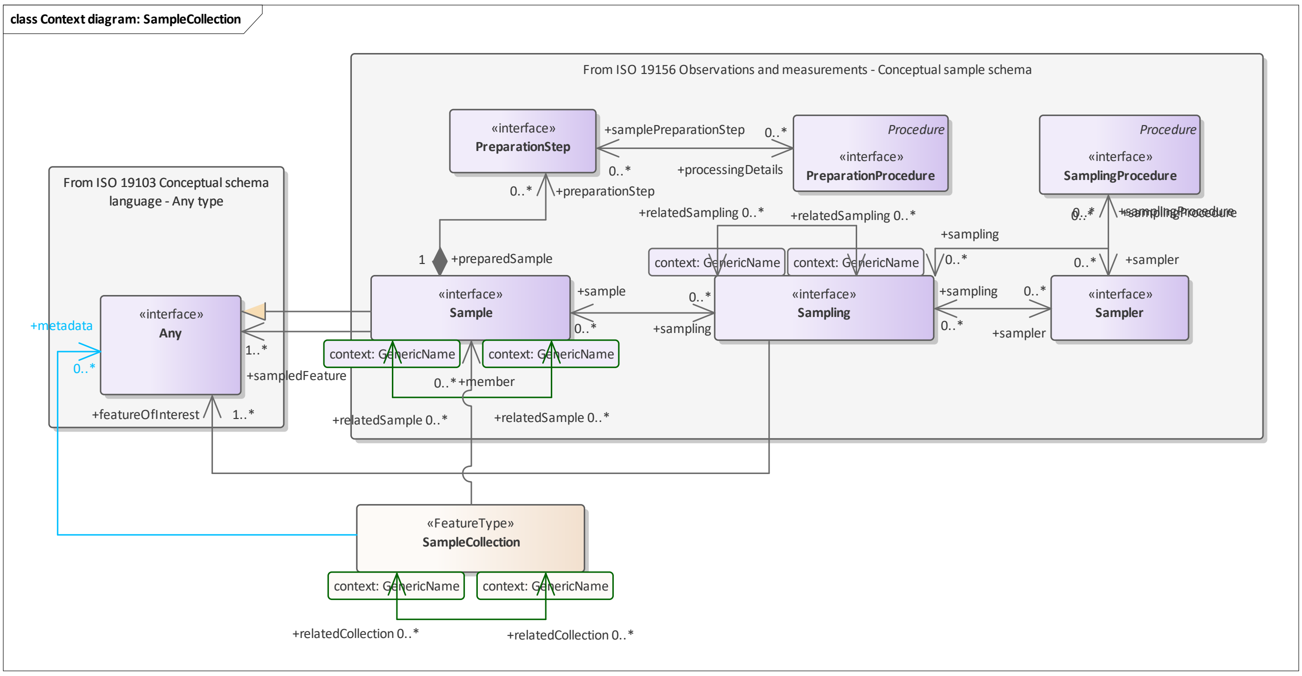

Figure 79 — Context diagram for Basic Samples — SampleCollection

13.8.2 Feature type SampleCollection

Requirement /req/sam-basic/SampleCollection/SampleCollection-sem	A collection of **Sample**s.

13.8.3 Association member

Requirement /req/sam-basic/SampleCollection/member-sem	A **member** of the **SampleCollection**. If the **SampleCollection** has members, the association with the role **member** SHALL be used.

13.8.4 Association relatedCollection

Requirement /req/sam-basic/SampleCollection/relatedCollection-sem	A **SampleCollection** the **SampleCollection** is related to. If a reference to a related **SampleCollection** is provided, the association with role **relatedCollection** SHALL be used. The **context:GenericName** qualifier of this association may be used to provide further information as to the nature of the relation.

13.9 PhysicalDimension

13.9.1 PhysicalDimension Requirements Class

Requirements Class	/req/sam-basic/PhysicalDimension
Target type	Logical model
Name	Basic Samples - PhysicalDimension
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Requirement	/req/sam-basic/PhysicalDimension/PhysicalDimension-sem
Requirement	/req/sam-basic/PhysicalDimension/dimension-sem
Requirement	/req/sam-basic/PhysicalDimension/value-sem

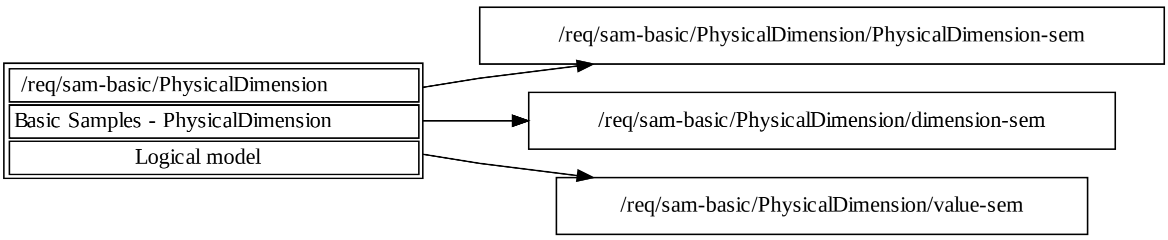

Figure 80 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — PhysicalDimension requirements class

13.9.2 Data type PhysicalDimension

Requirement /req/sam-basic/PhysicalDimension/PhysicalDimension-sem	A dataType for the provision of various size quantities.

13.9.3 Attribute dimension

Requirement /req/sam-basic/PhysicalDimension/dimension-sem	The **PhysicalDimension** about which a **value** is provided. The name of the physical dimension, the quantity being provided in the value SHALL be provided in the attribute **dimension:URI.**

13.9.4 Attribute value

Requirement /req/sam-basic/PhysicalDimension/value-sem	The **value** of the **PhysicalDimension**. The measure of the quantity being provided SHALL be provided in the attribute **value:Measure**

13.10 NamedLocation

13.10.1 NamedLocation Requirements Class

Requirements Class	/req/sam-basic/NamedLocation
Target type	Logical model
Name	Basic Samples - NamedLocation
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreTypes conformance class
Dependency	ISO 19107:2019 Geographic information — Spatial schema, Geometry conformance class
Requirement	/req/sam-basic/NamedLocation/NamedLocation-sem
Requirement	/req/sam-basic/NamedLocation/address-sem
Requirement	/req/sam-basic/NamedLocation/name-sem

Requirement	/req/sam-basic/NamedLocation/representativeGeometry-sem

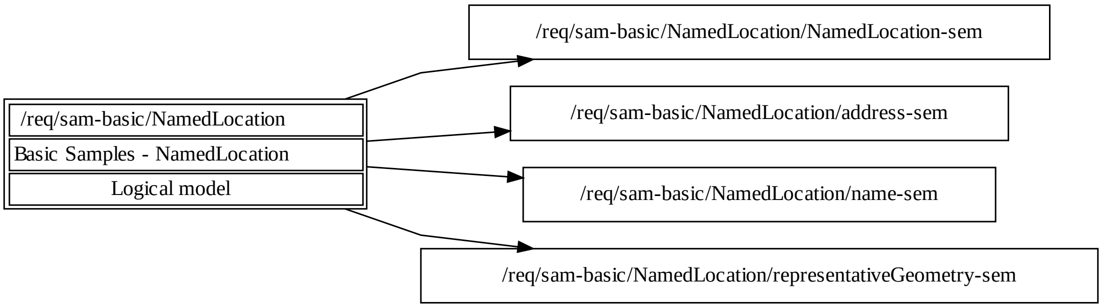

Figure 81 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — NamedLocation requirements class

13.10.2 Data type NamedLocation

Requirement /req/sam-basic/NamedLocation/NamedLocation-sem	A location identified by its name, address, spatial geometry or a combination of any of these three.

13.10.3 Attribute address

Requirement /req/sam-basic/NamedLocation/address-sem	An **address** used for identifying a **NamedLocation.** If **address** information is provided, the attribute **address:Any** SHALL be used.

13.10.4 Attribute name

Requirement /req/sam-basic/NamedLocation/name-sem	A **name** used for identifying a **NamedLocation.** If **name** information is provided, the attribute **name:GenericName** SHALL be used.

13.10.5 Attribute representativeGeometry

Requirement /req/sam-basic/NamedLocation/representativeGeometry-sem	A geometry used for providing a representative spatial location of a **NamedLocation.** If **geometry** is provided, the attribute **representativeGeometry:Geometry** SHALL be used.

13.11 StatisticalClassification

13.11.1 StatisticalClassification Requirements Class

Requirements Class	/req/sam-basic/StatisticalClassification
Target type	Logical model
Name	Basic Samples - StatisticalClassification
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015 Geographic information – Conceptual schema language, CoreExtendedTypes conformance class
Requirement	/req/sam-basic/StatisticalClassification/StatisticalClassification-sem
Requirement	/req/sam-basic/StatisticalClassification/concept-sem
Requirement	/req/sam-basic/StatisticalClassification/classification-sem

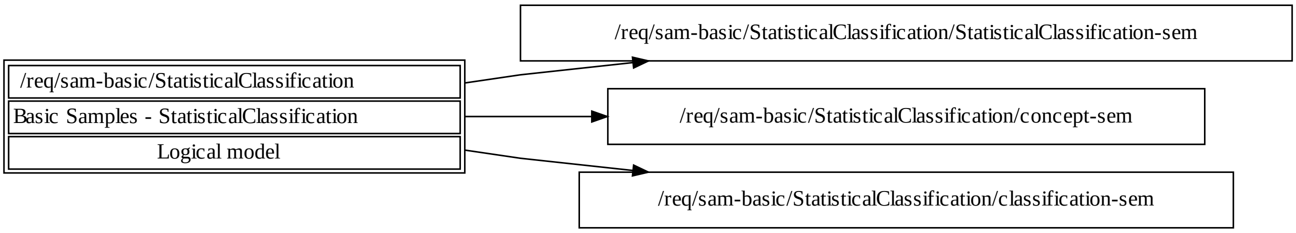

Figure 82 — (Informative) Included direct and indirect requirements and recommendations of the Basic Samples — StatisticalClassification requirements class

13.11.2 Data type StatisticalClassification

Requirement /req/sam-basic/StatisticalClassification/StatisticalClassification-sem	A dataType for the provision of information on statistical classifications.

13.11.3 Attribute concept

Requirement /req/sam-basic/StatisticalClassification/concept-sem	The **concept** by which a **StatisticalClassification** is to be performed. The name of the concept by which the statistical classification is performed SHALL be provided in the attribute **concept:URI.**

EXAMPLE The concept for a statistical classification could be age, gender, color, size etc.

13.11.4 Attribute classification

Requirement /req/sam-basic/StatisticalClassification/classification-sem	The explicit **classification class** pertaining to the classification concept described by the **StatisticalClassification** . The classification class of the **StatisticalClassification** SHALL be provided in the attribute **classification:URI.**

EXAMPLE The classification for a statistical classification could be:

— Age Brackets: [0-10], [10-20]

— Genders: Male, Female, Other

— Color: Red, Green, Blue

Annex A
(normative)

Abstract Test Suite

A.1 Abstract tests for Conceptual Observation schema package

A.1.1 Conceptual Observation schema package

Conformance Class	**/conf/obs-cpt**
Requirements	/req/obs-cpt
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.2 Conceptual Observation - Deployment

Conformance Class	**/conf/obs-cpt/Deployment**
Requirements	/req/obs-cpt/Deployment
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.3 Conceptual Observation - Host

Conformance Class	**/conf/obs-cpt/Host**
Requirements	/req/obs-cpt/Host
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.4 Conceptual Observation - ObservableProperty

Conformance Class	**/conf/obs-cpt/ObservableProperty**
Requirements	/req/obs-cpt/ObservableProperty
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.5 Conceptual Observation - Observation

Conformance Class	**/conf/obs-cpt/Observation**
Requirements	/req/obs-cpt/Observation
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.6 Conceptual Observation - Observer

Conformance Class	**/conf/obs-cpt/Observer**
Requirements	/req/obs-cpt/Observer
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.7 Conceptual Observation - ObservingProcedure

Conformance Class	**/conf/obs-cpt/ObservingProcedure**
Requirements	/req/obs-cpt/ObservingProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.1.8 Conceptual Observation - Procedure

Conformance Class	**/conf/obs-cpt/Procedure**
Requirements	/req/obs-cpt/Procedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2 Abstract tests for Abstract Observation core package

A.2.1 Abstract Observation core package

Conformance Class	**/conf/obs-core**
Requirements	/req/obs-core
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.2 Abstract Observation core - AbstractDeployment

Conformance Class	**/conf/obs-core/AbstractDeployment**
Requirements	/req/obs-core/AbstractDeployment
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.3 Abstract Observation core - AbstractHost

Conformance Class	**/conf/obs-core/AbstractHost**
Requirements	/req/obs-core/AbstractHost
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.4 Abstract Observation core - AbstractObservableProperty

Conformance Class	**/conf/obs-core/AbstractObservableProperty**
Requirements	/req/obs-core/AbstractObservableProperty
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.5 Abstract Observation core - AbstractObservation

Conformance Class	**/conf/obs-core/AbstractObservation**
Requirements	/req/obs-core/AbstractObservation
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.6 Abstract Observation core - AbstractObservationCharacteristics

Conformance Class	**/conf/obs-core/AbstractObservationCharacteristics**
Requirements	/req/obs-core/AbstractObservationCharacteristics
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.7 Abstract Observation core - AbstractObserver

Conformance Class	**/conf/obs-core/AbstractObserver**
Requirements	/req/obs-core/AbstractObserver
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.8 Abstract Observation core - AbstractObservingProcedure

Conformance Class	**/conf/obs-core/AbstractObservingProcedure**
Requirements	/req/obs-core/AbstractObservingProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.2.9 Abstract Observation core - NamedValue

Conformance Class	**/conf/obs-core/NamedValue**
Requirements	/req/obs-core/NamedValue
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3 Abstract tests for Basic Observations package

A.3.1 Basic Observations package

Conformance Class	**/conf/obs-basic**
Requirements	/req/obs-basic
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.2 Basic Observations - Deployment

Conformance Class	**/conf/obs-basic/Deployment**
Requirements	/req/obs-basic/Deployment
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.3 Basic Observations - GenericDomainFeature

Conformance Class	**/conf/obs-basic/GenericDomainFeature**
Requirements	/req/obs-basic/GenericDomainFeature
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.4 Basic Observations - Host

Conformance Class	**/conf/obs-basic/Host**
Requirements	/req/obs-basic/Host
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.5 Basic Observations - ObservableProperty

Conformance Class	**/conf/obs-basic/ObservableProperty**
Requirements	/req/obs-basic/ObservableProperty
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.6 Basic Observations - Observation

Conformance Class	**/conf/obs-basic/Observation**
Requirements	/req/obs-basic/Observation
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.7 Basic Observations - ObservationCharacteristics

Conformance Class	**/conf/obs-basic/ObservationCharacteristics**
Requirements	/req/obs-basic/ObservationCharacteristics
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.8 Basic Observations - ObservationCollection

Conformance Class	**/conf/obs-basic/ObservationCollection**
Requirements	/req/obs-basic/ObservationCollection
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.9 Basic Observations - Observer

Conformance Class	**/conf/obs-basic/Observer**
Requirements	/req/obs-basic/Observer
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.10 Basic Observations - ObservingCapability

Conformance Class	**/conf/obs-basic/ObservingCapability**
Requirements	/req/obs-basic/ObservingCapability
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.3.11 Basic Observations - ObservingProcedure

Conformance Class	**/conf/obs-basic/ObservingProcedure**
Requirements	/req/obs-basic/ObservingProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4 Abstract tests for Conceptual Sample schema package

A.4.1 Conceptual Sample schema package

Conformance Class	**/conf/sam-cpt**
Requirements	/req/sam-cpt
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.2 Conceptual Sample - PreparationProcedure

Conformance Class	**/conf/sam-cpt/PreparationProcedure**
Requirements	/req/sam-cpt/PreparationProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.3 Conceptual Sample - PreparationStep

Conformance Class	**/conf/sam-cpt/PreparationStep**
Requirements	/req/sam-cpt/PreparationStep
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.4 Conceptual Sample - Sample

Conformance Class	**/conf/sam-cpt/Sample**
Requirements	/req/sam-cpt/Sample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.5 Conceptual Sample - Sampler

Conformance Class	**/conf/sam-cpt/Sampler**
Requirements	/req/sam-cpt/Sampler
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.6 Conceptual Sample - Sampling

Conformance Class	**/conf/sam-cpt/Sampling**
Requirements	/req/sam-cpt/Sampling
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.4.7 Conceptual Sample - SamplingProcedure

Conformance Class	**/conf/sam-cpt/SamplingProcedure**
Requirements	/req/sam-cpt/SamplingProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5 Abstract tests for Abstract Sample core package

A.5.1 Abstract Sample core package

Conformance Class	**/conf/sam-core**
Requirements	/req/sam-core
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.2 Abstract Sample core - AbstractPreparationProcedure

Conformance Class	**/conf/sam-core/AbstractPreparationProcedure**
Requirements	/req/sam-core/AbstractPreparationProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.3 Abstract Sample core - AbstractPreparationStep

Conformance Class	**/conf/sam-core/AbstractPreparationStep**
Requirements	/req/sam-core/AbstractPreparationStep
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.4 Abstract Sample core - AbstractSample

Conformance Class	**/conf/sam-core/AbstractSample**
Requirements	/req/sam-core/AbstractSample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.5 Abstract Sample core - AbstractSampler

Conformance Class	**/conf/sam-core/AbstractSampler**
Requirements	/req/sam-core/AbstractSampler
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.6 Abstract Sample core - AbstractSampling

Conformance Class	**/conf/sam-core/AbstractSampling**
Requirements	/req/sam-core/AbstractSampling
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.5.7 Abstract Sample core - AbstractSamplingProcedure

Conformance Class	**/conf/sam-core/AbstractSamplingProcedure**
Requirements	/req/sam-core/AbstractSamplingProcedure
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6 Abstract tests for Basic Samples package

A.6.1 Basic Samples package

Conformance Class	**/conf/sam-basic**
Requirements	/req/sam-basic
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.2 Basic Samples - MaterialSample

Conformance Class	**/conf/sam-basic/MaterialSample**
Requirements	/req/sam-basic/MaterialSample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.3 Basic Samples - NamedLocation

Conformance Class	**/conf/sam-basic/NamedLocation**
Requirements	/req/sam-basic/NamedLocation
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.4 Basic Samples - PhysicalDimension

Conformance Class	**/conf/sam-basic/PhysicalDimension**
Requirements	/req/sam-basic/PhysicalDimension
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.5 Basic Samples - Sample

Conformance Class	**/conf/sam-basic/Sample**
Requirements	/req/sam-basic/Sample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.6 Basic Samples - SampleCollection

Conformance Class	**/conf/sam-basic/SampleCollection**
Requirements	/req/sam-basic/SampleCollection
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.7 Basic Samples - Sampler

Conformance Class	**/conf/sam-basic/Sampler**
Requirements	/req/sam-basic/Sampler
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.8 Basic Samples - Sampling

Conformance Class	**/conf/sam-basic/Sampling**
Requirements	/req/sam-basic/Sampling
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.9 Basic Samples - SpatialSample

Conformance Class	**/conf/sam-basic/SpatialSample**
Requirements	/req/sam-basic/SpatialSample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.10 Basic Samples - StatisticalClassification

Conformance Class	**/conf/sam-basic/StatisticalClassification**
Requirements	/req/sam-basic/StatisticalClassification
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

A.6.11 Basic Samples - StatisticalSample

Conformance Class	**/conf/sam-basic/StatisticalSample**
Requirements	/req/sam-basic/StatisticalSample
Test purpose	Verify that all requirements from the requirements class have been fulfilled.
Test method	Inspect the documentation of the application, schema or profile.
Test Type	Capability

Annex B
(informative)

Common usage of O&M terminology

B.1 Introduction

This International Standard defines terminology in support of a generic, cross-domain model for observations and measurements. Terms are taken from a variety of disciplines. The terms are used within the model in a consistent manner, but in order to achieve internal consistency, this varies from how the same terms are used in some application domains. In order to assist in the correct application of the model across domains, this annex provides a mapping from Observations and Measurements (O&M) terminology to some domain vocabularies.

B.2 Earth Observations (EO)

Table B.1 — Earth Observations (EO)

O&M	EO	Example
Observation::result	Observation value, measurement value, observation	35 µg/m3
Observation::procedure	Method, sensor	ASTER, U.S. EPA Federal Reference Method for PM 2.5
Observation::observedProperty	Parameter, variable	Reflectance, Particulate Matter 2.5
Observation::proximateFeatureOfInterest:SpatialSample	2-D swath or scene	Sampling grid
SpatialSample:sampledFeature	Earth surface	http://sweetontology.net/realm/PlanetarySurface
Observation::ultimateFeatureOfInterest	Earth surface	http://sweetontology.net/realm/PlanetarySurface
Observation::proximateFeatureOfInterest:SpatialSample	3-D sampling space	Sampling grid
SpatialSample::sampledFeature	Media (air, water, …), Global Change Master Directory "Topic"	Troposphere
Observation::ultimateFeatureOfInterest	Media (air, water, …), Global Change Master Directory "Topic"	Troposphere

B.3 Metrology

Table B.2 — Metrology

O&M	Metrology	Example: mass measurement
Observation::result	Value	35 mg
Observation::procedure	Instrument	Balance
Observation::observedProperty	Measurand	Mass

B.4 Earth science simulations

Table B.3 — Earth science simulations

O&M	Earth science
Observation::result	A model or field
Observation::observedProperty	Variable, parameter
Observation::proximateFeatureofInterest:SpatialSample	Section, swath, volume, grid
Observation::proximateFeatureofInterest:SpatialSample::sampledFeature	Atmosphere, ocean, solid earth
Observation::ultimateFeatureofInterest	Atmosphere, ocean, solid earth
Observation::procedure	Earth process simulator
Observation::phenomenonTime	Future date (forecasts), past date (hindcasts)
Observation::resultTime	Simulator execution date
Observation::validTime	Period when result is intended to be used

B.5 Assay/Chemistry

Table B.4 — Assay/Chemistry

O&M	Geochemistry
Observation::proximateFeatureOfInterest:MaterialSample	Sample
MaterialSample::sampledFeature:GeologicUnit	Ore body, Geologic Unit
MaterialSample::relatedSample:MaterialSample	Pulp, separation
MaterialSample::preparationStep	Sample preparation process
MaterialSample::sampling:Sampling:samplingProcedure	Sample collection process
MaterialSample::sourceLocation	Sample collection location
MaterialSample::size	Mass, length
MaterialSample::storageLocation	Store location
MaterialSample::sampling:Sampling:time	Sample collection date
Observation::phenomenonTime	Sample collection date
Observation::resultTime	Analysis date
Observation::result	Analysis

Observation::observedProperty	Analyte
Observation::procedure	Instrument, analytical process

B.6 Geology field observations

Table B.5 — Geology field observations

O&M	Geology
Observation::proximateFeatureOfInterest:SampleCollection	Outcrop
SampleCollection::member:SpatialSample	Location of structure observation
SpatialSample::sampledFeature:GeologicUnit	Geologic Unit
Observation::phenomenonTime	Outcrop visit date
Observation::observedProperty	Strike and dip, lithology, alteration state, etc.
SampleCollection::member:MaterialSample	Rock sample
MaterialSample::sampledFeature:GeologicUnit	Ore body, Geologic Unit

B.7 Geotechnics observations

Table B.6 — Geotechnics observations

O&M	Geotechnical in situ test
Observation::result	A log
Observation::observedProperty	A soil property (eg. gamma ray, resistivity, sound speed propagation)
Observation::proximateFeatureofInterest:SpatialSample	The borehole trajectory
Observation::proximateFeatureofInterest:SpatialSample::sampledFeature	A part of the Earth
Observation::ultimateFeatureofInterest	A part of the Earth
Observation::procedure	Geotechnical test procedure
Observation::phenomenonTime	Date and time of the test
Observation::resultTime	Date and time of the test
Observation::validTime	Date and time of the test

B.8 Water quality observations

Table B.7 — Water quality observations

O&M	Water quality
Observation::proximateFeatureOfInterest:SpatialSample	Water quality station at Cénac (France)
SpatialSample::sampledFeature:WaterBody	River (e.g. the Dordogne river)
SpatialSample::relatedSample:MaterialSample	Water Sample as sampled on-site
MaterialSample::sampledFeature:WaterBody	River (e.g. the Dordogne river)
MaterialSample::relatedSample:MaterialSample	Filtered sample (sub-sample of the initial one)
MaterialSample::sampledFeature:MaterialSample	The initial water sample that was sub-sampled
MaterialSample::preparationStep	Sample preparation process
MaterialSample::sampling:Sampling:samplingProcedure	Sample collection process
MaterialSample::sourceLocation	Sample collection location
MaterialSample::size	Volume of the water sampled
MaterialSample::storageLocation	Store location
MaterialSample::sampling:Sampling:time	Sample collection date
Observation::phenomenonTime	Sample collection date
Observation::resultTime	Analysis date
Observation::result	Analysis
Observation::observedProperty	Analyte (Nitrates, Phosphates …)
Observation::procedure	Instrument, analytical process (e.g. NF EN ISO 13395 Octobre 1996 / T90-012)

B.9 Soil quality observations

Table B.8 — Soil quality observations

O&M	Soil quality
Observation::proximateFeatureOfInterest:MaterialSample	A sub sample or the initial soil sample
MaterialSample::relatedSample:MaterialSample	Soil sample (can be a drilling core)
MaterialSample:relatedSample:SpatialSample	The borehole that was drilled and the core extracted from
Observation::ultimateFeatureOfInterest	Part of the lithosphere
MaterialSample::preparationStep	Sample preparation process
MaterialSample::sampling:Sampling:samplingProcedure	How the sample was collected or prepared
MaterialSample::sourceLocation	Where the sample has been collected
MaterialSample::storageLocation	Where the sample is stored
MaterialSample::sampling:Sampling:time	When the sample was collected
Observation::phenomenonTime	Sample collection date
Observation::resultTime	Analysis date
Observation::result	The result of the analysis
Observation::observedProperty	The analysed property (generally concentration of a constituent)
Observation::procedure	The analysis method

Annex C
(informative)

Alignment with ISO 19156:2011

In order to ease in the understanding of this revised version but also to support implementation of it an alignment is proposed with ISO 19156:2011.

For the sake of readability:

— it starts with this revised version and maps it to ISO 19156:2011, and

— elements introduced by this version don't appear in the alignment proposed.

As the UML mechanics involved between ISO 19156:2011 and this current revision are really different, this annex is informative. The mapping provided only involves Basic packages from the current revision.

C.1 Observation Model

Basic Observations:Observation	equivalent class	19156:2011:OM_Observation
Basic Observations:Observation.parameter	equivalent property	19156:2011:OM_Observation.parameter
Basic Observations:Observation.phenomenonTime	equivalent property	19156:2011:OM_Observation.phenomenonTime
Basic Observations:Observation.resultQuality	equivalent property	19156:2011:OM_Observation.resultQuality
Basic Observations:Observation.resultTime	equivalent property	19156:2011:OM_Observation.resultTime
Basic Observations:Observation.validTime	equivalent property	19156:2011:OM_Observation.validTime
Basic Observations:Observation.result	equivalent property	19156:2011:OM_Observation.result
Basic Observations:Observation.ultimateFeatureOfInterest	sub-property of	19156:2011:OM_Observation.featureOfInterest
Basic Observations:Observation.proximateFeatureOfInterest	sub-property of	19156:2011:OM_Observation.featureOfInterest
Basic Observations:Observation.observedProperty	equivalent property	19156:2011:OM_Observation.observedProperty

Basic Observations:Observation.procedure	equivalent property	19156:2011:OM_Observation.procedure
Basic Observations:ObservingProcedure	equivalent class	19156:2011:OM_Process
Basic Observations:Observation.metadata	equivalent property	19156:2011:OM_Observation.metadata
Basic Observations:Observation.relatedObservation	equivalent property	19156:2011:OM_Observation.relatedObservation

C.2 Sample Model

Basic Samples:Sample	equivalent class	19156:2011:SF_SamplingFeature
Basic Samples:Sample.metadata	has subProperty	19156:2011:SF_SamplingFeature.lineage
Basic Samples:Sample.parameter	equivalent property	19156:2011:SF_SamplingFeature.parameter
Basic Samples:Sample.sampledFeature	equivalent property	19156:2011:SF_SamplingFeature.sampledFeature
Basic Samples:Sample.relatedObservation	equivalent property	19156:2011:SF_SamplingFeature.relatedObservation
Basic Samples:Sample.relatedSample	equivalent property	19156:2011:SF_SamplingFeature.relatedSamplingFeature
Basic Samples:SpatialSample	equivalent class	19156:2011:SF_SpatialSamplingFeature
Basic Samples:SpatialSample.horizontalPositionalAccuracy	equivalent property	19156:2011:SF_SpatialSamplingFeature.positionalAccuracy
Basic Samples:SpatialSample.verticalPositionalAccuracy	equivalent property	19156:2011:SF_SpatialSamplingFeature.positionalAccuracy
Basic Samples:SpatialSample.shape	equivalent property	19156:2011:SF_SpatialSamplingFeature.shape
Basic Samples:Sample.preparationStep	equivalent property	19156:2011:SF_SpatialSamplingFeature.hostedProcedure
Basic Samples:MaterialSample	equivalent class	19156:2011:SF_Specimen

Basic Samples:MaterialSample.storageLocation	equivalent property	19156:2011:SF_Specimen.curren Location
Basic Samples:MaterialSample.sourceLocation	equivalent property	19156:2011:SF_Specimen.sampli ngLocation
Basic Samples:MaterialSample.sampling.Sampling.sam plingProcedure	equivalent property	19156:2011:SF_Specimen.sampli ngMethod
Basic Samples:MaterialSample.sampling.Sampling.tim e	equivalent property	19156:2011:SF_Specimen.sampli ngTime
Basic Samples:MaterialSample.size	equivalent property	19156:2011:SF_Specimen.size
Basic Samples:Sample.sampleType	equivalent property	19156:2011:SF_Specimen.specim enType
Basic Samples:Sample.preparationStep	equivalent property	19156:2011:SF_Specimen.process ingDetails
Basic Samples:SampleCollection	equivalent class	19156:2011:SF_SamplingFeature Collection

Annex D
(informative)

Best practices in use of the Observation and Sampling models

D.1 Features, coverages and observations — Different views of information

ISO 19109 describes the feature as a “fundamental unit of geographic information”. The “General Feature Model” (GFM) presented in ISO 19101 and ISO 19109 defines a feature type in terms of its characteristic set of properties, including attributes, association roles, and behaviours, as well as generalization and specialization relationships, and constraints.

Typical concrete feature types have names like “road”, “watercourse”, “mine”, “atmosphere”, etc. For a road, the set of properties might include its name, its classification, the curve describing its centreline, the number of lanes, the surface material, etc. The complete description of a road instance, therefore, is the set of values for the set of properties that define a road type. This use of the feature model is object-centric, and supports a viewpoint of the world in terms of the set of discrete identifiable objects that occupy it.

The principal alternative model for geographic information is the coverage, described in ISO 19123. This viewpoint focuses on the variation of a property within the (spatiotemporal) domain of interest. The domain might be a scene, a grid, a transportation network, a volume, a set of sampling stations, etc. The range of the coverage can be any property, such as reflectance, material type, concentration of some pollutant, number of lanes, etc. But the key to the coverage viewpoint is that it is property-centric, concerning the distribution of the values of a property within its domain space.

These viewpoints are not exclusive, and both are used in analysis and modelling. For example, a feature might be detected from the analysis of variation of a property in a region of interest (e.g. an ore-body from a distribution of assay values). Also, for some feature types, the value of one or more properties might vary across the feature, in which case the shape of the feature provides the coverage domain (e.g. ore-grade within a mine).

Observations focus on the data collection event. An act of Observation serves to assign a value to a property of a feature. If the property is non-constant, the value is a function or coverage. The results of a set of observations of different properties on the same feature-of-interest can provide a complete description of the feature instance. Alternatively, the results of a set of observations of the same property on a set of different features provide a discrete coverage of that property over a domain composed of the geometry of the feature set. The result of an observation of one property on one feature over time is a Temporal Coverage/Time-Series. The other properties of the Observation are metadata concerning the estimation of the value(s) of a property on a feature-of-interest.

In particular, Observations concern properties (e.g. shape, colour) whose values are determined using an identifiable procedure, in which there is a finite uncertainty in the result. This can be contrasted with properties whose values are specified by assertion (e.g. name, owner) and are therefore exact. The observation instance provides “metadata” for the property value-estimation process.

An observation event is clearly a “feature” in its own right, according to the GFM definition. An observation instance is a useful unit of information, therefore observation is a feature type.

Transformation between viewpoints is frequently required.

This is illustrated in Figure D.1, which schematically shows a dataset comprising values of a set of properties at a set of locations. A row of the table provides the complete description of the properties at a single location. This is a representation of a potential feature description. A column of the table describes the variation of a single property across the set of locations. This is a representation of a discrete coverage. A single cell in the table provides the value of a single property on a single feature. This might be the result of an observation.

Observations, Coverage and Feature representations might be associated with different phases of the data-processing cycle or value-chain:

— The observation view is associated with data collection, when an observation event causes values for a property of a feature to be determined, and during data entry when the data-store is updated by inserting values into fields in the datastore.

— A coverage view can be assembled from results of observations of a specific property, and represents data assembled for analysis, when the objective is to find signals in the variation of a property over a domain.

— A discrete feature description is a “summary” viewpoint, assembled from results of observation on the same target, or an “inferred” viewpoint, by extraction of a signal from a coverage.

Observations, Coverage and Feature representations are also often interlinked. Just as an Observation references the Feature it provides property information for, the Feature representation may also reference known Observations with more detailed property information. The same applies to Observations and Coverages; just as a Coverage can be the result of an Observation, an Observation can also be utilized to provide valuable meta-information on how the values being provided in the Range of the Coverage were derived.

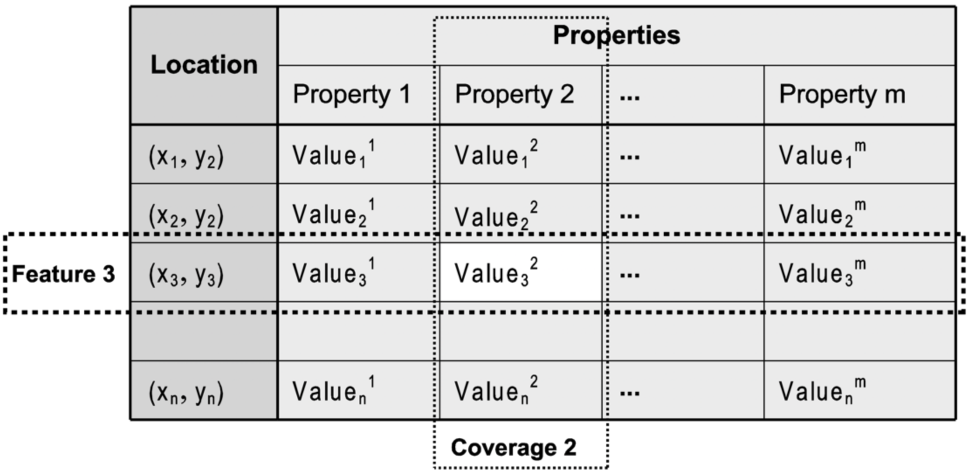

Figure D.1 — Tabular representation of information associated with a set of locations

D.2 Observation concerns

D.2.1 Domain specialization

Specialization of the observation model for an application domain is accomplished primarily using a domain model and its feature-type catalogue. For example, an instance of a feature type in the domain feature-type catalogue will provide the ultimate feature-of-interest for the investigation of which the observation is a part, and the characteristic properties of the feature type provide potential observed properties. A description of a sensor type or process familiar within the application domain is the value of the observation procedure, while the explicit device or person performing this procedure is provided as the observer.

The observation model encourages encapsulation of domain specialization in the associated classes, while the observation class itself rarely needs specialization. Nevertheless, other choices could be made in partitioning information between the classes in the model. For some applications, it might be convenient for information that is strictly associated with a second-layer object (procedure, feature-of-interest) to be associated with a specialized observation type.

For example, when measuring chemistry or contamination, the process often involves retrieving material samples from a sampling site, which are then sent to a laboratory for analysis. The material sample is a very tangible feature instance, with an identity. For some applications, it might be important to recognize the existence of the material sample, and retain a separate description of it. However, in other applications, particularly when the focus is on monitoring the change in a property at a sampling site, the existence of a series of distinct material samples is of minor or no interest. In this case, creating a series of objects and identifiers is superfluous to the user's requirements.

In certain cases, some additional properties strictly associated with such a material sample must also be recorded, an example is the "sampling elevation" in a water or atmospheric column. A number of choices can be made. For example, the elevation could be:

a) a property of each distinct material sample on which atomic observations are actually made,

b) a property of the sampling site(which would require distinct sites for all elevations at which observations are made),

c) a parameter of the observation procedure (which makes the procedure specific to this observation series only), or

d) a parameter of the observation event, either using the soft-typed arbitrary event-specific parameter, or through specialization of the observation type.

Any of these is a legitimate approach. The optimum one will be dependent on the application.

All of the classes in the models presented here for observations and procedures can be further specialized for domain-specific purposes, whereby the abstract classes provided in the Abstract Core models have been specifically foreseen as a neutral basis for such domain extensions. Additional attributes and associations can be added as necessary.

EXAMPLE "Assay" might be derived from Observation, fixing the observedProperty to be "ChemicalConcentration" and adding an additional attribute "analyte".

D.2.2 Comparison with provider-oriented models

The O&M model is intended to provide a basic output- or user-oriented information model for sensor web and related applications. The goal is to provide a common language for discourse regarding sensor, sample and observation systems.

In comparison, SensorML [16] has a process- or provider-oriented data model. These are usually used to describe data at an early stage in the data processing and value-adding chain. This might be prior to the details of the feature-of-interest and observed property being assembled and assigned to the result in a way that carries the key semantics to end-users of observation data. In particular, part of a SensorML datastream might include information that must be processed to determine the position of the target or feature-of-interest. At the early processing stage such positional and timing information might be embedded within the result.

Nevertheless, even within these low-level models the O&M formalization can be applied. The proximate feature-of-interest is the vicinity of the sensor. The observed property is a composite type including components representing observation timing, and position and attitude of a sensor, etc. This must be processed to obtain the details of the ultimate feature-of-interest. The procedure is a description of sensor methodology including elements that capture all of the elements of the composite phenomenon or property type, etc. while the observer references the explicit sensor utilized.

D.2.3 Observation discovery and use

The Observation and Measurements model presented here offers a user-oriented viewpoint. The information object is characterized by a small set of properties, which are likely to be of interest to a user for discovery and request of observation data. The user will typically be interested primarily in a feature-of-interest, or the variation of a phenomenon. The model provides these items as first-order elements. An interface to observation information should expose these properties explicitly.

Observation discovery and use is often done querying APIs; although with LinkedData practices being more and more used, one might discover an observation simply because an instance of a domain feature uses its URI or it has been crawled by a search engine bot.

Observation oriented APIs, be them from the previous generation (OGC SOS [17]) or the current one (OGC SensorThings API[18]) share commonalities in the way they approach this topic. They both leverage the Observations and Measurements model to directly allow filtering on featureOfInterest, observedProperty and procedure.

— The SensorThings API model and OData query graph allow filtering on all aspects of the observational data model, both for discovery and data retrieval (both ‘operations’ being intertwined in the REST pattern)

— SOS [17], having these three concepts as classifiers for an observationOffering in the capabilities description, allows them to be used for discovery and as explicit parameters in the GetObservation request

From a user point of view, these associated objects (procedure, target feature, phenomenon) are primarily metadata, which are only of interest to specialists during discovery, and then to assist evaluation or processing of individual results.

Each of these associated objects might require a complex description. Hence they are modelled as distinct classes, which can be as simple or complex as necessary.

In a serialized representation (e.g. JSON, XML following the GML pattern, etc…), they might appear inline, perhaps described using one of the models presented here, or they can be indicated by reference using a URI [4]. The URI identifier might be a URL link or service call, which should resolve immediately to yield a complete resource. Or it might be a canonical identifier, such as a URN, which the user and provider are preconfigured to recognize and understand.

On the other hand, SensorML takes a process- or provider-oriented viewpoint. Discovery and request is based primarily on the user having knowledge of specific sensor systems and their application. While this is a reasonable assumption within technical communities, specialist knowledge of sensor systems would not be routinely available within a broader set of potential users of sensor data, particularly as this is made widely available through interfaces like OGC SensorThings API and SOS.

D.2.4 Observations, interpretations, simulations

Some conceptual frameworks make a fundamental distinction between observations, interpretations and simulations as the basis for their information modelling approach. This supports a pattern in which observations are given precedence and archived, while interpretations or simulations are more transient, being the result of applying the current algorithms and paradigms to the currently available observations.

An alternative view is that the distinction is not absolute, but is one of degree. Even the most trivial "observations" are mediated by some theory or procedure. For example, the primary measurement when using a mercury-in-glass thermometer is the position of the meniscus relative to graduations. This allows the length of the column to be estimated. A theory of thermal expansion plus a calibration for the physical realization of the instrument allows conversion to an inferred temperature. Other observations and measurements all involve some kind of processing from the primary observable property. For modern instruments, the primary observable property is almost always voltage or resistance or frequency from some kind of sensing element, so the "procedure" typically involves calibrations, etc., built on a theory of operation for the sensor. Pertaining to simulations, the O&M model allows for the description of simulated observations (ex : forecast) and can capture entire processing chains starting from initial observation(s) (e.g. surface/ground water level, rainfall) to generate corresponding forecasts scenarios (e.g. flood, drought) through the use of simulation algorithms. Similarly, aggregates can be calculated (e.g. averages over space or time) and provided referencing their primary source observations.

However, the same high-level information model — that every "value" is an estimate of the value of a property, generated using a procedure and inputs — applies to "observations", "interpretations" as well as “simulations”. It is just that the higher the semantic value of the estimate, the more theory and processing is involved.

Within the model provided in this standard, there is no conceptual distinction between observations, interpretations and simulations. The O&M model allows for the description of the observational workflow together with the explicit description of the processing chain instance that has taken a more primitive observation (e.g. an image) and retrieved a higher level observation (e.g. the presence of a certain type of feature instance) through the application of one or more processing steps. The result is the entire continuum of primary and processed data provided in a harmonized model.

D.3 Sample, Sampling concerns

D.3.1 Sample as observation-collector

The sample model provides

a) an intermediate Sample class that allows the assignment of primitive and intermediate properties within a processing chain,

b) three sub-types of Samples corresponding to practices applied by communities where Sample are either defined by their geospatial characteristics, statistical characteristics or their material ones (being taken ex-situ for further observation), and

c) additional classes providing a context for the description of sampling acts and regimes.

d) In addition, the sample model allows for references to observation(s) concerning a shared common feature-of-interest / sampledFeature. This provides an access route to observation information that is convenient under some project scenarios, where the sampling strategy provides the logical organization of observations.

EXAMPLE An observational mission or campaign might organize its data according to flightlines, ship's tracks, outcrops, sampling-stations, quadrats, etc., or an observation archive or museum might organize observations by specimen (a specific type of material sample targeting preservation).

D.3.2 Observation feature(s)-of-interest

Application of the Observations and Measurements model requires careful attention to identify the feature-of-interest context correctly. This can be straightforward if the observation is clearly concerned with an easily identified concrete feature type from a domain model. However, the ultimate feature-of-interest to the investigator might not be the proximate feature-of-interest for the observation. In some cases, a careful analysis reveals that the type of the feature-of-interest had not previously been identified in the application domain.

The key is that the proximate feature-of-interest must be capable of carrying this result as the value or component of the value of a relevant property. So a useful approach in analysis is to consider what the result of the observation is, and then the feature-of-interest can be deduced since it must have a property with this result as its value. If an observation produces a result with several elements, or if there are a series of related observations with different results, then this might help further refine the understanding of the type of the ultimate feature-of-interest.

EXAMPLES

— In groundwater monitoring the ultimate feature-of-interest is often a given hydrogeological unit but the proximate feature-of-interest is the Well (or a more precise Feature) where the Observation occurs.

— In air quality monitoring the ultimate feature-of-interest is either the general atmosphere or alternatively a defined region (e.g. air quality zone) the monitoring facility, while the proximate feature-of-interest is the bubble of air around the air intake of the monitoring facility

— In surface water quality monitoring the ultimate feature-of-interest is a river (or a section of it) but the proximate feature-of-interest of the observation is a vial of water (material sample) on which analysis are conducted in a laboratory.

D.3.3 Processing chains and intermediate features-of-interest

The Observation model implies a direct relationship between the observed property and the type of the feature-of-interest (e.g. a material sample type has a property 'mass' and the observation's observed property is 'mass'). However, as discussed in section 7.2.2.2 the relationship between the observed property and property(ies) of the ultimate feature-of-interest is often more complex.

The Sample model is a mechanism for preserving the strict association, by providing a specific intermediate feature type whose observable properties are unspecified in advance, but supplied through an unlimited set of related observations. The path from a sensed property obtained through observations related to the sample, to the interesting property on the ultimate feature-of-interest, is modelled as a processing chain.

If intermediate values are explicit, then the processing chain can be modelled as a sequence of "observations", with intermediate features of interest carrying intermediate property types. Each intermediate value must apply to a feature-of-interest that bears this property, or a sampling feature. Note that the types of these features might not be conventional or immediately recognisable, but the coherence of the Observations and Measurements model does imply their existence. Hence, if any intermediate result is made explicit, then a suitable intermediate feature must also be identified.

D.4 Observations and Coverages

Within the Open Geospatial Consortium (OGC), different data models have evolved for the provision of sensor data (Observation & Measurements Model (O&M)) and datacubes (OGC Coverage Implementation Schema (CIS))[25]. While these models are formally distinct, and were developed mostly independently of each other, there are great similarities as well as overlaps between these models. At its core, the O&M model provides an exact description of how an observation or measurement value came to be, while the CIS model has concerned itself with providing alignment with a spatial swath and data recorded for this region; these differing approaches have led to a focus on different aspects of the entirety of observational data and measurement metadata. In addition, these models are often used conjunctively, as each model provides relevant information missing from the other model.

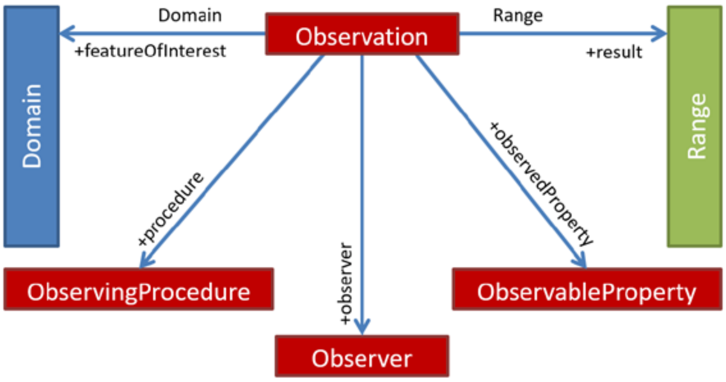

Figure D.2 — O&M model key elements

Upon closer analysis it becomes clear just how similar these models are at their core, as well as how each provides concepts essential for precise description of the data that have been neglected within the other models. Both O&M and CIS provide a set of values (Range) over a given extent (Domain). However, while the CIS model provides information on the explicit points within the Domain extent for which values are provided (domainSet, usually some sort of grid) as well as the mapping of these points to these values provided within the Range (provided via the coverageFunction), the O&M model provides far more detailed information on the measurement methodology and process via the ObservableProperty, ObservingProcedure and Observer types.

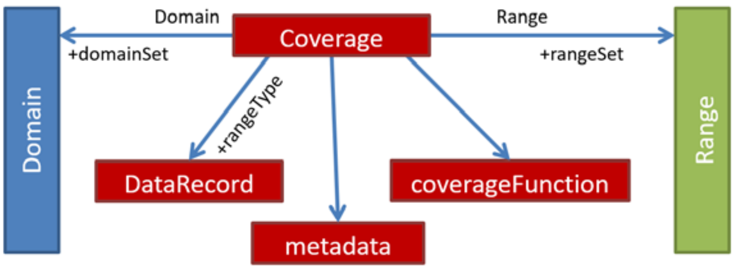

Figure D.3 — CIS model key elements

When O&M and CIS models are used in conjunction, care must be taken in ensuring alignment pertaining to the Domains being referenced. The observation community often provides domain features with a bounding polygon as the Domain of complex Observations, assuming the explicit points within for which values are provided to be contained within the coverage provided as a result of the Observation. Under the CIS model the Domain is always provided together with some explicit representation of the actual points within the Domain for which values are provided, e.g. origin and offsets for the definition of regular grid points.

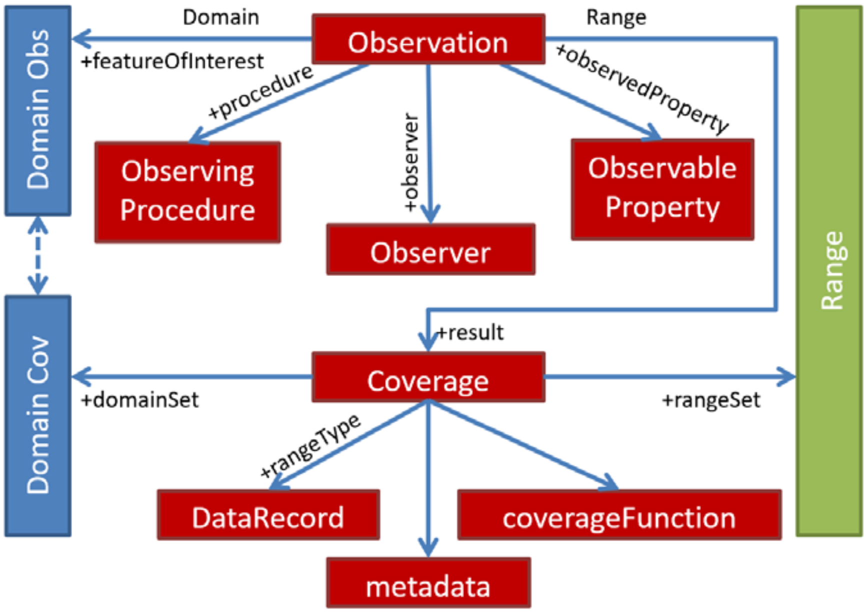

Figure D.4 — Coverage as a result of an Observation

For example when providing data on a transect or vertical profile, the ultimateFeatureOfInterest (O&M Domain) may reference a feature representing this transect or profile, while the Coverage provided as result (O&M Range) contains both the explicit measurement locations (CIS Domain), often as offsets along the given transect or profile, and the measurement values (CIS Range).

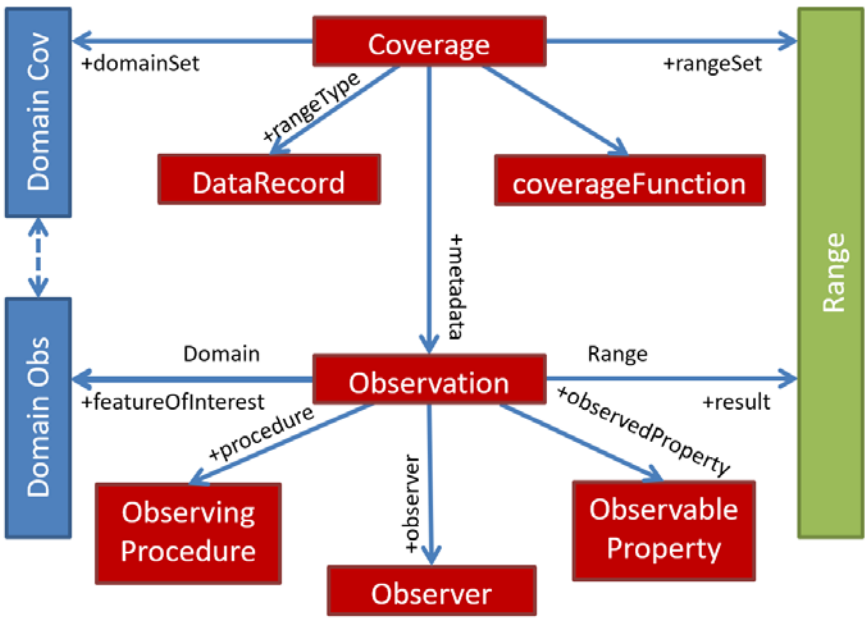

Figure D.5 — Observation as metadata of a Coverage.

Conversely to the model described above, O&M Observations have long been utilized for the provision of more explicit metadata on how the values provided in the rangeSet have been ascertained, whereby the Observation result was left as void. In this updated version, the ObservationCharacteristics type has been foreseen for utilization or extension within this context, as the constraints on this type are far looser than on the Observation. When O&M and CIS models are used in conjunction, it is recommended that the O&M Domain provided as ultimateFeatureOfInterest is an envelope of the CIS Domain.

Bibliography

[1] Chrisman, N.R. *Exploring Geographical Information Systems*, 2nd Edition. Wiley. 2001

[2] Fowler, M. *Analysis Patterns: reusable object models*. Addison Wesley Longman, Menlo Park, CA. 1998

[3] *GML Encoding of Discrete Coverages (interleaved pattern),* OpenGIS® Best Practice OGC document 06188r1

[4] IETF RFC 2396, *Uniform Resource Identifiers (URI): Generic Syntax.* August 1998

[5] *VIM3: International vocabulary of metrology – Basic and general concepts and associated terms* : BIPM/ISO 2012

[6] ISO/TS 19101-2, 2008, *Geographic information — Reference model — Part 2: Imagery*

[7] ISO 19115-2:2019, *Geographic information — Metadata — Part 2: Extensions for imagery and gridded data*

[8] (removed as no longer relevant)

[9] ISO 19143:2010, *Geographic information — Filter encoding*

[10] Krantz, D.H., Luce, R.D., Suppes, P., Tversky, A. (1971), *Foundations of measurement, Vol. I: Additive and polynomial representations*, New York: Academic Press

[11] Luce, R.D., Krantz, D.H., Suppes, P., Tversky, A. (1990), *Foundations of measurement, Vol. III: Representation, axiomatization, and invariance*, New York: Academic Press

[12] Nieva, T. *Remote data acquisition of embedded systems using internet technologies: a role-based generic system specification.* Thesis, Ecole Polytech. Fed. Lausanne 2001. Available (viewed 2020-09-29) at http://infoscience.epfl.ch/record/313/files/Nieva01.pdf

[13] *Object Constraint Language (OCL) v2.0*. OMG Available Specification formal/06-05-01. Object Management Group, Needham, Mass. USA

[14] Sarle, W.S., *Measurement theory: frequently asked questions*. Originally published in the Disseminations of the International Statistical Applications Institute, 4th edition, 1995, Wichita: ACG Press, pp. 6166. Revised 1996, 1997. Available (viewed 2020-09-2) at ftp://ftp.sas.com/pub/neural/measurement.html

[15] Schadow, G., McDonald, C.J. (eds.), *UCUM, Unified Code for Units of Measure*. Available (viewed 2020-09-29) at https://ucum.org/trac. Tentative ontology at http://finto.fi/ucum/en/ (viewed 2020-09-24)

[16] *Sensor Model Language (SensorML)*, OpenGIS® Implementation Standard, OGC 12-000r2. Available (viewed viewed 2020-09-29) at http://www.opengeospatial.org/standards/sensorml

[17] *Sensor Observation Service*, OpenGIS® Implementation Specification OGC document 12-006

[18] The OGC SensorThings API Part 1: Sensing (2016). OGC Document OGC: 15-078R6,

[19] Stevens, S.S. On the theory of scales of measurements. *Science* 1946, **103**, pp. 677680

[20] Suppes, P., Krantz, D.H., Luce, R.D., Tversky, A. (1989), *Foundations of measurement, Vol. II: Geometrical, threshold, and probabilistic representations*, New York: Academic Press

[21] *SWE Common Data Model Encoding Standard,* OpenGIS® Implementation Standard OGC document 08094r1

[22] (removed as no longer relevant)

[23] Yoder, J.W., Balaguer, F., Johnson, R. *From analysis to design of the observation pattern*. Available (viewed 2011-10-14) at citeseerx.ist.psu.edu

[24] *OGC: The Specification Model - A Standard for Modular specifications (2009). OGC document 08-131r3,*

[25] K. Schleidt and P. Baumann, "Interconnecting Sensor Data and Datacubes," *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan, 2019, pp. 5555-5558, doi: 10.1109/IGARSS.2019.8898232.

[26] *QUDT - Quantities, Units, Dimensions and Data Types Ontologies*. Ralph Hodgson; Paul J. Keller; Jack Hodges; Jack Spivak. Available (viewed 2020-09-29) at http://www.qudt.org/

[27] *Semantic Sensor Network Ontology.* Armin Haller, Krzysztof Janowicz, Simon Cox, Danh Le Phuoc, Kerry Taylor, Maxime Lefrançois. Available (viewed 2020-09-29) at https://www.w3.org/TR/vocab-ssn/

[28] *Guidelines for the use of Observations & Measurements and Sensor Web Enablement-related standards in INSPIRE*. Sylvain Grellet , Gerhard Dünnebeil, Anders Foureaux, Carsten Hollmann, Frédéric Houbie, Diomede Illuzzi, Simon Jirka, Barbara Magagna, Matthes Rieke, Alessandro Sarretta, Katharina Schleidt, Paweł Soczewski, Paolo Tagliolato, Mickael Treguer and Alexander Kotsev, Michael Lutz. Available (viewed 2020-09-29) at https://inspire.ec.europa.eu/id/document/tg/d2.9-o%26m-swe

[29] *Ontology for observations and sampling features, with alignments to existing models*. S.J.D. Cox. Semantic Web. 2017. Available (viewed 2020-09-29) at https://content.iospress.com/articles/semantic-web/sw214

